

# PROJECT SISTEM KONTROL BERBASIS ARDUINO

Oleh :

**Ahmad Sahru Romadhon**

**Faikul Umam**



# PROJECT SISTEM KONTROL BERBASIS ARDUINO

© 2021

## Penulis:

Ahmad Sahru Romadhon  
Faikul Umam

## Desain Cover & Penata Isi

Tim MNC Publishing

Cetakan I, Desember 2021

## Diterbitkan oleh :



**Media Nusa Creative**

Anggota IKAPI (162/JTI/2015)

Bukit Cemara Tidar H5 No. 34, Malang

Telp. : 0812.3334.0088

E-mail : [mncpublishing.layout@gmail.com](mailto:mncpublishing.layout@gmail.com)

Website : [www.mncpublishing.com](http://www.mncpublishing.com)

**ISBN 978-602-462-810-9**

Hak Cipta dilindungi undang-undang. Dilarang memperbanyak atau memindahkan sebagian atau seluruh isi buku ke dalam bentuk apapun, secara elektronik maupun mekanis, termasuk fotokopi, merekam, atau dengan teknik perekaman lainnya, tanpa izin tertulis dari Penulis dan/ atau Penerbit. Undang-Undang Nomor 19 Tahun 2000 tentang Hak Cipta, Bab XII Ketentuan Pidana, Pasal 72, Ayat (1), (2), dan (6)

# KATA PENGANTAR

Puji syukur kehadirat Allah SWT yang telah memberikan rahmat dan karunia-NYA, sehingga kami dapat menyelesaikan buku "Project Sistem Kontrol Berbasis ARDUINO" untuk Mahasiswa dan Umum.

Arduino adalah platform prototyping opensource hardware yang dapat digunakan untuk membuat proyek berbasis pemrograman. Arduino dirancang untuk memudahkan pengguna elektronik atau siapapun yang ingin mengembangkan peralatan elektronik interaktif berdasarkan hardware dan software yang fleksibel dan mudah untuk digunakan.

Buku ini menjelaskan tentang Arduino secara lengkap dan disertai dengan contoh Project yang menggunakan Arduino. Buku ini berisikan tentang Dasar-dasar mikrokontroler, Pengenalan Arduino, Pengenalan Aplikasi Arduino IDE, dan berbagai project berbasis Arduino.

Kami ucapkan terima kasih kepada semua pihak yang turut membantu dalam penulisan dan penerbitan buku ini. Kami sadari, masih banyak kekurangan dan kelemahan dalam penyusunan buku ini, untuk itu kritik dan saran akan penulis terima dengan senang hati. Semoga buku ini dapat memberikan manfaat bagi kita semua.

Bangkalan, 1 Desember 2021



# DAFTAR ISI

<b>KATA PENGANTAR .....</b>	<b>iii</b>
<b>DAFTAR ISI.....</b>	<b>v</b>

## **BAB 1**

<b>PENGANTAR TEKNOLOGI MIKROKONTROLER.....</b>	<b>1</b>
1. Sejarah Mikrokontroler .....	1
2. Perkembangan Teknologi Mikrokontroler.....	4
3. Implementasi Teknologi Mikrokontroler .....	6
4. Ringkasan.....	7

## **BAB 2**

<b>MIKROKONTROLER PLATFORM ARDUINO .....</b>	<b>9</b>
1. Konsep Arduino.....	9
2. Perangkat Lunak Arduino.....	14
3. Ringkasan.....	18

## **BAB 3**

<b>PEMROGRAMAN ARDUINO.....</b>	<b>21</b>
1. Arduino Dasar-Dasar Mikrokontroler dan Arduino .....	21
2. Jenis-Jenis Arduino .....	24
3. Konsep Dasar Pemrograman Arduino .....	38
4. Program Sederhana .....	41
5. Ringkasan.....	44

## **BAB 4**

<b>PENGGUNAAN SENSOR DALAM ARDUINO .....</b>	<b>45</b>
1. Sensor Cahaya (LDDR) .....	45
2. Sensor Cahaya (HC-SR04) .....	48
3. Sensor DHT11.....	55
4. Sensor IR Proximity .....	63
5. Sensor Sentuh.....	66

6. Sensor Suara .....	70
7. Sensor Warna (TCS34725).....	73
8. Sensor Gyroscope (MPU-6050) .....	78
9. NodeMCU ESP8266.....	87

**BAB 5**

<b>INTERCAFE SERIAL I2C .....</b>	<b>93</b>
1. Pengenalan I2C.....	93
2. RTC sebagai Interface I2C.....	95
3. Interface I2C pada Arduino.....	97
<b>DAFTAR PUSTAKA.....</b>	<b>99</b>

# RINGKASAN

**Buku Arduino** ini saya tulis, karena saya melihat Arduino ini tren-nya bagus ke depannya. Bukan saja karena mudah penggunaannya (*user friendly*), tetapi ada komunitas penggemar yang turut membesarkannya. Komunitas yang tersebar di seluruh dunia ini sangat aktif melakukan riset, eksperimen dan diskusi sehingga teknologi Arduino terus berkembang dan semakin menarik. Tentu ini jadi semacam *credit point* untuk Arduino, karena akan memberikan banyak manfaat bagi penggunanya. Bagi pemula, pelajar dan kalangan akademisi akan banyak mendapat referensi gratis tentang Arduino, bagi kalangan *hobbyist* tentu akan mendapat banyak contoh-contoh project, bagi kalangan profesional tentu Arduino akan banyak mendapat alternatif dalam menyelesaikan project-nya, dan bagi kalangan bisnis, tentu akan mendapat *niche* bisnis yang *sustainable* karena pengguna Arduino semakin hari semakin bertambah.

Di dalam buku disertai contoh-contoh projek yang lengkap, melibatkan berbagai jenis sensor dan motor, sehingga bisa menjadi bahan referensi atau pengembangan projek yang lebih kompleks sesuai kebutuhan pengguna.

Pembahasan buku meliputi pengenalan komponen elektronik dan microcontroller Arduino, instalasi dan konfigurasi software pendukung Arduino, Sejarah Mikrokontroler, Perkembangan Teknologi Mikrokontroler, Implementasi Teknologi Mikrokontroler, Konsep Arduino, Perangkat Lunak Arduino, Arduino Dasar-Dasar Mikrokontroler dan Arduino, Jenis-Jenis Arduino, Konsep Dasar Pemrograman Arduino, Program Sederhana. Penggunaan Sensor-sensor dalam Arduino Seperti: Sensor Cahaya (LDDR), Sensor cahaya (HC-SR04), Sensor DHT11, Sensor IR Proximity, Sensor Sentuh, Sensor Suara, Sensor Warna (TCS34725), Sensor Gyroscope, (MPU-6050), NodeMCU ESP8266.

Pengenalan I2C, RTC sebagai Interface I2C, Interface I2C pada Arduino.



# BAB 1

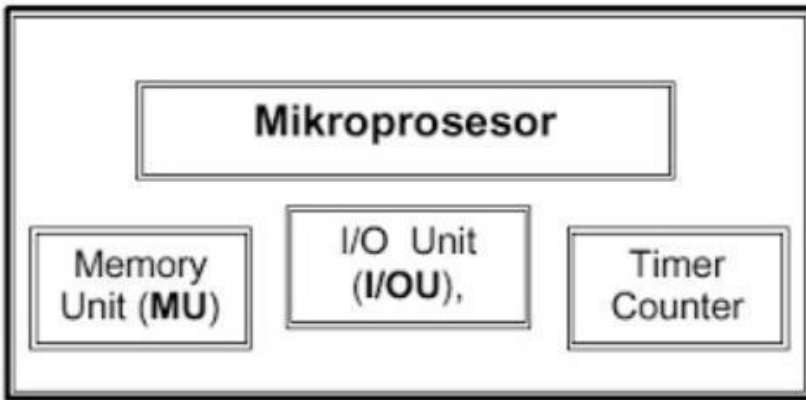
## PENGANTAR TEKNOLOGI MIKROKONTROLER

### 1. Sejarah Mikrokontroler

Sejarah awal perkembangan mikrokontroler dimulai pada awal tahun 1970-an. Motorola mengeluarkan seri mikrokontroler 6800 yang terus dikembangkan menjadi 68HC05, 68HC08, 68HC11, 68HC12, dan 68HC16. Zilog juga mengeluarkan seri mikroprosesor Z80-nya yang terkenal dan terus dikembangkan hingga kini menjadi Z180 dan kemudian diadopsi juga oleh mikroprosesor Rabbit. Tahun 1974, Texas Instrument mengenalkan mikrokontroler dengan seri TMS 1000 yang merupakan mikrokontroler 4 bit. TMS 1000 adalah mikrokomputer dalam sebuah chip, lengkap dengan RAM dan ROM. Tahun 1976, Intel mengeluarkan mikrokontroler 8-bit seri Intel 8748 yang merupakan mikrokontroler dari keluarga MCS 48. Intel juga mengeluarkan mikrokontroler-nya yang populer di dunia yaitu 8051, yang karena begitu populernya maka arsitektur 8051 tersebut kemudian diadopsi oleh vendor lain seperti Phillips, Siemens, Atmel, dan vendor-vendor lain. Selain itu masih ada mikrokontroler populer lainnya seperti Basic Stamps, PIC dari Microchip, MSP 430 dari Texas Instrument dan masih banyak lagi.

Mikrokontroler (*microcontroller*) atau disingkat dengan "*micron*" adalah pengendali yang merupakan suatu komputer kecil yang terletak di dalam sebuah chip atau IC (*integrated circuit*) yang berisikan inti prosesor, memori, dan komponen input/output yang dapat diprogram. Mikrokontroler biasa digunakan pada produk dan perangkat yang dapat dikontrol secara otomatis, seperti sistem kontrol mesin mobil (*engine control*), perangkat medis (*medical devices*), pengendali jarak jauh (*remote control*), mesin perkantoran (*office machines*), dan juga mainan (*games*). Penggunaan

mikrokontroler lebih ekonomis dibandingkan sebuah desain sistem yang berisikan mikroprosesor, memori, dan perangkat input/ouput terpisah. Mikrokontroler adalah komputer mikro dalam satu chip tunggal. Mikrokontroler memadukan CPU, ROM, RWM, I/O paralel, I/O seri, counter-timer, dan rangkaian clock dalam satu chip tunggal seperti terlihat pada Gambar 1.1.



Gambar 1.1. Blok Diagram Mikrokontroller

Sama halnya dengan mikroprosesor, mikrokontroler adalah perangkat yang dirancang untuk kebutuhan umum (*specific purpose*). Sesuai dengan fungsinya sebagai pengendali, mikrokontroler berisikan sepaket chip lengkap yang terdiri dari fitur-fitur pengolah data yang juga terdapat dalam mikroprosesor, ditambah RAM, ROM, I/O, dan fitur lain yang terintegrasi di dalamnya. Contohnya dapat ditemui pada perangkat otomotif, mesin industri, elektronik dan perangkat- perangkat lain yang memiliki embedded sistem di dalamnya. Mikrokontroler sebagai sebuah chip telah mengalami perkembangan baik dari sisi arsitektur, teknologi dan kemampuannya. Untuk melihat perbedaan konsep diantara mikroprosesor dan mikrokontroler di bawah ini ditunjukkan Tabel 1.1 perbandingan konfigurasi, arsitektur, dan set instruksi diantara mikroprosesor Z-80 CPU dengan mikrokontroler AT89C51.

Tabel 1.1. Perbedaan Mikroprosesor dan Mikrokontroler

No	Konfigurasi Pin	Mikroprosesor Z-80 CPU	Mikrokontroler AT89C51
1.	Jumlah pin	40	40
2.	Jumlah pin alamat	16	16
3.	Jumlah pin data	8	8
4.	Jumlah pin interrupt	2	2
5.	Pin I/O	-	32
6.	Register 8 bit	20	34
7.	Register 16 bit	4	2
8.	Ukuran Stack	64 K	128
9.	ROM Internal	-	4 Kbyte
10.	RAM Internal	-	128 byte
11.	Memori eksternal	64 Kbyte	128 Kbyte
12.	Flag	6	4
13.	Timer	0	2
14.	Port Paralel	0	4 x 8 bit
15.	Port Serial	0	1

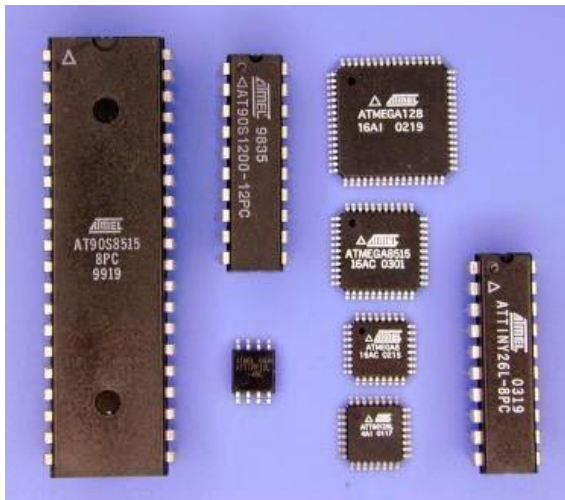
Terdapat perbedaan yang signifikan antara mikrokontroler dan mikroprosesor. Perbedaan yang utama antara keduanya dapat dilihat dari dua faktor utama yaitu arsitektur perangkat keras (*hardware architecture*) dan aplikasi masing-masing.

- 1) Ditinjau dari segi arsitekturnya, mikroprosesor hanya merupakan *single chip* CPU, sedangkan mikrokontroler dalam IC-nya selain CPU juga terdapat device lain yang memungkinkan mikrokontroler berfungsi sebagai suatu *single chip* komputer. Dalam sebuah IC mikrokontroler telah terdapat ROM, RAM, EPROM, serial interface dan paralel interface, timer, interrupt controller, konverter Analog ke Digital, dan lainnya (tergantung feature yang melengkapi mikrokontroler tersebut).
- 2) Sedangkan dari segi aplikasinya, mikroprosesor hanya berfungsi sebagai *Central Processing Unit* yang menjadi otak komputer, sedangkan mikrokontroler, dalam bentuknya yang mungil, pada umumnya ditujukan untuk melakukan tugas-tugas yang berorientasi kendali (*control*) pada rangkaian yang membutuhkan jumlah komponen minimum dan biaya rendah (*low cost*). Berisi tentang apa yang ingin dicapai oleh mahasiswa

setelah mempelajari bab ini. Dapat disajikan dalam bentuk paragraf atau poin (nomor).

## 2. Perkembangan Teknologi Mikrokontroler

Mikrokontroler sebagai teknologi baru yaitu teknologi semikonduktor kehadirannya sangat membantu perkembangan dunia elektronika. Dengan arsitektur yang praktis tetapi memuat banyak kandungan transistor yang terintegrasi, sehingga mendukung dibuatnya rangkaian elektronika yang lebih portable. Mikrokontroler dapat diproduksi secara masal sehingga harganya menjadi lebih murah dibandingkan dengan mikroprosesor, tetapi tetap memiliki kelebihan yang bisa diandalkan. Gambar 1.2 memperlihatkan beberapa contoh chip mikrokontroler.



Gambar 1.2. Chip Mikrokontroler

Mikrokontroler memiliki perbandingan ROM dan RAM-nya yang besar, artinya program kontrol disimpan dalam ROM (bisa *Masked ROM* atau *Flash PEROM*) yang ukurannya relatif lebih besar, sedangkan RAM digunakan sebagai tempat penyimpanan sementara, termasuk register-register yang digunakan pada mikrokontroler yang bersangkutan.

Perkembangan Mikrokontroler mengalami perubahan dari segi rancangan dan aplikasinya, seperti faktor kecepatan pengolahan data yang semakin meningkat (cepat) dibanding pendahulunya. Seperti halnya sebuah mikroprosesor, mikrokontroler juga berkembang sesuai rancangan dan model-model aplikasinya. Mikrokontroler berdasarkan jumlah bit data yang dapat diolah dapat dibedakan dalam :

- 1) Mikrokontroler 4 Bit : merupakan mikrokontroler dengan jumlah bit data terkecil. Mikrokontroler jenis ini diproduksi untuk meminimalkan jumlah pin dan ukuran kemasan.
- 2) Mikrokontroler 8 Bit : merupakan mikrokontroler yang paling banyak digunakan untuk pekerjaan-pekerjaan perhitungan skala kecil. Dalam komunikasi data, Data ASCII serial juga disimpan dalam ukuran 8 bit. Kebanyakan IC memori dan fungsi logika dibangun menggunakan data 8 bit sehingga interface bus data menjadi sangat mudah dibangun. Penggunaan mikrokontroler 8 bit jauh lebih banyak dibandingkan dengan mikrokontroler 4 bit. Aplikasinya juga sangat variatif mulai dari aplikasi kendali sederhana sampai kendali mesin berkecepatan tinggi.
- 3) Mikrokontroler 16 Bit : keterbatasan-keterbatasan yang ada pada mikrokontroler 8 bit berkaitan dengan semakin kompleksnya pengolahan data dan pengendalian serta kecepatan tanggap/respon disempurnakan dengan menggunakan mikrokontroler 16 bit. Salah satu solusinya adalah dengan menaikkan kecepatan clock, dan ukuran data. Mikrokontroler 16 bit digunakan untuk mengatur tangan robot, dan aplikasi *Digital Signal Processing* (DSP).
- 4) Mikrokontroler 32 Bit : ditargetkan untuk aplikasi Robot, Instrumen cerdas, Avionics, Image Processing, Telekomunikasi, Automobil, dan sebagainya. Program-program aplikasinya bekerja dengan sistim operasi dan dipadukan dengan perangkat pintar lainnya.

Karena kebutuhan yang tinggi terhadap “*smart chip*” dengan berbagai fasilitasnya, maka berbagai vendor juga berlomba untuk

menawarkan produk-produk mikrokontrolernya. Selain mikroprosesor dan mikrokontroler, sebenarnya telah bermunculan chip-chip pintar lain seperti DSP prosesor dan *Application Specific Integrated Circuit* (ASIC). Di masa depan, chip-chip mungil berkemampuan sangat tinggi akan mendominasi semua desain elektronik di dunia sehingga mampu memberikan kemampuan komputasi yang tinggi serta meminimumkan jumlah komponen-komponen konvensional.

### 3. Implementasi Teknologi Mikrokontroler

Karena kemampuannya yang tinggi, bentuknya yang kecil, konsumsi dayanya yang rendah, dan harga yang murah maka mikrokontroler begitu banyak digunakan di dunia. Mikrokontroler digunakan mulai dari mainan anak-anak, perangkat elektronik rumah tangga, perangkat pendukung otomotif, peralatan industri, peralatan telekomunikasi, peralatan medis dan kedokteran, sampai dengan pengendali robot serta persenjataan militer. Terdapat beberapa keunggulan yang diharapkan dari alat-alat yang berbasis mikrokontroler (*microcontroller-based solutions*) :

- 1) Keandalan tinggi (*high reliability*) dan kemudahan integrasi dengan komponen lain (*high degree of integration*)
- 2) Ukuran yang semakin dapat diperkecil (*reduced in size*)
- 3) Penggunaan komponen dipersedikit (*reduced component count*) yang juga akan menyebabkan biaya produksi dapat semakin ditekan (*lower manufacturing cost*)
- 4) Waktu pembuatan lebih singkat (*shorter development time*) sehingga lebih cepat pula dijual ke pasar sesuai kebutuhan (*shorter time to market*)
- 5) Konsumsi daya yang rendah (*lower power consumption*)

Penerapan teknologi di masyarakat akan memberikan banyak keuntungan. Mikrokontroler merupakan sebuah sistem komputer fungsional dalam sebuah chip. Di dalamnya terkandung

sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan input output. Dengan penggunaan mikrokontroler ini maka:

- 1) Sistem elektronik akan menjadi lebih ringkas,
- 2) Rancang bangun sistem elektronik akan lebih cepat karena sebagian besar dari sistem adalah perangkat lunak yang mudah dimodifikasi,
- 3) Pencarian gangguan lebih mudah ditelusuri karena sistemnya yang kompak,

Teknologi mikrokontroler dapat diterapkan di berbagai bidang, baik di industri maupun di masyarakat. Dengan menerapkan teknologi Mikrokontroler di masyarakat akan memberikan banyak keuntungan, contohnya :

- 1) Kunci Kombinasi Digital dengan Remote Kontrol Inframerah
- 2) Jam Wekker Digital Bertampilan M1632 LCD
- 3) Alat penyiram tanaman otomatis
- 4) Pemberi pakan ikan otomatis
- 5) Deteksi kebakaran menggunakan sensor suhu dan mikrokontroler
- 6) Pendeteksi kebocoran gas
- 7) Sistem keamanan mobil
- 8) Pintu Air Otomatis
- 9) Pengendali lampu rumah dengan mikrokontroler melalui SMS
- 10) Robot Cerdas

#### 4. Ringkasan

1. Mikrokontroler (*microcontroller*) adalah pengendali yang merupakan suatu komputer kecil yang terletak di dalam sebuah chip atau IC (*integrated circuit*) yang berisikan inti prosesor, memori, dan komponen input/output yang dapat diprogram.
2. Ditinjau dari segi arsitekturnya, mikroprosesor hanya merupakan single chip CPU, sedangkan mikrokontroler dalam

IC-nya selain CPU juga terdapat device lain yang memungkinkan mikrokontroler berfungsi sebagai suatu single chip computer.

3. Mikrokontroler memiliki perbandingan ROM dan RAM-nya yang besar, artinya program kontrol disimpan dalam ROM (bisa Masked ROM atau Flash PEROM) yang ukurannya relatif lebih besar, sedangkan RAM digunakan sebagai tempat penyimpanan sementara, termasuk register-register yang digunakan pada mikrokontroler yang bersangkutan.



# BAB 2

## MIKROKONTROLER PLATFORM ARDUINO

### 1. Konsep Arduino

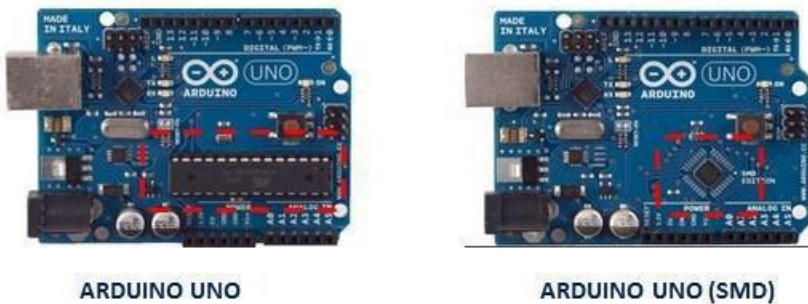
Arduino merupakan papan-tunggal mikrokontroler serba guna yang bisa diprogram dan bersifat *open-source*. Platform Arduino sekarang ini menjadi sangat populer dengan penambahan jumlah pengguna baru yang terus meningkat. Hal ini karena kemudahannya dalam penggunaan dan penulisan kode program. Tidak seperti kebanyakan papan sirkuit pemrograman sebelumnya, Arduino tidak lagi membutuhkan perangkat keras terpisah (disebut *programmer* atau *downloader*) untuk memuat atau meng-*upload* kode baru ke dalam mikrokontroler. Cukup dengan menggunakan kabel USB untuk mulai menggunakan Arduino. Selain itu, Arduino IDE menggunakan bahasa pemrograman C++ dengan versi yang telah disederhanakan, sehingga lebih mudah dalam belajar pemrograman. Arduino akhirnya berhasil menjadi papan sirkuit pemrograman paling disukai hingga menjadikannya sebagai bentuk standar dari fungsi mikrokontroler dengan paket yang mudah untuk diakses.

Hardware dan software Arduino dirancang bagi para seniman, desainer, pe-hobi, hacker, pemula dan siapapun yang tertarik untuk menciptakan objek interaktif dan pengembangan lingkungan. Arduino mampu berinteraksi dengan tombol, LED, motor, speaker, GPS, kamera, internet, ponsel pintar bahkan dengan televisi. Fleksibilitas ini dihasilkan dari kombinasi ketersediaan software Arduino yang gratis, papan perangkat keras yang murah, dan keduanya yang mudah untuk dipelajari. Hal inilah yang menciptakan jumlah pengguna menjadi sebuah komunitas besar dengan berbagai kontribusinya yang telah dirilis pada berbagai proyek dengan berbasiskan Arduino.

Arduino Uno merupakan versi terbaru dari keluarga Arduino, berbasis mikrokontroler ATmega328, menyempurnakan tipe sebelumnya, Duemilanove. Perbedaan Arduino tersebut adalah tidak menggunakan IC FTDI (*Future Technology Devices International*) *USB to Serial* sebagai driver komunikasi USB-nya tetapi menggunakan mikrokontroler ATmega8U2 yang diprogram sebagai konverter USB ke serial. Uno sendiri diambil dari bahasa Italia yang artinya 1 (satu).

Gambar 2.1 adalah *board* Arduino Uno dengan spesifikasi hardware:

- Microcontroller : ATmega328
- Tegangan Operasi : 5 V
- Tegangan Input : 7 - 12V
- Digital I/O : 14 pin
- PWM : 6 channel
- Analog Input : 6 channel
- Memory : 32KB Flash PEROM (0,5 KB digunakan oleh *bootloader*), 2KB SRAM dan 1KB EEPROM
- Frekuensi Clock : 16 MH



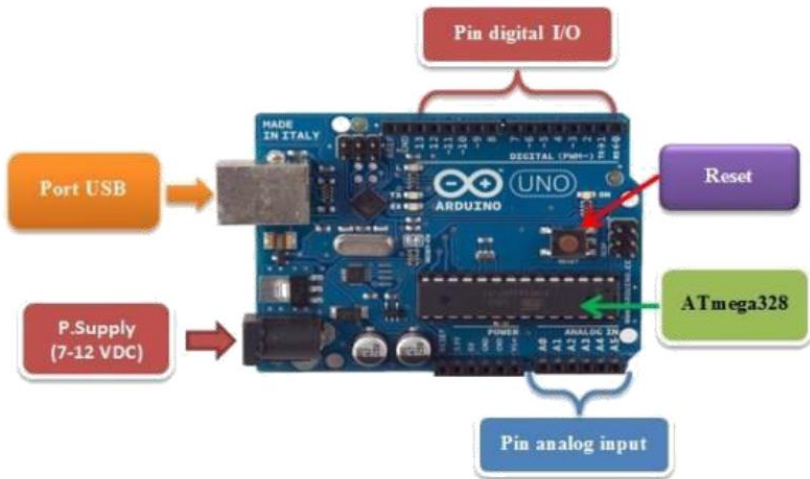
Gambar 2.1. Arduino Uno Board

Tabel 2.1. menunjukkan beberapa produk board yang diproduksi Arduino yang ada di pasaran :

Tabel 2.1 Produk Board Arduino

No	Nama Board	No	Nama Board
1	Arduino Uno	14	LilyPad Arduino
2	Arduino Leonardo	15	Arduino NG (Nuova Generazione)
3	Arduino Due	16	Arduino Extreme
4	Arduino Yun	17	Arduino Diecimila
5	Arduino Tre	18	Arduino Mega
6	Arduino Micro	19	Arduino Mega 2560
7	Arduino Robot	20	Arduino Pro
8	Arduino Esplora	21	Arduino Pro Mini
9	Arduino Mega ADK	22	Arduino Ethernet
10	Arduino Fio	23	Arduino Duemilanove
11	Arduino USB	24	Arduino Shields
12	Arduino Nano	25	Arduino Single Side Serial
13	Arduino Serial	26	Arduino Mini

Pada Gambar 2.2. memperlihatkan bagian utama dari papan Arduino uno, yakni terminal power supply, port USB, pin digital I/O, tombol reset, mikrokontroler ATmega328 dan pin analog input.



Gambar 2.2. Bagian Utama Arduino Uno Board

Berikut ini adalah penjelasan beberapa bagian utama dari papan Arduino uno, yaitu:

#### 1. Power Supply.

Pada Arduino board, ada 2 (dua) pilihan sumber tegangan yang dapat digunakan, yakni dari port USB maupun dari power supply eksternal. Dengan menghubungkan port USB di komputer/laptop dengan Arduino maka secara otomatis *power supply* Arduino bersumber dari port USB. Untuk sumber tegangan eksternal (non-USB) yakni dengan menghubungkan Arduino board dengan sumber tegangan DC. Tegangan yang direkomendasikan adalah 7 sampai 12 V, jika kurang dari 7V akan menyebabkan ketidakstabilan tegangan, sedangkan jika lebih dari 12V akan menyebabkan panas dan akibat fatal berupa kerusakan pada board Arduino.

#### 2. Input - Output.

Port Arduino berbeda penamaannya dengan sistem minimum mikrokontroler atau *microncontroller development system*. Sebagai contoh pada system minimum ATmega8535 penamaan port adalah PORTA, PORTB, PORTC dan PORTD, untuk akses per-bit maka PORTA.0 s/d PORTA.7, contoh lain pada AT89S51 maka PORT0, PORT1 dan seterusnya. Sistem penamaan port pada Arduino merupakan urutan nomor port, mulai dari nomor nol (0), satu (1) dan seterusnya. Untuk digital I/O dengan nama pin 1, 2 sampai 13, sedangkan untuk analog input menggunakan nama A0, A1 sampai A5.

Pada Arduino uno terdapat 14 pin digital input - output. Secara umum berfungsi sebagai port input - output biasa, namun ada beberapa pin yang mempunyai fungsi alternatif. Sebagai contoh, pin 2 dan 3 juga sebagai pin interupsi eksternal. Kemudian pin 5,6,9,10 dan 11 dapat dipakai untuk PWM (*Pulse Width Modulation*) yang sering dipakai untuk kendali motor DC maupun motor servo. Tabel 4.2.menunjukkan nomor dan fungsi pin pada Arduino uno.

Tabel 2.2. Konfigurasi Pin Arduino Uno

No Pin	Fungsi	Fungsi alternatif
0	Digital I/O 0	RX (serial - receiver)
1	Digital I/O 1	TX (serial - transmitter)
2	Digital I/O 2	Interupsi external
3	Digital I/O 3	Interupsi external & PWM
4	Digital I/O 4	-
5	Digital I/O 5	PWM
6	Digital I/O 6	PWM
7	Digital I/O 7	-
8	Digital I/O 8	-
9	Digital I/O 9	PWM
10	Digital I/O 10	SPI - SS & PWM
11	Digital I/O 11	SPI - MOSI & PWM
12	Digital I/O 12	SPI - MISO
13	Digital I/O 13	SPI - SCK & LED

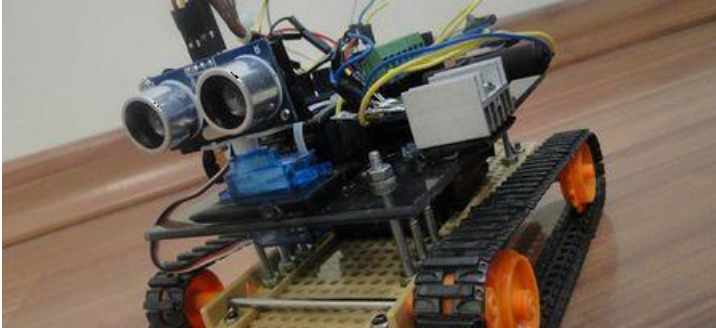
### 3. Analog Input.

Arduino memiliki 6 pin analog input, berfungsi membaca sinyal masukan analog seperti sensor analog. Meskipun demikian pin analog input dapat pula digunakan untuk keperluan digital I/O. Tabel 2.3. menunjukkan nomor dan fungsi pin input analog.

Tabel 2.3. Konfigurasi Pin Analog Input

No pin	Fungsi	Fungsi Alternatif
A0	Analog Input 1	-
A1	Analog Input 2	-
A2	Analog Input 3	-
A3	Analog Input 4	-
A4	Analog Input 5	TWI - SDA
A5	Analog Input 6	TWI - SCL

Arduino memberikan kemudahan bagi penggunanya untuk membuat berbagai proyek berbasis mikrokontroler. Contohnya yang dapat dibuat dengan Arduino antara lain, untuk membuat simulasi lampu, membuat robot, mengontrol motor dc, mengontrol motor stepper, pengatur suhu, display LCD, dan masih banyak yang lainnya. Gambar 2.3 adalah contoh produk alat menggunakan Arduino.



Gambar 2.3 Robot berbasis Arduino Uno

## 2. Perangkat Lunak Arduino

Arduino dapat digunakan untuk mengembangkan objek interaktif, mengambil masukan dari berbagai *switch* atau sensor, dan mengendalikan berbagai lampu, motor, dan output fisik lainnya. Proyek Arduino dapat berdiri sendiri, atau berkomunikasi dengan perangkat lunak (*software*) yang berjalan pada komputer (misalnya: Flash, Pengolahan, MaxMSP, database, dsb). Board dapat dirakit sendiri atau dibeli; open-source IDE dapat didownload secara gratis.

Arduino adalah software open source yang memudahkan untuk menulis kode program dan meng-upload-nya ke board Arduino. Software Arduino dapat berjalan pada Windows, Mac OS X, dan Linux. Software ini ditulis dalam bentuk Java dan berbasis processing, avr-gcc, dan perangkat lunak open source lainnya.

Software Arduino yang ada dalam situs Arduino (<https://www.arduino.cc/>) telah memiliki versi 1.6.6, seperti terlihat pada Gambar 2.4. Software Arduino IDE (*Integrated Development Environment*) adalah sebuah perangkat lunak yang

memudahkan dalam mengembangkan aplikasi mikrokontroler mulai dari menuliskan source program, kompilasi, upload hasil kompilasi, dan uji coba secara terminal serial. Arduino dapat dijalankan di komputer dengan berbagai macam *platform* karena didukung atau berbasis Java. Source program yang dibuat untuk aplikasi mikrokontroler adalah bahasa C/C++ dan dapat digabungkan dengan *assembly*.

Download the Arduino Software



Gambar 2.4. Tampilan Software Arduino 1.6.6

Di samping IDE Arduino sebagai jantungnya, *bootloader* adalah jantung dari Arduino lainnya yang berupa program kecil yang dieksekusi sesaat setelah mikrokontroler diberi catu daya. *Bootloader* ini berfungsi sebagai pemonitor aktifitas yang diinginkan oleh Arduino. Jika dalam IDE terdapat file hasil kompilasi yang akan di-*upload*, *bootloader* secara otomatis menyambutnya untuk disimpan dalam memori program. Jika pada saat awal mikrokontroler bekerja, *bootloader* akan mengeksekusi program aplikasi yang telah diupload sebelumnya. Jika IDE hendak mengupload program baru, *bootloader* seketika menghentikan eksekusi program berganti menerima data program untuk selanjutnya diprogramkan dalam memori program mikrokontroler.

Hubungan komunikasi data antara IDE arduino dengan board Arduino digunakan komunikasi secara serial dengan protokol RS232. Jika board arduino sudah dilengkapi dengan komunikasi serial RS232 (biasanya USB), maka dapat langsung ditancapkan ke USB komputer. Piranti serial RS232 ini digunakan

jika board arduino atau arduino buatan sendiri tidak dilengkapi dengan piranti serial 232.

## Prosedur Menggunakan Arduino Board

### 1. Menyiapkan Arduino Board dan Kabel USB (Gambar 2.5)



### 2. Men-download Software Arduino

Dapatkan versi terbaru dari halaman download yang tersedia di situs resmi Arduino, <https://www.arduino.cc/en/Main/Software>. Setelah download selesai, unzip file yang didownload (jika mendownload file tipe .zip). Pastikan untuk mempertahankan struktur folder. Klik dua kali pada folder untuk membukanya, dan akan ada beberapa file dan sub-folder di dalam.

### 3. Menghubungkan Arduino Board ke Komputer

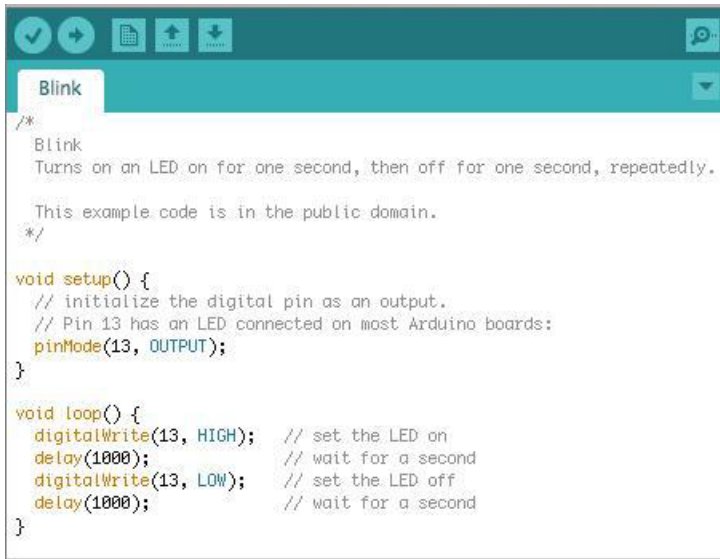
Arduino Uno, Mega Duemilanove dan Arduino Nano memerlukan sumber listrik dari salah satu koneksi USB komputer atau power supply eksternal. Sumber daya dipilih dengan jumper, plastik kecil yang terdapat antara USB dan jack listrik. Periksa apakah jumper diatur pada dua pin paling dekat dengan port USB atau tidak. Hubungkan papan Arduino ke komputer menggunakan kabel USB. LED indikator daya hijau (berlabel PWR) akan menyala.

### 4. Membuka Software Arduino (Gambar 2.6)

Klik dua kali aplikasi Arduino, lalu dapat dilakukan simulasi dengan hal-hal sebagai berikut

- Buka contoh blink
- Buka LED berkedip contoh sketsa: File > Examples > 1.Basics > Blink.





```
/*
  Blink
  Turns on an LED on for one second, then off for one second, repeatedly.

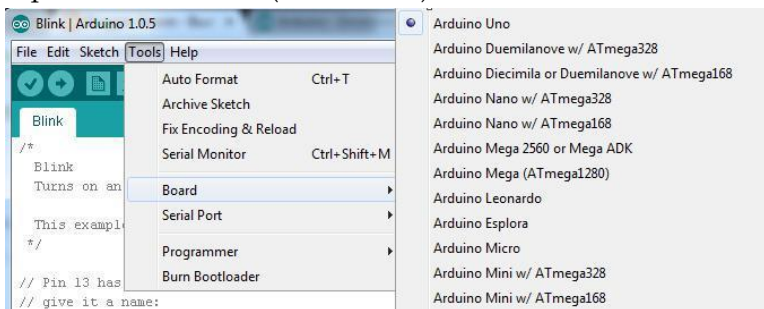
  This example code is in the public domain.
  */

void setup() {
  // initialize the digital pin as an output.
  // Pin 13 has an LED connected on most Arduino boards:
  pinMode(13, OUTPUT);
}

void loop() {
  digitalWrite(13, HIGH); // set the LED on
  delay(1000);           // wait for a second
  digitalWrite(13, LOW); // set the LED off
  delay(1000);           // wait for a second
}
```

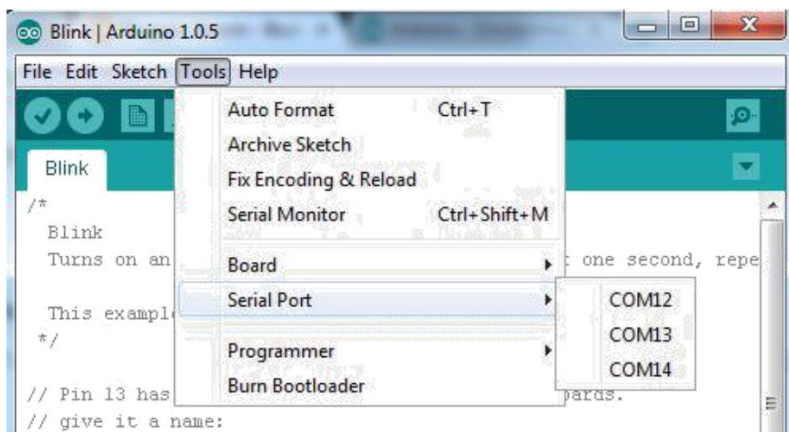
Gambar 2.6. Contoh Blink LED

- Pilih Board yang digunakan dengan memilih entri dalam menu Tools > Board yang sesuai dengan Arduino. Misalnya dipilih Arduino Uno (Gambar 2.7).



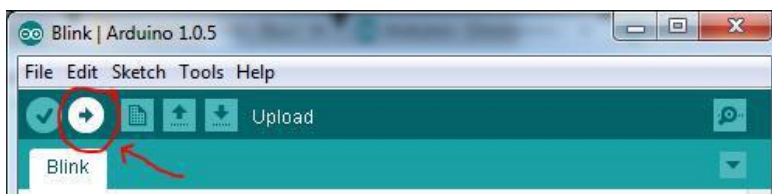
Gambar 2.7. Memilih Board

- Pilih port serial (Gambar 2.8)  
Pilih perangkat serial Arduino dari menu Tools | Serial Port. Biasanya Port akan otomatis terdeteksi.



Gambar 2.8. Memilih Port

- Upload Program (Gambar 2.9)  
 Dengan meng-klik tombol "Upload". Tunggu beberapa detik - akan terlihat akan berkedip LED RX dan TX pada Arduino Board.



Gambar 2.9. Upload Program

### 3. Ringkasan

1. Arduino merupakan papan-tunggal mikrokontroler serba guna yang bisa diprogram dan bersifat *open-source*.
2. Arduino Uno adalah salah satu kit mikrokontroler yang berbasis pada ATmega328. Modul ini sudah dilengkapi dengan berbagai hal yang dibutuhkan untuk mendukung mikrokontroler untuk bekerja.
3. Arduino IDE menggunakan bahasa pemrograman C++ dengan versi yang telah disederhanakan, sehingga lebih mudah dalam belajar pemrograman.

4. Software Arduino IDE (*Integrated Development Environment*) adalah sebuah perangkat lunak yang memudahkan dalam mengembangkan aplikasi mikrokontroler mulai dari menuliskan source program, kompilasi, upload hasil kompilasi, dan uji coba secara terminal serial.
5. Hubungan komunikasi data antara IDE arduino dengan board Arduino digunakan komunikasi secara serial dengan protokol RS232. Jika board arduino sudah dilengkapi dengan komunikasi serial RS232 (biasanya USB), maka dapat langsung ditancapkan ke USB komputer.



# BAB 3

## PEMROGRAMAN ARDUINO

### 1. Arduino Dasar-Dasar Mikrokontroler dan Arduino

Sebelum membahas lebih detail tentang Arduino, marilah kita sama-sama pahami tentang dasar-dasar mikrokontroler. Menurut pengertian atau istilah yang sering digunakan, mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol rangkaian elektronik yang terdiri dari CPU (*Central Processing Unit*), memori, I/O (*input/output*), bahkan sudah dilengkapi dengan ADC (*Analog-to-Digital Converter*) yang sudah terintegrasi di dalamnya.

Kelebihan utama dari mikrokontroler adalah tersedianya RAM (*Random Access Memory*) dan peralatan I/O pendukung sehingga ukuran board mikrokontroler menjadi sangat ringkas.

Mikrokontroler pertama kali dikenalkan oleh Texas Instrument pada tahun 1974 dengan seri TMS-1000. Mikrokontroler pertama ini merupakan mikrokontroler 4 bit. Mikrokontroler ini memiliki sebuah chip yang telah dilengkapi dengan RAM dan ROM (*Read Only Memory*). Selanjutnya, pada tahun 1976 Intel mengeluarkan mikrokontroler 8 bit dengan nama 8748 yang merupakan mikrokontroler dari keluarga MCS-48. Untuk saat ini telah pasaran banyak ditemui tipe-tipe mikrokontroler mulai dari 8 bit sampai dengan 64 bit. Masing-masing fabrikasi (vendor) mengeluarkan mikrokontroler yang telah dilengkapi dengan berbagai fasilitas sehingga memudahkan pengguna (*user*) untuk merancang sebuah sistem dengan komponen luar yang relatif lebih sedikit.

Saat ini mikrokontroler yang banyak beredar dipasaran adalah mikrokontroler 8 bit varian keluarga MCS51 dengan tipe CISC, yaitu *Complex Instruction Set Computing* yang dikeluarkan oleh Atmel dengan seri AT89Sxx, dan mikrokontroler AVR yang

merupakan mikrokontroler RISC (*Reduced Instruction Set Computer*) dengan seri ATMegaxx. Dengan mikrokontroler tersebut pengguna sudah bisa membuat sebuah sistem untuk keperluan sehari-hari, seperti pengendali peralatan rumah tangga jarak jauh yang menggunakan remote control televisi (*smart home*), jam digital, text berjalan, termometer digital, dan lain-lain. Dalam perkembangannya, modul atau minimum sistem dari mikrokontroler dibuat dalam bentuk chip yang lebih memudahkan pengguna untuk menggunakannya. Satu hal yang saat ini sedang atau banyak digemari oleh pengguna mikrokontroler adalah modul Arduino.

Ketika kita membahas tentang Arduino, maka timbul sebuah pertanyaan tentang apakah Arduino sama dengan mikrokontroler?. Mungkin diantara kita masih banyak yang akan mengatakan, “ya”.

Akan tetapi, apabila kita merujuk pada pemahaman yang benar, maka Arduino tidak tepat jika dikatakan sebagai mikrokontroler. Secara detail, perbedaan dari keduanya dapat kita peroleh dari situs resmi Arduino ([www.arduino.cc](http://www.arduino.cc)) sebagai berikut:

*“Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists and anyone interested in creating interactive objects or environments”*

Apabila kita coba memahami makna dari pengertian di atas, maka Arduino dapat dikatakan sebagai ***prototyping platform***. Dalam Bahasa Indonesia, *prototype* dapat diartikan sebagai purwarupa, yaitu suatu alat yang dapat digunakan untuk menghasilkan berbagai karya cipta dalam tahapan desain. Pada kenyataannya, Arduino tidak hanya digunakan pada tahapan desain, namun sampai produk jadi. Kita dapat berkreasi apapun dengan menggunakan Arduino, seperti aplikasi dalam bidang robotika, atau aplikasi-aplikasi *embedded system* lainnya. Arduino

memberikan banyak kemudahan bagi pengguna untuk merealisasikan karya-karyanya.

Arduino telah dilengkapi dengan sistem IDE (*Intergrated Development Environment*) untuk menuliskan program aplikasi yang kita buat. Selain itu, pada board Arduino juga telah dilengkapi dengan berbagai fasilitas sehingga lebih memudahkan para pencinta atau penggunaannya. Dari pemaparan di atas, maka dapat kita perjelas bahwa Arduino adalah *platform prototyping open-source hardware* yang dapat digunakan untuk membuat proyek berbasis pemrograman.

*Hardware* Arduino memiliki prosesor mikrokontroler ATmega yang dirilis oleh Atmel AVR, tetapi *software* yang digunakan memiliki bahasa pemrograman tersendiri. Arduino dirancang untuk memudahkan pengguna elektronik atau siapapun yang ingin mengembangkan peralatan elektronik interaktif berdasarkan *hardware* dan *software* yang fleksibel dan mudah untuk digunakan.

Sampai di sini, kita sudah harus memahami perbedaan antara mikrokontroler dengan Arduino. Mari kita sama-sama simpulkan, Arduino adalah papan elektronik yang menggunakan mikrokontroler jenis tertentu. Sebagai contoh adalah jenis Arduino yang paling banyak digunakan, yaitu Arduino UNO. Jenis ini menggunakan mikrokontroler keluaran Atmel, yaitu seri ATmega328.

Dalam berbagai aplikasi, Arduino dapat digunakan untuk mendeteksi lingkungan dengan menerima *input* dari berbagai sensor atau tombol (sensor cahaya, suhu, inframerah, ultrasonik, jarak, tekanan, kelembaban) dan dapat mengontrol perangkat lainnya seperti mengontrol kecepatan dan arah putar motor, menyalakan LED, dan sebagainya. Keuntungan yang kita dapatkan ketika menggunakan Arduino, antara lain:

- Harga relatif murah dibandingkan dengan mikrokontroler lainnya dengan kelebihan yang ditawarkan.
- Dapat digunakan pada berbagai sistem operasi Windows, Linux, Max, dan lain-lain.

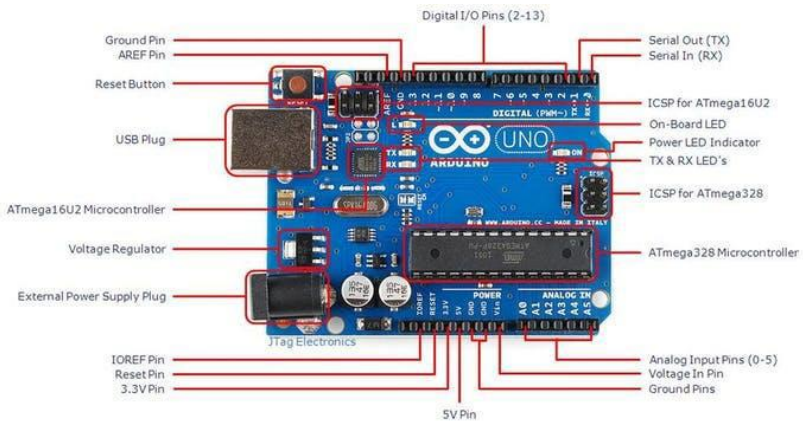
- Memiliki bahasa pemrograman yang mudah dipahami, projek Arduino sudah banyak dipelajari karena *open source*.

## 2. Jenis-Jenis Arduino

Banyak jenis Arduino mulai dari yang paling murah, mudah dicari, dan yang paling banyak digunakan seperti Arduino Uno atau Nano, hingga Arduino yang berbentuk mini PC. Karena sifatnya yang *open source*, maka banyak sekali vendor yang membuat dan menjual produk-produk berbasis Arduino. Berikut ini adalah jenis jenis Arduino yang banyak beredar dipasaran.

### 1. Arduino Uno

Arduino Uno merupakan salah satu Arduino yang murah, mudah didapat, dan sering digunakan. Arduino Uno ini dibekali dengan mikrokontroler ATMEGA328P dan versi terakhir yang dibuat adalah versi R3. Modul ini sudah dilengkapi dengan berbagai hal yang dibutuhkan untuk mendukung mikrokontroler untuk bekerja. ATmega328P yang sudah terbentuk modul Arduino uno seperti terlihat pada gambar di bawah ini.

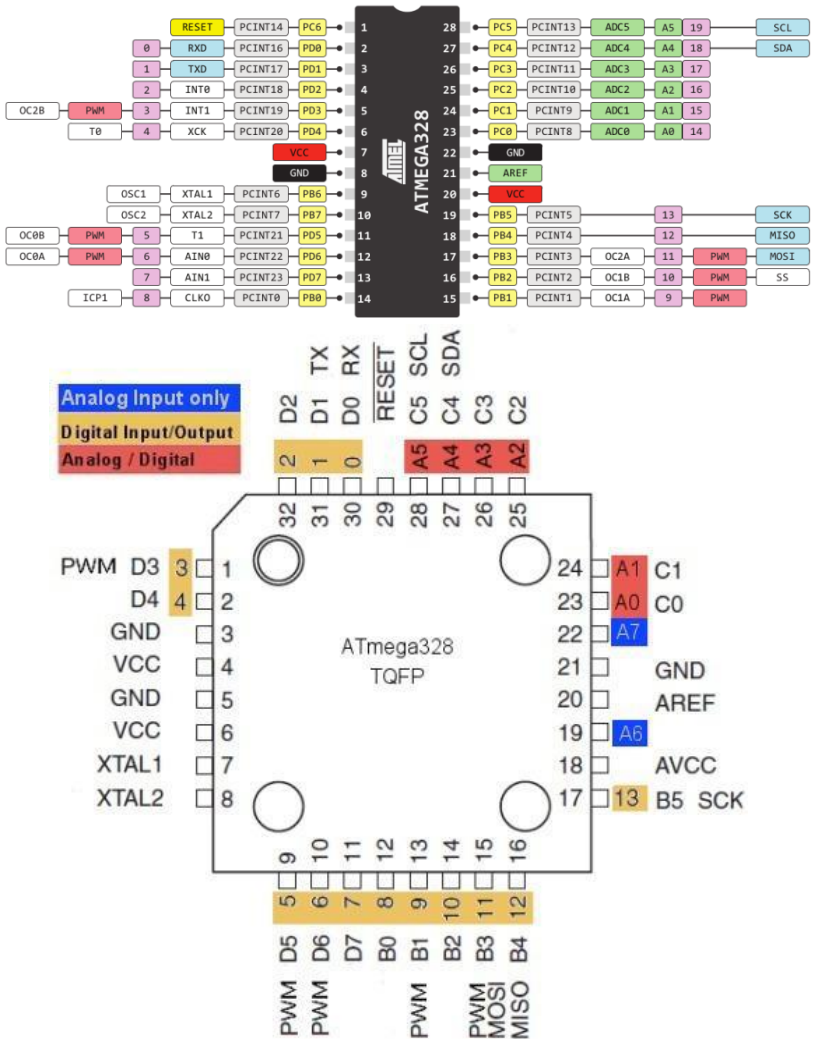


**Gambar 3.1** Arduino Uno R3

ATmega328 adalah mikrokontroler keluaran dari atmel yang mempunyai arsitektur RISC (*Reduce Instruction Set*



Computer) dimana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (Completed Instruction Set Computer).



Gambar 3.2 (a) Pinout ATmega328 model DIP dan (b) Pinout ATmega328 model SMD

ATmega328 memiliki beberapa fitur antara lain:

- Memiliki 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus clock.
- Memiliki 32 x 8-bit register serba guna.

- c. Kecepatan akses mencapai 16 MIPS dengan clock 16 MHz.
- d. Memiliki 32 KB Flash memory dan pada Arduino memiliki bootloader yang menggunakan 2 KB dari flash memori sebagai bootloader.
- e. Memiliki EEPROM (*Electrically Erasable Programmable Read Only Memory*) sebesar 1KB sebagai tempat penyimpanan data semi permanen karena EEPROM tetap dapat menyimpan data meskipun catu daya dimatikan.
- f. Memiliki SRAM (*Static Random Access Memory*) sebesar 2 KB.
- g. Memiliki pin I/O digital sebanyak 14 pin, 6 pin diantaranya dapat digunakan sebagai pin PWM (*Pulse Width Modulation*).
- h. Memiliki Master/Slave SPI Serial interface.

Mikrokontroler ATmega328 memiliki arsitektur Harvard, yaitu pemisah antara memori untuk kode program dan memori untuk data sehingga dapat memaksimalkan kerja dari mikrokontroler. Instruksi - instruksi dalam memori program dieksekusi dalam satu alur tunggal, dimana pada saat satu instruksi dikerjakan instruksi berikutnya sudah diambil dari memori program. Konsep inilah yang memungkinkan instruksi - instruksi dapat dieksekusi dalam setiap satu siklus clock. Sebanyak 32 x 8-bit register serba guna digunakan untuk mendukung operasi pada ALU (*Arithmetic Logic Unit*) yang dapat dilakukan dalam satu siklus. Sebanyak 6 dari register serbaguna ini dapat digunakan sebagai 3 buah register pointer 16-bit pada mode pengalamatan tidak langsung untuk mengambil data pada ruang memori data. Ketiga register pointer 16-bit ini disebut dengan register X (gabungan R26 dan R27), register Y (gabungan R28 dan R29), dan register Z (gabungan R30 dan R31). Hampir semua instruksi AVR memiliki format 16-bit. Setiap alamat memori program terdiri dari instruksi 16-bit atau 32-bit.

Selain register serba guna di atas, terdapat register lain yang terpetakan dengan teknik *memory mapped* I/O berukuran 64 byte. Beberapa register ini digunakan untuk fungsi khusus antara

lain sebagai register control Timer/Counter, Interupsi, ADC, USART, SPI, EEPROM, dan fungsi I/O lainnya. Register - register ini menempati memori pada alamat 0x20h - 0x5Fh. Arduino Uno R3 memiliki 14 pin digital I/O (dimana 6 pin dapat digunakan sebagai *Output* PWM), 6 pin analog *input*, 2x3 pin ICSP (untuk memprogram Arduino dengan software lain), dan kabel USB. Untuk menghidupkannya cukup dengan menghubungkan kabel USB ke komputer atau menggunakan adaptor 5 VDC. Arduino ini sangat disarankan untuk untuk pemula yang ingin belajar Arduino.

Berikut ini adalah spesifikasi untuk Arduino Uno R3 yang dikutip dari <https://store.Arduino.cc/usa/Arduino-uno-rev3>.

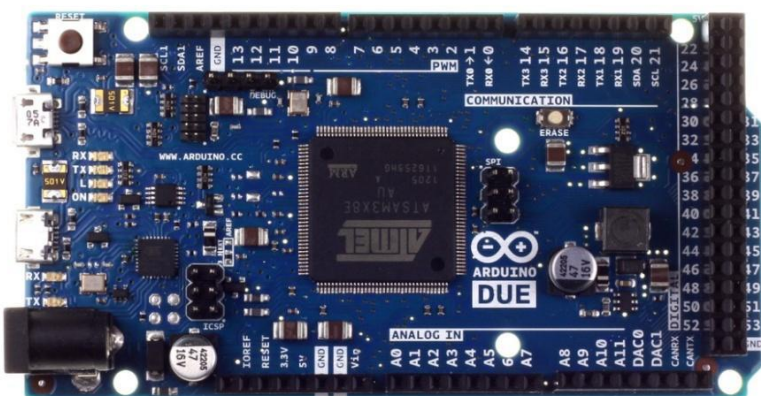
Parameter	Spesifikasi
Tegangan Operasi	5V
Tegangan Input (Rekomendasi )	7-12V
Tegangan Input (Limit )	6-20V
Pin Digital I/O	14 (6 PinOutput PWM)
Pin Digital PWM I/O	6
Pin Analog Input	6
Arus DC tiap Pin I/O	20 mA
Arus DC Pin 3.3V	50 mA
Flash Memory	32 KB (ATmega328P) dimana 0.5 KB digunakan untuk bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Panjang	68,6 mm
Lebar	53,4 mm
Berat	25 g

Arduino Uno memiliki kelebihan-kelebihan yang membuat tipe Arduino ini menjadi lebih mudah dan menyenangkan, antara lain:

- a. Pengembangan *project* mikrokontroler akan menjadi lebih mudah dan menyenangkan. Pengguna dapat langsung menghubungkan board Arduino ke komputer atau laptop melalui kabel USB. Board Arduino juga tidak membutuhkan downloader untuk mendownloadkan program yang telah dibuat dari computer ke mikrokontrolernya.
- b. Didukung oleh Arduino IDE dengan bahasa pemrograman dengan library yang lengkap.
- c. Terdapat modul yang siap pakai/shield sehingga dapat langsung dipasang pada board Arduino.

## 2. Arduino Due

Arduino Due adalah pengembangan dari mikrokontroler Arduino yang menggunakan CPU Atmel SAM3X8E ARM Cortex-M3. Arduino Due adalah Arduino yang pertama kali menggunakan prosesor ARM 32-bit Cortex-M3 keluaran Atmel. Dengan demikian, Arduino Due adalah Arduino Development Board pertama yang dibuat dengan menggunakan mikrokontroler ARM 32-bit.



Gambar 3.3 Arduino Due

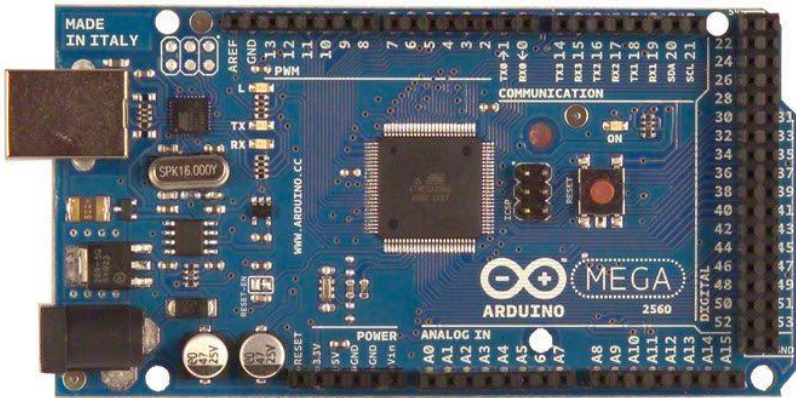
Arduino ini memiliki 54 pin I/O digital (dimana 12 pin dapat digunakan sebagai *output* PWM), 12 pin analog *Input*, 4 pin UART (pin serial untuk komunikasi serial asinkron TX/RX), dan dilengkapi dengan Micro USB yang umumnya terdapat pada handphone saat ini.

Kelebihan utama Arduino Due dibanding varian Arduino lainnya terletak pada penggunaan CPU ARM Cortex-M3 yang memiliki fitur sebagai berikut.

- a. Memiliki core processor 32-bit yang memungkinkan operasi data sebanyak 4 byte sekaligus dengan tipe data DWORD pada satu siklus waktu.
- b. CPU dilengkapi dengan frekuensi clock sebesar 84 MHz.
- c. RAM statis (SRAM) sebesar 96 KB (48 kali lipat lebih besar dibanding Arduino Uno, 12 kali lipat lebih besar dibanding Arduino Mega 2560).
- d. Ruang untuk kode program (*Flash Memory*) sebesar 512 KB (16 kali lipat lebih besar dibanding Arduino Uno, 2 kali lipat lebih besar dibanding Arduino Mega 2560).
- e. Terdapat pengendali akses memori langsung (*Direct Memory Access* atau DMA controller) yang dapat membebaskan CPU dari operasi memori yang intensif.
- f. Terdapat dua kanal DAC (Digital-Analog-Converter) terpadu (Arduino lainnya memiliki ADC tapi tidak memiliki DAC yang merupakan komplemen fungsi dari ADC - ADC mengubah sinyal analog menjadi digital, DAC mengubah sinyal digital menjadi sinyal analog yang sesungguhnya / true analog. Bedakan dengan PWM yang men-simulasi-kan keluaran analog).
- g. Resolusi ADC yang lebih presisi dengan 12-bit ( $2^{12} = 4096$  jenjang, 0-4095) sebanyak 12 kanal.
- h. Sumber catu daya dapat diambil dari port USB atau catu daya eksternal antara 6 ~ 16 Volt DC (direkomendasikan antara 7 ~ 12 VDC).

### 3. Arduino Mega

Arduino Mega umumnya dibuat menggunakan jenis mikrokontroler ATmega 2560. Sesuai dengan namanya, Arduino ini dibekali dengan prosesor ATmega2560 yang memiliki 54 pin digital I/O (dimana 15 pin dapat digunakan sebagai output PWM), 16 pin analog input, 4 pin UART, 2x3 pin ICSP (untuk memprogram Arduino dengan software lain), dan kabel USB komputer yang sekaligus digunakan sebagai sumber tegangan.



Gambar 3.4 Arduino Mega 2560

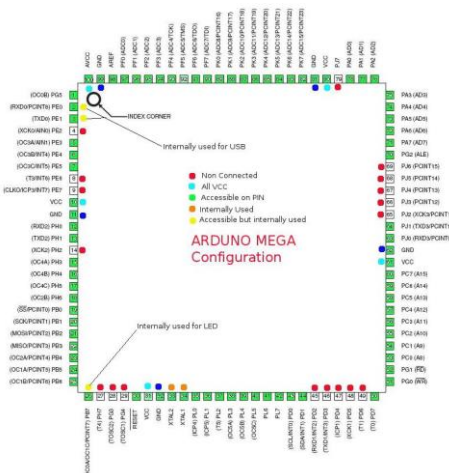
Spesifikasi dari Arduino mega 2560 secara lebih lengkap seperti diuraikan sebagai berikut:

Parameter	Spesifikasi
Tegangan Operasional	5V
Tegangan Input	(rekomendasi) 7-12V
Tegangan Input (limit)	6-20V
Pin Digital I/O	54 (15 PWM)
Pin Analog Input	16
Arus DC per Pin I/O	20 mA
Arus DC untuk Pin 3.3 V	50 mA
Memori Flash	256 KB (8 KB: bootloader)
SRAM	8 KB
EEPROM	4 KB

Clock Speed	16 MHz
LED_BUILTIN	13
Panjang	101,52 mm
Lebar	53,3 mm
Berat	37 g

Arduino Mega 2560 dilengkapi dengan 54 Pin digital yang dapat di gunakan sebagai input atau output dan 16 Pin Analog berlabel A0 sampai A15 sebagai ADC. Setiap Pin Analog memiliki resolusi sebesar 10 bit. Arduino Mega 2560 juga dilengkapi dengan fitur yang memiliki fungsi khusus, sebagai berikut:

- Memiliki 4 buah masukan serial, yaitu Port Serial 0: Pin 0 (RX) dan Pin 1 (TX); Port Serial 1: Pin 19 (RX) dan Pin 18 (TX); Port Serial 2: Pin 17 (RX) dan Pin 16 (TX); Port Serial 3: Pin 15 (RX) dan Pin 14 (TX). Pin Rx digunakan untuk menerima data serial TTL dan Pin (Tx) untuk mengirim data serial TTL.
- Memiliki external Interrupts sebanyak 6 buah: Pin 2 (Interrupt 0), Pin 3 (Interrupt 1), Pin 18 (Interrupt 5), Pin 19 (Interrupt 4), Pin 20 (Interrupt 3), dan Pin 21 (Interrupt 2).



Gambar 3.5 Pinout ATMega2560

- c. Memiliki 15 buah PWM, yaitu pada pin: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, dan 44, 45, dan 46. Pin-pin tersebut dapat digunakan sebagai output PWM berukuran 8 bit
- d. Pin I2C: Pin 20 (SDA) dan Pin 21 (SCL), Komunikasi I2C menggunakan wire library.
- e. Pin SPI: Pin 50 (MISO), Pin 51 (MOSI), Pin 52 (SCK), Pin 53 (SS), digunakan untuk komunikasi SPI menggunakan SPI Library.

#### 4. Arduino Leonardo

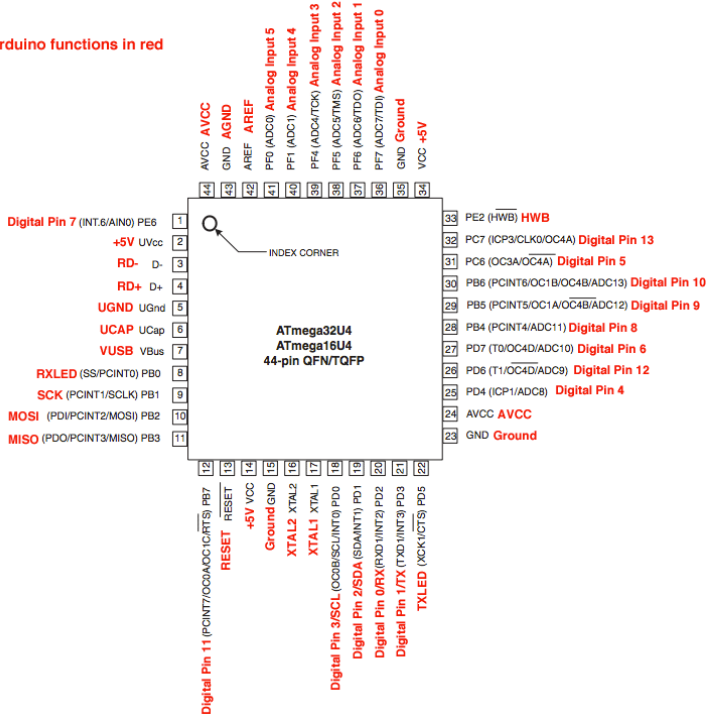
Arduino Leonardo adalah jenis Arduino yang dibuat menggunakan mikrokontroler ATmega32u4. Arduino Leonardo memiliki 20 pin digital I/O (dimana 7 pin dapat digunakan sebagai Output PWM), 12 pin analog Input, 2x3 pin ICSP, dan juga dibekali dengan Mikro USB seperti Arduino Due. Dilihat dari bentuk dan spesifikasinya, Arduino Leonardo memiliki kemiripan dengan Arduino Uno.

Perbedaannya hanya terletak pada penggunaan micro USB untuk pemrograman sekaligus sumber tegangan. Arduino Leonardo berbeda dari semua papan Arduino yang lainnya karena ATmega32u4 secara terintegrasi (*built-in*) telah memiliki komunikasi USB, sehingga tidak lagi membutuhkan prosesor sekunder (tanpa chip ATmega16U2 sebagai konverter USB-toserial).

Hal ini memungkinkan Arduino Leonardo yang terhubung ke komputer digunakan sebagai mouse dan keyboard, selain bisa digunakan sebagai virtual (CDC) serial/COM port. Mikrokontroler ATmega32u4 merupakan microchip 8-bit AVR RISC berkapasitas rendah yang menampilkan memori program flash berkapasitas 32 KB (self-programming), memiliki SRAM 2,5 KB, 1 KB EEPROM, perangkat kecepatan berkecepatan tinggi USB 2.0/kecepatan rendah, 12-channel 10-bit ADC, dan antarmuka JTAG. Perangkat ini mencapai hingga 16 throughput MIPS pada 16 MHz dan tegangan operasi sekitar 2,7 - 5,5 volt. Pinout dari ATmega32u4 seperti tampak pada **Gambar 3.6**.



Arduino functions in red



Gambar 3.6 Pinout ATmega32u4.



Gambar 3.7 Arduino Leonardo

Pin tegangan yang tersedia pada papan Arduino Leonardo adalah sebagai berikut:

- a. Tegangan VIN: Adalah input tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal (sebagai „saingan" tegangan 5 volt dari koneksi USB atau sumber daya terregulator lainnya). Anda dapat memberikan tegangan melalui pin ini, atau jika memasok tegangan untuk papan melalui jack power, kita bisa mengakses/ mengambil tegangan melalui pin ini.
- b. Tegangan 5V: Tegangan listrik ter-regulator yang digunakan untuk daya mikrokontroler dan komponen lainnya pada papan Arduino. Tegangan dapat menggunakan pin VIN melalui regulator on-board, atau dipasok oleh USB atau power suplai lain dengan besar tegangan 5V ter-regulator.
- c. Tegangan 3V3: Sebuah pin yang menghasilkan tegangan 3,3 Volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (on-board). Arus maksimum yang dihasilkan adalah 50 mA.
- d. Pin GND: Pin Ground atau Massa.
- e. Pin IOREF: Pin ini pada papan Arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler (atau VCC untuk papan). Pin ini bertegangan V pada Leonardo.

20 pin digital I/O pada Leonardo dapat digunakan sebagai input atau output, menggunakan fungsi `pinMode()`, `digitalWrite()`, dan `digitalRead()`. Mereka beroperasi pada tegangan 5 volt. Setiap pin dapat memberikan atau menerima maksimum 40 mA dan memiliki resistor pull-up internal sebesar 20-50 kOhm yang terputus secara default. Selain itu, beberapa pin memiliki fungsi khusus, yaitu:

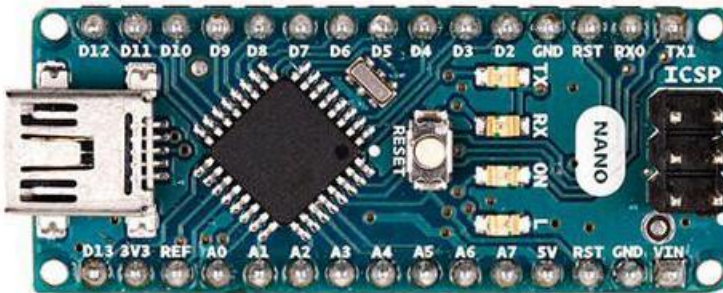
- a. Serial: Pin 0 (RX) dan pin 1 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data serial TTL menggunakan hardware ATmega32U4 yang memiliki kemampuan serial didalamnya. Perhatikan bahwa pada Leonardo, kelas Serial

- mengacu pada komunikasi USB (CDC); untuk TTL serial pada pin 0 dan 1, menggunakan kelas Serial 1.
- b. TWI: Pin 2 (SDA) dan pin 3 (SCL). Dukungan komunikasi TWI menggunakan sistem Wire.
  - c. Eksternal Interupsi: Pin 3 (interrupt 0), pin 2 (interrupt 1), pin 0 (interrupt 2), pin 1 (interrupt 3) dan pin 7 (interrupt 4). Pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau merubah nilai.
  - d. Pin PWM: Pin 3, 5, 6, 9, 10, 11, dan 13 berukuran 8-bit dengan fungsi analogWrite().
  - e. Pin SPI: Pin pada header ICSP ini mendukung komunikasi SPI menggunakan sistem SPI. Pada tipe Arduino ini pin SPI tidak terhubung ke satu pun pin digital I/O karena yang terhubung langsung hanya pada Arduino Uno yang hanya menyediakan konektor ICSP. Ini berarti bahwa jika kita memiliki shield yang menggunakan SPI, tetapi tidak terdapat 6 pin konektor ICSP yang terhubung ke 6 pin ICSP header Arduino Leonardo, maka shield tidak akan bekerja.
  - f. LED: Pin 13. Tersedia secara built-in pada papan Arduino ATmega2560. LED terhubung ke pin digital 13. Ketika pin diset bernilai HIGH, maka LED menyala (ON), dan ketika pin diset bernilai LOW, maka LED padam (OFF).
  - g. Input Analog: Pin A0-A5, Pin A6 - A11 (pada pin digital 4, 6, 8, 9, 10, dan 12). Arduino Leonardo memiliki 12 input analog, berlabel A0 sampai A11, yang semuanya juga dapat digunakan sebagai digital I/O. Pin A0-A5 terdapat di lokasi yang sama seperti pada Arduino Uno; Pin input A6-A11 masing-masing ada pada digital I/O pin 4, 6, 8, 9, 10, dan 12. Masing-masing pin menyediakan resolusi 10 bit (yaitu 1024 nilai yang berbeda). Secara default pin ini dapat diukur/diatur dari mulai 0-5 volt, dan memungkinkan untuk mengubah titik jangkauan tertinggi atau terendah dengan menggunakan pin AREF dan fungsi analogReference().

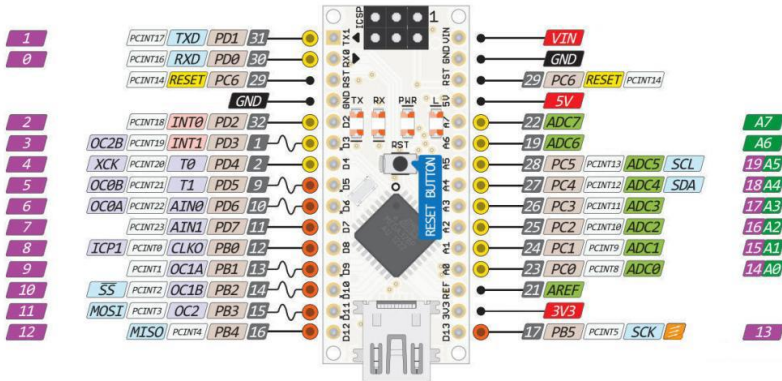
- h. AREF: Referensi tegangan untuk input analog. Digunakan dengan fungsi `analogReference()`.
- i. RESET: Jalur LOW ini digunakan untuk me-reset (menghidupkan ulang) mikrokontroler. Jalur ini biasanya digunakan untuk menambahkan tombol reset pada shield yang menghalangi papan utama Arduino. *duino | 19*

## 5. Arduino Nano

Sesuai dengan namanya, Arduino Nano memiliki ukuran yang relatif kecil dan sangat sederhana. Dengan ukurannya yang kecil, bukan berarti jenis Arduino ini tidak mampu menyimpan banyak fasilitas. Hampir sama dengan tipe Arduino UNO, Arduino Nano dibekali dengan prosesor ATmega328P dengan bentuk SMD dan memiliki 14 Pin Digital I/O, 8 Pin Analog Input (lebih banyak dari Uno), dan menggunakan FTDI untuk pemrograman lewat Mikro USB. Selain itu juga ada yang menggunakan prosesor ATmega168.



Gambar 3.8 Arduino Nano



Gambar 3.9 Pinout Arduino Nano

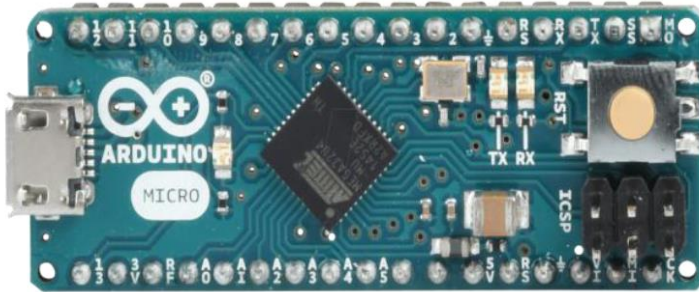
Spesifikasi dari Arduino nano adalah sebagai berikut:

Parameter	Spesifikasi
Tegangan operasi	5V
Tegangan input (rekomendasi)	7V - 12V
Digital I/O pin	14 buah, 6 diantaranya menyediakan PWM
Pin Analog Input	8 buah
Arus DC per pin I/O	40 mA
Memori Flash	32 KB, 0,5 KB telah digunakan untuk bootloader
SRAM	2 KB
EEPROM	1 KB
Clock speed	16 Mhz
Dimensi	45 mm x 18 mm
Berat	5 g

## 6. Arduino Micro

Arduino Micro adalah suatu jenis Arduino yang dibekali dengan mikrokontroler ATmega32U4. Ukurannya lebih panjang dari Arduino Nano dan Arduino Pro Mini, karena memang fasilitasnya lebih banyak yaitu dengan 20 pin digital I/O (7pin

bisa digunakan untuk *Output* PWM), 12 pin analog *Input*, dan Mikro USB.

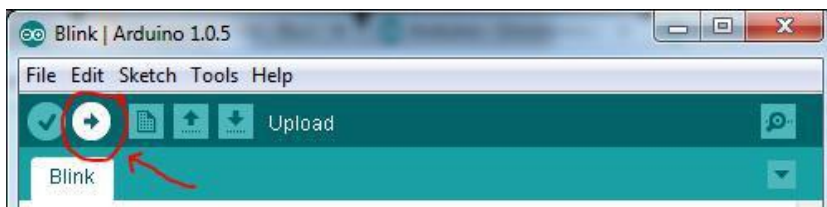


**Gambar 3.10** Arduino Micro

Komunikasi serial pada Arduino Micro dilakukan dengan menggunakan Pin 0 (RX) dan pin 1 (TX) untuk menerima (RX) dan mengirimkan (TX) data serial TTL menggunakan hardware ATmega32U4. Perhatikan bahwa pada Arduino Micro, pengiriman ata Serial mengacu pada komunikasi USB (CDC); untuk TTL serial pada pin 0 dan 1, menggunakan kelas Serial 1.

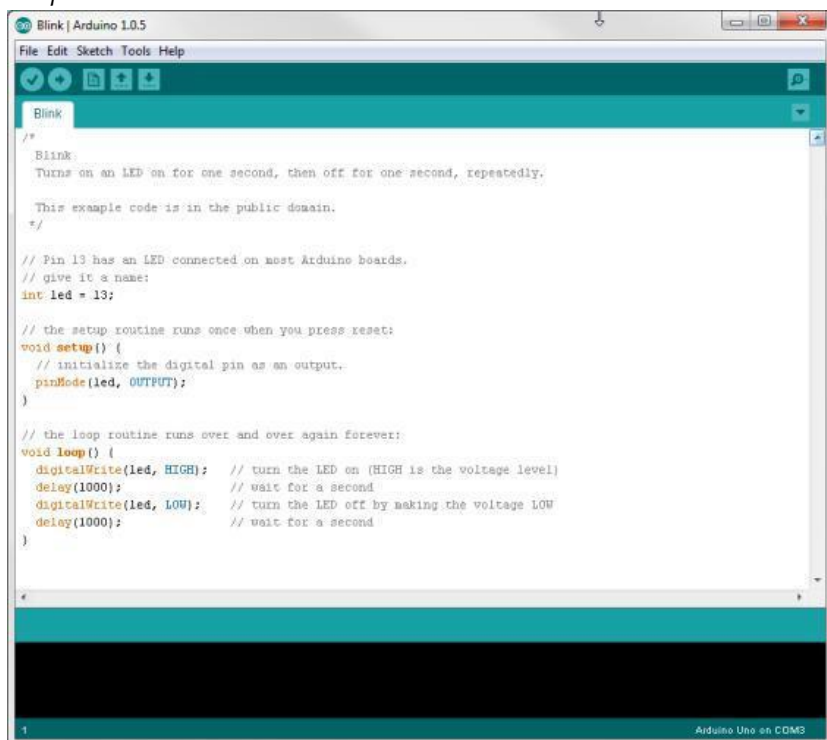
### **3. Konsep Dasar Pemrograman Arduino**

Software/program yang ditulis menggunakan Arduino disebut sketsa (*sketch*). Sketsa ini ditulis dalam editor teks. Sketsa disimpan dengan ekstensi file .ino, yang memiliki fitur untuk meng-*cut*, meng-*copy*, mem-*paste*, mencari/ mengganti teks, dll. Area pesan (*console*) memberikan umpan balik, menyimpan dan mengeksport juga menampilkan kesalahan. Konsol menampilkan teks output dalam lingkungan Arduino termasuk detail pesan error dan informasi lainnya. Bagian sudut bawah sebelah kanan jendela menampilkan board dan port serial yang sedang dipakai. Tombol-tombol toolbar (Gambar 3.1.) memungkinkan untuk memverifikasi dan meng-*upload* program, membuat, membuka, dan menyimpan sketsa, juga membuka monitor serial.



Gambar 3.11. Toolbar dan Menu Arduino

Code Program Arduino biasa disebut *sketch* dan dibuat menggunakan bahasa pemrograman C. Program atau sketch yang sudah selesai ditulis di Arduino IDE dapat langsung di-*compile* dan di-*upload* ke Arduino Board.



Gambar 3.12. Sketch dalam Arduino IDE

Secara sederhana, *sketch* dalam Arduino dikelompokkan menjadi 3 blok (Gambar 3.2.), yakni : Header, Setup dan Loop. Untuk program yang lebih kompleks akan ada blok lain berupa fungsi-fungsi pendukung.

## Header

Pada bagian ini biasanya ditulis definisi-definisi penting yang akan digunakan selanjutnya dalam program, misalnya penggunaan library dan pendefinisian variable. Code dalam blok ini dijalankan hanya sekali pada waktu compile. Di bawah ini contoh code untuk mendeklarasikan variable led (integer) dan sekaligus di isi dengan angka 13

```
int led = 13;
```

## Setup

Di sinilah awal program Arduino berjalan, yaitu di saat awal, atau ketika power on Arduino board. Biasanya di blok ini diisi penentuan apakah suatu pin digunakan sebagai input atau output, menggunakan perintah pinMode. Inisialisasi variable juga bisa dilakukan di blok ini

```
// the setup routine runs once when you press reset:  
void setup() { // initialize the digital pin as an output.  
  pinMode(led, OUTPUT);  
}
```

OUTPUT adalah suatu makro yang sudah didefinisikan Arduino yang berarti = 1. Jadi perintah di atas sama dengan pinMode(led, 1); Suatu pin bisa difungsikan sebagai OUTPUT atau INPUT. Jika difungsikan sebagai output, dia siap mengirimkan arus listrik (maksimum 100 mA) kepada beban yang disambungkannya. Jika difungsikan sebagai INPUT, pin tersebut memiliki impedance yang tinggi dan siap menerima arus yang dikirimkan kepadanya.

## Loop

Blok ini akan dieksekusi secara terus menerus. Apabila program sudah sampai akhir blok, maka akan dilanjutkan dengan mengulang eksekusi dari awal blok. Program akan berhenti apabila tombol power Arduino di matikan. Di sinilah fungsi utama program Arduino kita berada.

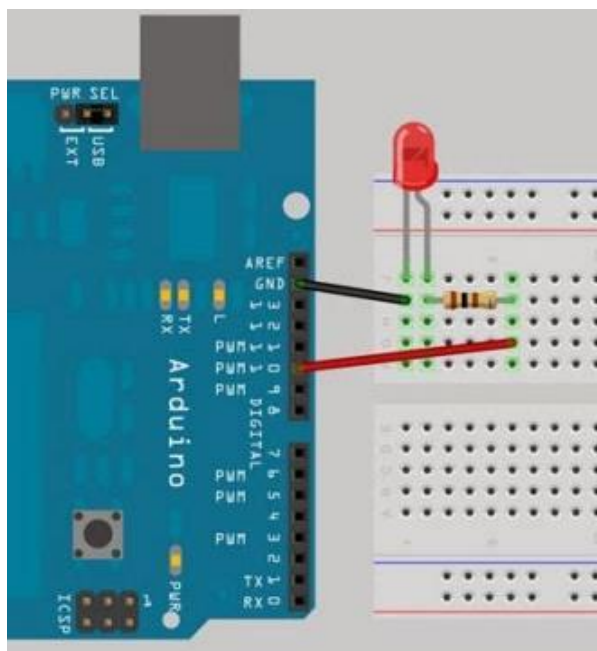


```
void loop() {  
  digitalWrite(led, HIGH); // nyalakan LED delay(1000);  
  // tunggu 1000 milidetik digitalWrite(led, LOW);  
  // matikan LED  
  delay(1000); // tunggu 1000 milidetik  
}
```

Perintah `digitalWrite (pinNumber,nilai)` akan memerintahkan arduino untuk menyalakan atau mematikan tegangan di `pinNumber` tergantung nilainya. Jadi perintah di atas `digitalWrite(led,HIGH)` akan membuat pin nomor 13 (karena di header dideklarasikan `led = 13`) memiliki tegangan = 5V (HIGH). Hanya ada dua kemungkinan nilai `digitalWrite` yaitu HIGH atau LOW yang sebetulnya adalah nilai integer 1 atau 0

#### 4. Program Sederhana

Program sederhana adalah *Led Blink*, program ini akan mengakses pin 10 dan memerintahkan Arduino untuk mengulang *blink led*, Gambar 3.3.



Gambar 3.13. Wiring LED blinking

## Sketch

```
// Project 1 - LED Flasher int ledPin = 10;

void setup() {
  pinMode(ledPin, OUTPUT);
}

void loop() {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
}
```

## Pembahasan Sketch

```
// Project 1 - LED Flasher
```

Ini adalah komentar baris yang berguna untuk dokumentasi program, kompiler akan mengabaikan bagian ini. Baris komentar berguna bagi programmer agar bisa mengerti maksud program.

```
int ledPin = 10;
```

Inisialisasi variable, dalam hal ini inisialisasi variable bernama ledPin dengan type data integer dan nilai 10.

```
void setup() {  
  pinMode(ledPin, OUTPUT);  
}
```

Setiap sketch Arduino wajib memiliki fungsi setup() dan loop(). Fungsi setup() dipanggil hanya sekali saat pertama kali program berjalan. Fungsi setup() biasanya tempat untuk men-setup hal-hal umum agar program siap dijalankan, seperti setup pin modes, setting serial baud rates, dan lainnya. Pada sketch Led Blink, fungsi setup hanya memiliki 1 baris perintah yaitu

```
pinMode(ledPin, OUTPUT);
```

pinMode fungsi yang berguna untuk memberitahu arduino bahwa pin pada board akan digunakan sebagai input atau output. Dalam baris program di atas, memberitahu arduino untuk menset pin 10 (nilai ledPin adalah 10) sebagai Output.

```
void loop() {  
  digitalWrite(ledPin, HIGH);  
  delay(1000);  
  digitalWrite(ledPin, LOW);  
  delay(1000);  
}
```

Fungsi loop()function adalah program utama yang dipanggil secara continue selama Arduino menyala. Setiap perintah dalam fungsi loop()akan dipanggil satu persatu sampai perintah terakhir dalam

blok loop dicapai, lalu Arduino akan kembali ke awal perintah di blok fungsi `loop()`, sampai Arduino dimatikan atau tombol reset ditekan.

## 5. Ringkasan

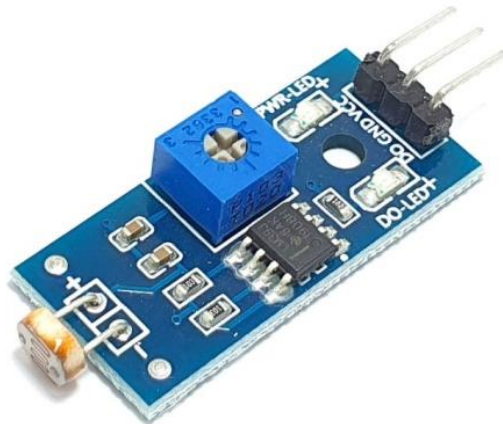
1. Code Program Arduino biasa disebut *sketch* dan dibuat menggunakan bahasa pemrograman C. Program atau sketch yang sudah selesai ditulis di Arduino IDE dapat langsung di-*compile* dan di-*upload* ke Arduino Board.
2. Secara sederhana, *sketch* dalam Arduino dikelompokkan menjadi 3 blok yakni : Header, Setup dan Loop.
3. Setiap sketch Arduino wajib memiliki fungsi `setup()` dan `loop()`. Fungsi `setup()` dipanggil hanya sekali saat pertama kali program berjalan. Fungsi `loop()` adalah program utama yang dipanggil secara continue selama Arduino menyala.

# BAB 4

## Penggunaan Sensor Dalam Arduino

### 1. Sensor Cahaya (LDDR)

*Light Dependent Resistor* atau LDR atau disebut juga Photoresistor merupakan salah satu jenis resistor dimana besarnya hambatan/resistansinya berdasarkan intensitas cahaya yang diterima. Nilai hambatan dari LDR akan besar apabila intensitas cahaya yang diterimanya rendah. Hambatan dari LDR akan kecil apabila instensitas cahaya yang diterima tinggi. Artinya nilai hambatan dari LDR berbanding terbalik dengan intensitas cahaya yang diterimanya. Dalam kondisi terang (instensitas cahaya tinggi), nilai hambatan dari LDR dapat mencapai 200 Kilo Ohm, sedangkan pada kondisi gelap (intensitas cahaya rendah) nilai hambatannya menurun menjadi 500 Ohm. Berikut gambar module sensor LDR di bawah ini.



**Gambar 4.1** Module LDR

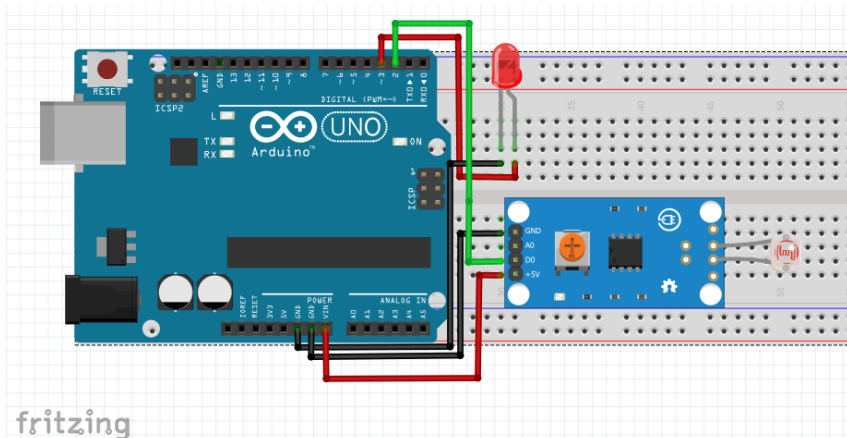
Modul sensor LDR memiliki 3 pin yaitu DO (output), GND, dan VCC. Modul sensor LDR bekerja pada tegangan DC 3.3 Volt - 5 Volt.

## Rangkaian Sensor Cahaya

Sensor cahaya yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor LDR	1
LED 5 mm	1
Breadboard	1
Kabel Jumper	5

Untuk lebih jelasnya perhatikan rangkaian dari sensor ultrasonikpada Gambar 4.2 berikut.



**Gambar 4.2** Rangkaian sensor cahaya

- Pasangkan pin ground sensor LDR ke pin ground arduino.
- Pasangkan pin vcc sensor LDR ke pin vcc aduino.
- Pasangkan pin DO (out) sensor LDR ke pin 2 arduino.
- Pasangkan kaki anoda LED ke pin 3 arduino dan kaki katoda LED ke pin ground arduino

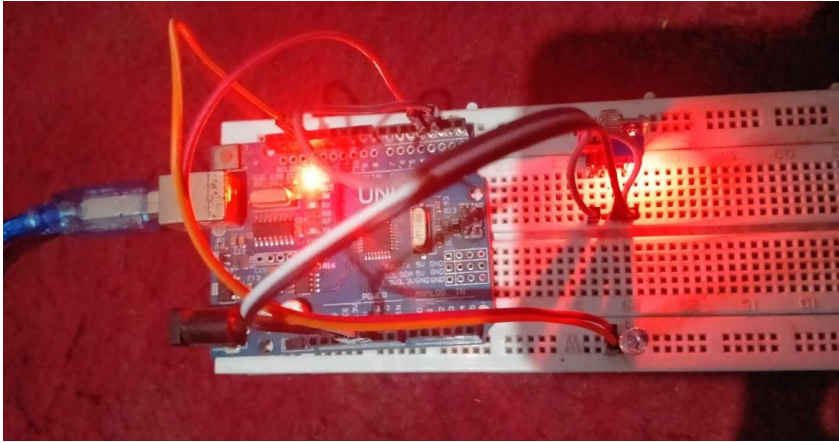
## Program Sensor Cahaya

Masukkan program berikut ini ke dalam aplikasi IDE Arduino kemudian upload ke arduino menggunakan kabel USB Downloader.

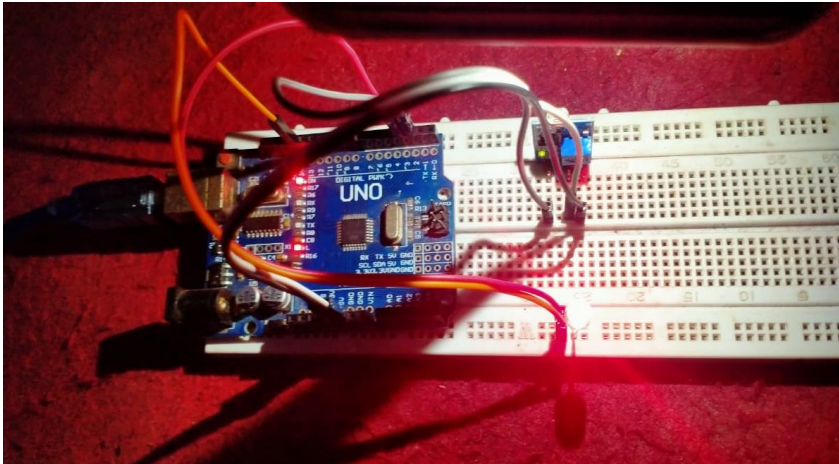
```
const int sensorLDR = 2;
const int Led = 3;
void setup()
{
  Serial.begin (9600);
  pinMode(sensorLDR, INPUT);
  pinMode(Led, OUTPUT);
  delay(1000);
}
void loop()
{
  int nilaiSensor = digitalRead(sensorLDR);
  if(nilaiSensor == LOW)
  {
    digitalWrite(Led, HIGH);
  }
  Else
  {
    digitalWrite(Led, LOW);
  }
  Serial.println(nilaiSensor);
  delay(500);
}
```

## Hasil Percobaan

Berikut hasil percobaan dari sensor cahaya (Module LDR)



**Gambar 4.3** Sensor sebelum ada cahaya



**Gambar 4.4** Sensor setelah ada halangan

## **2. Sensor cahaya (HC-SR04)**

Gelombang ultrasonik adalah gelombang bunyi yang mempunyai frekuensi sangat tinggi yaitu 20.000 Hz. Bunyi ultrasonik tidak dapat di dengar oleh telinga manusia. Bunyi ultrasonik dapat di dengar oleh anjing, kucing, kelelawar, dan lumba-lumba. Bunyi ultrasonik bisa merambat melalui zat padat, cair dan gas. Reflektivitas bunyi ultrasonik di permukaan zat padat hampir sama dengan reflektivitas bunyi ultrasonik di permukaan



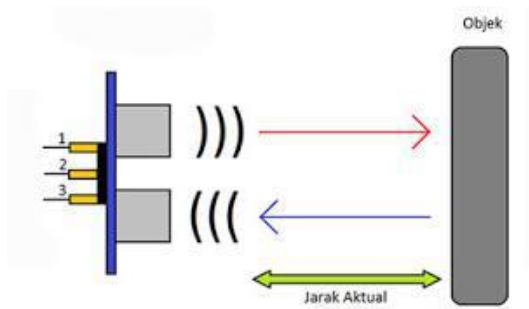
zat cair. Akan tetapi, gelombang bunyi ultrasonik akan diserap oleh tekstil dan busa.

Sensor jarak yang akan kita buat tersusun oleh sensor ultrasonik HC-SR04 yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Sensor ini merupakan sensor ultrasonik siap pakai yang sudah terdiri dari pengirim, penerima, dan pengontrol gelombang ultrasonik. Alat ini bisa digunakan untuk mengukur jarak benda dari 2cm – 4m dengan akurasi 3mm. Alat ini memiliki 4 pin, pin VCC, GND, Trigger, dan Echo. Pin VCC untuk listrik positif dan GND untuk ground-nya. Pin Trigger untuk trigger keluarnya sinyal dari sensor dan pin Echo untuk menangkap sinyal pantul dari benda.



**Gambar 4.5** Sensor Ultrasonik HC-SR04

Pada sensor ultrasonik, gelombang ultrasonik dibangkitkan melalui sebuah alat yang disebut dengan piezoelektrik dengan frekuensi tertentu. Piezoelektrik ini akan menghasilkan gelombang ultrasonik (umumnya berfrekuensi 40kHz) ketika sebuah osilator diterapkan pada benda tersebut. Secara umum, alat ini akan menembakkan gelombang ultrasonik menuju suatu area atau suatu target. Gelombang yang dipancarkan akan merambat sebagai gelombang bunyi dengan kecepatan sekitar 340 m/s.



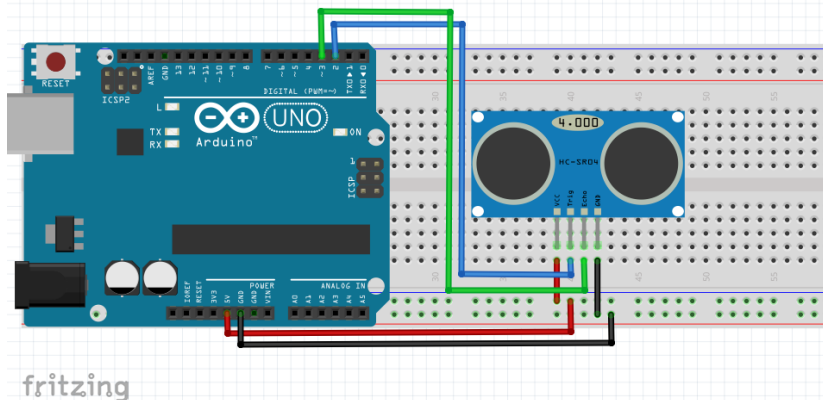
Gambar 4.6 Cara Kerja Sensor Ultrasonik

### Rangkaian Sensor Jarak

Sensor jarak yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor Ultrasonik HC-SR04	1
Breadboard	1
Kabel Jumper	6

Untuk lebih jelasnya perhatikan rangkaian dari sensor ultrasonik pada Gambar 4.3



Gambar 4.7 Rangkaian Sensor Jarak

- Pasangkan kabel jumper dari pin GND dan 5 V Arduino ke bagian bawah Breadboard.

- Pasangkan Sensor Ultrasonik HC-SR04 ke *Breadboard* dimana VCC ke 5 V pada *Breadboard*, Trig ke Pin 2 Arduino, Echo ke Pin 3 Arduino, dan GND ke GND *Breadboard*.
- Untuk pemasangan kabel jumper, bisa dengan warna yang berbeda atau lubang yang berbeda, asalkan komponen dan kabel penghubung sesuai dengan gambar diatas.

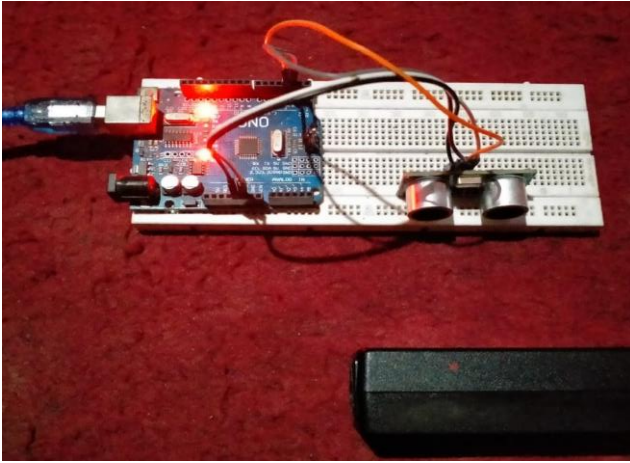
### Program Sensor Jarak

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

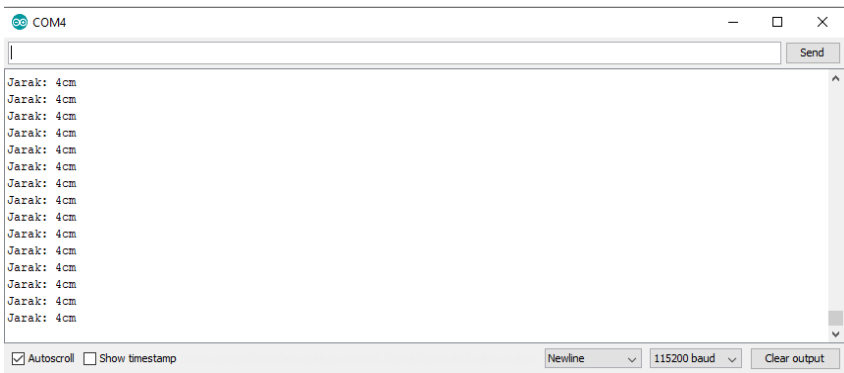
```
#include <NewPing.h>
#define TRIGGER_PIN 2
#define ECHO_PIN 3
#define MAX_DISTANCE 200
NewPing solar(TRIGGER_PIN, ECHO_PIN,
MAX_DISTANCE);
void setup()
{
  Serial.begin(115200);
}
void loop()
{
  delay(1000);
  int jarak=solar.ping_cm();
  Serial.print("Jarak: ");
  Serial.print(jarak);
  Serial.println("cm");
}
```

### Hasil Percobaan

Berikut hasil percobaan sensor jarak (HC-SR04) dengan serial monitor.



**Gambar 4.8.** Percobaan sensor jarak (HC-SR04)



**Gambar 4.5.** Hasil serial monitor sensor jarak (HC-SR04)

### **Sensor (HC-SR04) Dengan Modul LCD 16x2**

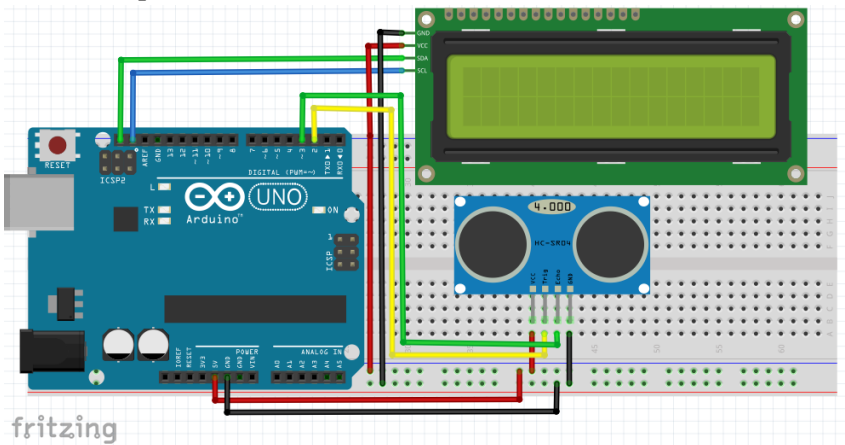
Pada *project* ini akan melanjutkan bahasan pada *project* sebelumnya. Jika pada bahasan sebelumnya hasil pengukuran ditampilkan di Serial Monitor IDE Arduino, maka pada bahasan ini akan ditampilkan menggunakan modul LCD 16x2 yang sudah dilengkapi dengan I2C Konverter.

### **Rangkaian Sensor Jarak Dengan Modul LCD 16x2**

Sensor jarak yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor Ultrasonik HC-SR04	1
LCD 16x2 dengan modul I2C konverter	1
<i>Breadboard</i>	1
Kabel jumper	10

Untuk lebih jelasnya perhatikan rangkaian dari sensor ultrasonik pada Gambar 5.6



**Gambar 4.6** Rangkaian sensor jarak dengan LCD 16x2 dan I2C

Buatlah rangkaian seperti pada **Gambar 5.1**.

- Pin GND dan 5 V Arduino ke *Breadboard*.
- LCD 16x2 dengan modul I2C Konverter, dimana VCC ke 5 V *Breadboard*, GND ke GND *Breadboard*. SDA ke SDA Arduino dan SCL ke SCL Arduino.
- Sensor Ultrasonik HC-SR04, dimana VCC ke VCC *Breadboard*, Trig ke Pin 2 Arduino, Echo ke Pin 3 Arduino, dan GND ke GND *Breadboard*.

## Program

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

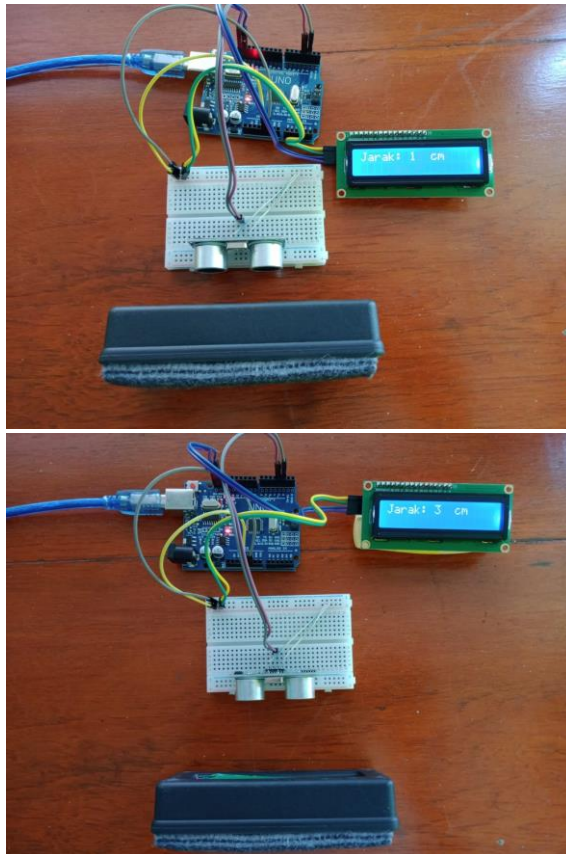
```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
#include <NewPing.h>
#define TRIGGER_PIN 2
#define ECHO_PIN 3
#define MAX_DISTANCE 200
NewPing solar(TRIGGER_PIN, ECHO_PIN,
MAX_DISTANCE);

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

void loop()
{
  delay(1000);
  int jarak=solar.ping_cm();
  lcd.setCursor(0,0);
  lcd.print("Jarak: ");
  lcd.setCursor(7,0);
  lcd.print(jarak);
  lcd.setCursor(10,0);
  lcd.print("cm");
  delay(1000);lcd.clear();
}
```

## Hasil Percobaan

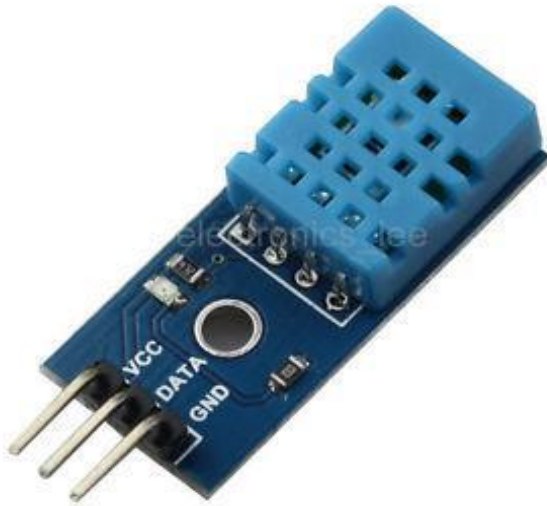
Berikut hasil percobaan sensor jarak (HC-SR04) dengan modul LCD I2C 16x2.



**Gambar 4.7.** Percobaan sensor jarak (HC-SR04) dan LCD I2C 16x2

### **3. Sensor DHT11**

Sensor DHT11 ini merupakan sensor suhu dan kelembaban, data suhu dan kelembaban ruang dideteksi oleh sensor DHT11. Keluaran sensor DHT11 berupa sinyal digital yang sudah terkalibrasi. Jangkauan pengukuran temperatur dari sensor ini adalah 0-50°C dan jangkauan pengukuran kelembaban relatif sebesar 20-90%. Sensor DHT11 memiliki 3 pin yaitu vcc, data, dan gnd. Sensor DHT11 membutuhkan catu daya sebesar 3.3 sampai 5 Volt DC. Keakuratan untuk kelembaban relatifnya sebesar  $\pm 4\%$  dan keakuratan untuk temperatur sebesar  $\pm 2^\circ\text{C}$ . Bentuk fisik dari sensor DHT11 ditunjukkan pada Gambar 6.1.



**Gambar 4.8** Sensor DHT11

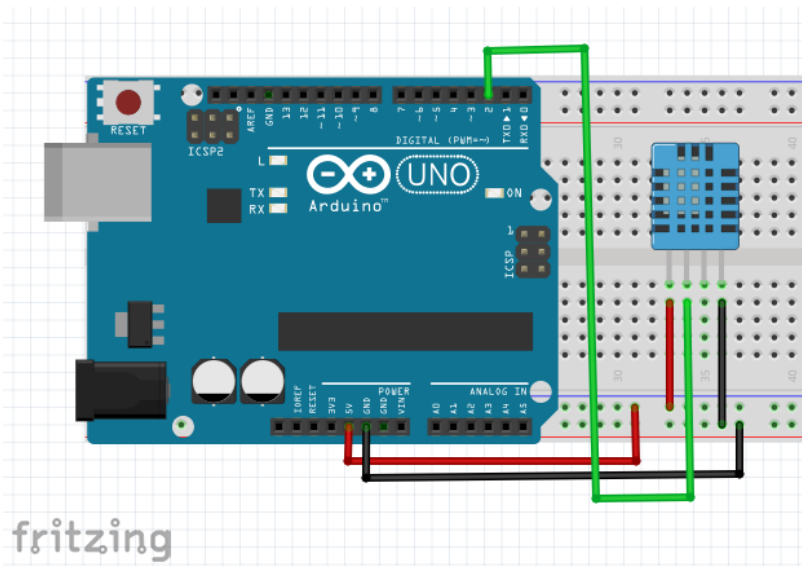
### **Rangkaian Sensor DHT11**

Pada project kali ini mencoba membuat rangkaian sensor suhu dan kelembaban menggunakan sensor DHT11. Berikut ini komponen-komponen yang dibutuhkan.

<b>Komponen</b>	<b>Jumlah</b>
Arduino UNO	1
Sensor DHT11	1
<i>Breadboard</i>	1
Kabel jumper	5

Untuk lebih jelasnya perhatikan rangkaian dari sensor DHT11 pada Gambar 6.2





**Gambar 4.9** Rangkaian sensor DHT11

Buatlah rangkaian seperti **Gambar 4.9**

- Pasangkan sensor DHT11 pada *Breadboard*
- Pasangkan kaki vcc sensor DHT11 ke pin 5 V arduino
- Pasangkan kaki gnd sensor DHT11 ke pin gnd arduino
- Pasangkan kaki Data (out) sensor DHT11 ke pin 2 arduino

### Program Sensor DHT11

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);

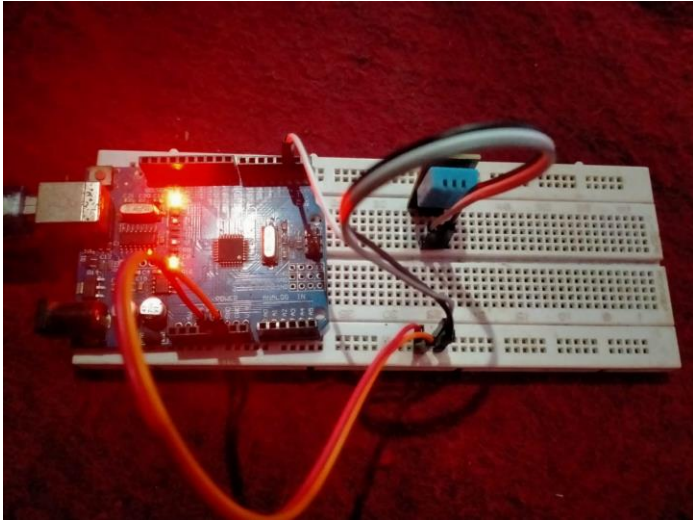
void setup()
{
  Serial.begin(9600);
  Serial.println("DHTxx test!");
}
```

```
dht.begin();
}
void loop()
{
  delay(2000);
  float h = dht.readHumidity();
  float t = dht.readTemperature();
  float f = dht.readTemperature(true);

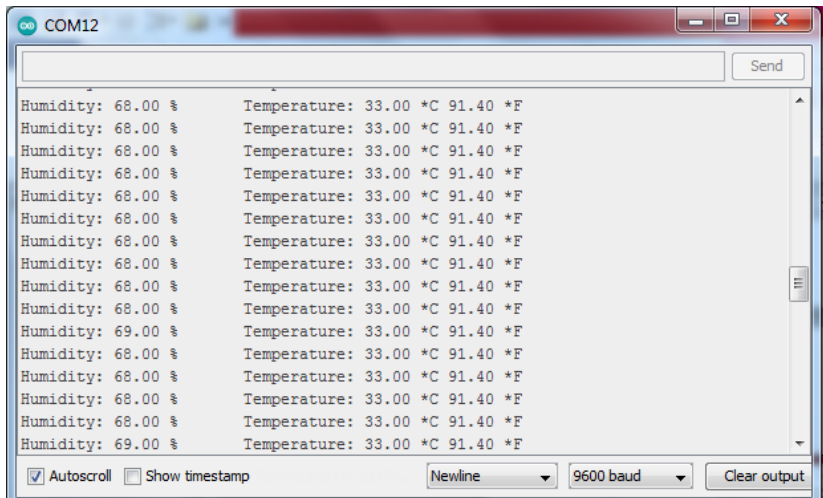
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }
  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(" *C ");
  Serial.print(f);
  Serial.println(" *F\t");
}
```

### Hasil percobaan

Berikut hasil percobaan sensor DHT11 dengan serial monitor.



Gambar 4.10 Percobaan Sensor DHT11



Gambar 4.11 Hasil serial monitor sensor DHT11

### Sensor DHT11 Dengan Modul LCD16x2

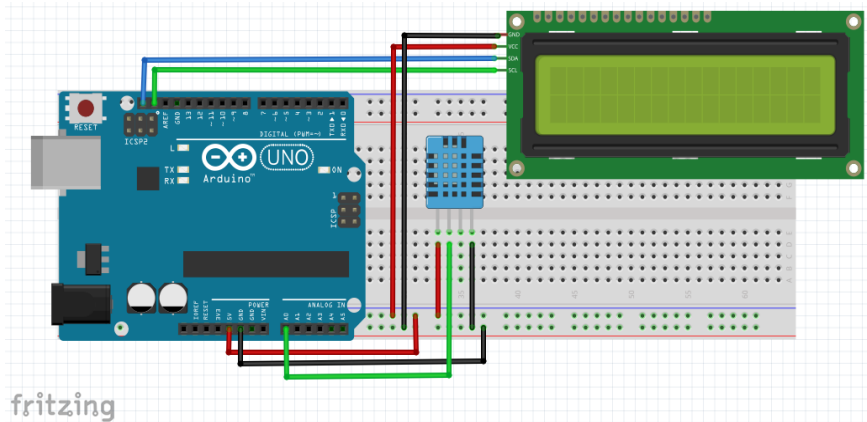
Pada *project* ini akan melanjutkan bahasan pada *project* sebelumnya. Jika pada bahasan sebelumnya hasil pengukuran sensor DHT11 ditampilkan di Serial Monitor IDE Arduino, maka pada bahasan ini akan ditampilkan menggunakan LCD 16x2 yang sudah dilengkapi modul I2C Konverter.

## Rangkaian

Sensor DHT11 yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor DHT11	1
LCD 16x2 dengan modul I2C Konverter	1
<i>Breadboard</i>	1
Kabel jumper	9

Untuk lebih jelasnya perhatikan rangkaian dari sensor DHT11 pada Gambar 7.1



**Gambar 4.12** Rangkaian Sensor DHT11 dengan Modul I2C

Buatlah rangkaian seperti pada **Gambar 4.12**

- Pasangkan Pin GND dan 5 V Arduino ke *Breadboard*
- Pasangkan kaki Data (out) sensor DHT11 ke pin A0 arduino
- LCD 16x2 dengan modul I2C Konverter, dimana VCC ke 5 V *Breadboard*, GND ke board Arduino. SDA ke SDA Arduino dan SCL ke SCL Arduino.

## 7.1 Program

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT11 // DHT 11
DHT dht(DHTPIN, DHTTYPE);

void setup()
{
  dht.begin();
  lcd.init();
  lcd.backlight();
}

void loop()
{
  delay(2000);

  float h = dht.readHumidity();
  float t = dht.readTemperature();
  float f = dht.readTemperature(true);

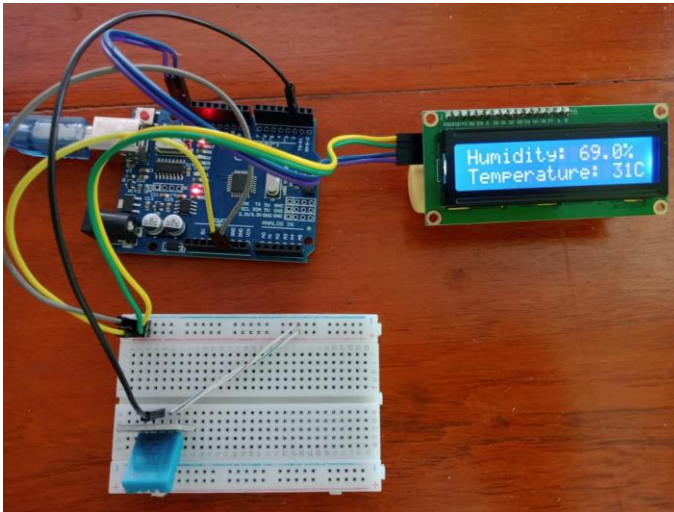
  if (isnan(h) || isnan(t) || isnan(f)) {
    lcd.setCursor(0,0);
    lcd.print("PEMBACAAN GAGAL");
    delay(1000);
    lcd.clear();
    return;
  }

  lcd.setCursor(0,0);
```

```
lcd.print("Humidity: ");  
lcd.setCursor(10,0);  
lcd.print(h);  
lcd.setCursor(14,0);  
lcd.print("%");  
lcd.setCursor(0,1);  
lcd.print("Temperature: ");  
lcd.setCursor(13,1);  
lcd.print(t);  
lcd.setCursor(15,1);  
lcd.print("C");  
delay(1000);  
lcd.clear();  
}
```

### Hasil percobaan

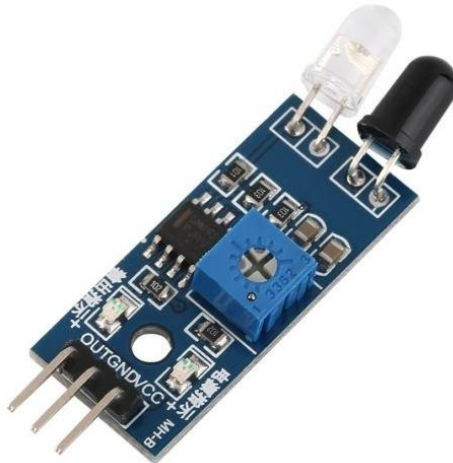
Berikut hasil percobaan sensor DHT11 dengan LCD 16x2.



**Gambar 4.13.** Percobaan sensor DHT11 dan LCD I2C 16x2

## 4. Sensor IR Proximity

Sensor IR proximity atau Infrared Proximity Sensor merupakan sensor inframerah yang dapat digunakan untuk mendeteksi halangan, mendeteksi warna (hitam atau putih) mendeteksi gerakan dll. Sensor infrared ini sangat rentan terhadap cahaya sekitar, penggunaan diluar ruangan bisa menambahkan penutup pada sensor untuk mengurangi cahaya yang masuk. Sensor infrared ini memiliki 3 pin yaitu pin vcc, gnd, dan out. Dan sensor infrared ini dapat membaca jarak dari 2 cm sampai 30 cm, dan teganan kerja 3.3 volt - 5 volt, dan memiliki sudut deteksi 30°.



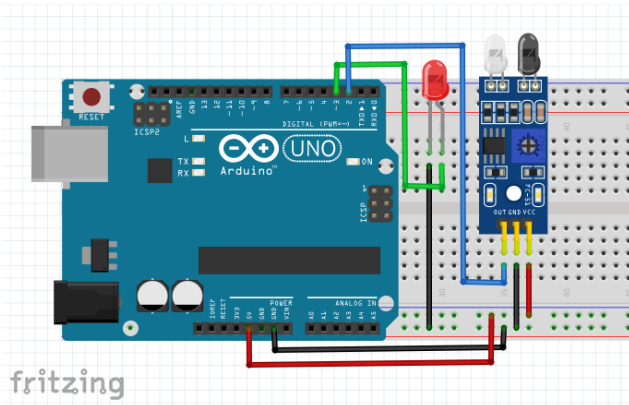
Gambar 4.14 IR proximity

### Rangkaian Sensor IR Proximity

Sensor IR Proximity yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor IR Proximity	1
LED 5 mm	1
Breadboard	1
Kabel Jumper	6

Untuk lebih jelasnya perhatikan rangkaian dari sensor IR Proximity pada Gambar 4.15



**Gambar 4.15** Rangkaian sensor IR proximity

Buatlah rangkaian seperti pada **Gambar 4.15**.

- Pasangkan Sensor IR dan LED ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki out sensor ke pin 2 Arduino
- Pasangkan kaki katoda LED ke gnd *Breadboard* dan kaki anoda LED ke pin 3 arduino

### Program Sensor IR Proximity

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```
const int sensorIR = 2;
const int Led = 3;
void setup()
{
  Serial.begin (9600);
  pinMode(sensorIR, INPUT);
  pinMode(Led, OUTPUT);
  delay(1000);
```



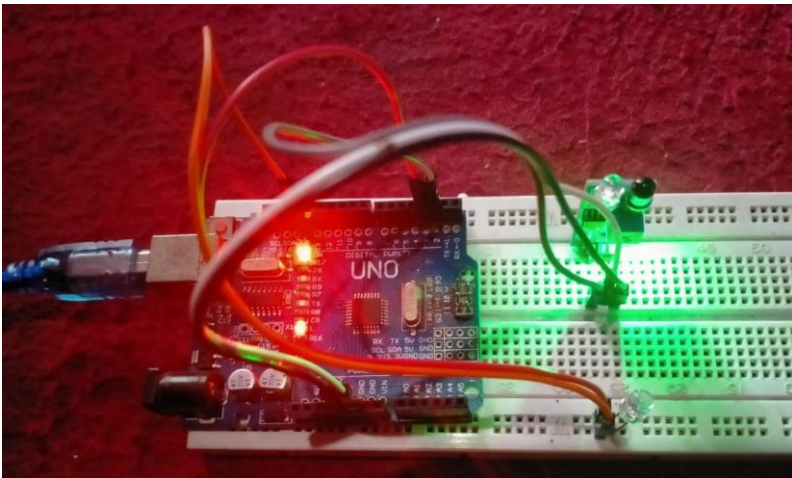
```

}
void loop()
{
  int nilaiSensor = digitalRead(sensorIR);
  if(nilaiSensor == LOW)
  {
    digitalWrite(Led, HIGH);
  }
  Else
  {
    digitalWrite(Led, LOW);
  }
  Serial.println(nilaiSensor);
  delay(500);
}

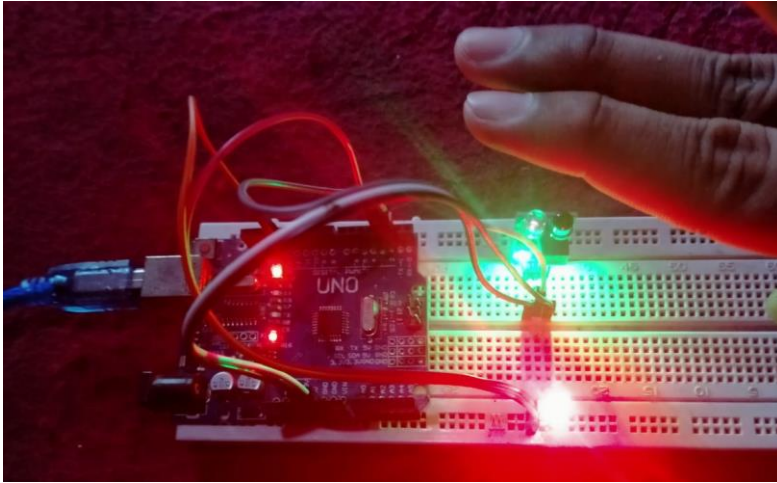
```

### Hasil percobaan

Berikut hasil percobaan dari sensor IR Proximity (Module Infrared)



**Gambar 4.16** Sensor sebelum ada halangan



**Gambar 4.17** Sensor setelah ada halangan

## **5. Sensor Sentuh (Touch Sensor)**

Sensor Sentuh adalah peralatan yang digunakan untuk mengubah suatu besaran fisik menjadi besaran listrik sehingga dapat dianalisa dengan rangkaian listrik tertentu. Pada saat ini, sensor tersebut telah dibuat dengan ukuran sangat kecil dengan orde nanometer. Ukuran yang sangat kecil ini sangat memudahkan pemakaian dan menghemat energi. Sensor sentuh ini mempunyai 3 pin yaitu pin VCC, GND, dan SIG (Signal/data). Bentuk fisik dari sensor sentuh atau touch sensor dapat di lihat pada Gambar 4.18



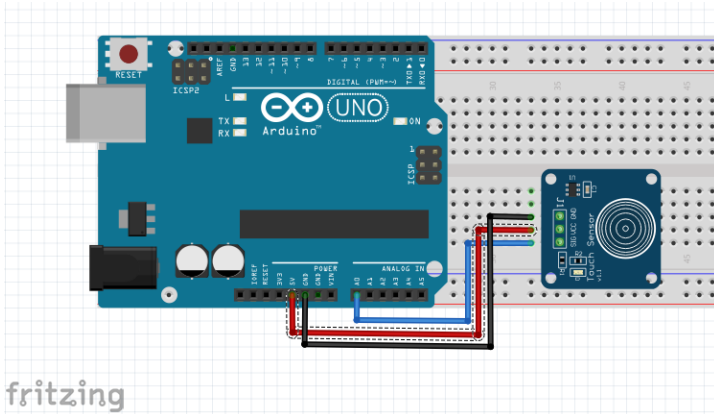
**Gambar 4.18** Sensor Sentuh (Touch Sensor)

### **Rangkaian Sensor Sentuh (Touch Sensor)**

Sensor Sentuh yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

<b>Komponen</b>	<b>Jumlah</b>
Arduino UNO	1
Sensor Sentuh (Touch Sensor)	1
<i>Breadboard</i>	1
Kabel Jumper	3

Untuk lebih jelasnya perhatikan rangkaian dari sensor sentuh pada Gambar 9.2



**Gambar 4.19** Rangkaian Sensor Sentuh

Buatlah rangkaian seperti pada **Gambar 4.19**.

- Pasangkan Sensor Sentuh dan ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki out sensor ke pin A0 Arduino

### Program Sensor Sentuh (Touch Sensor)

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

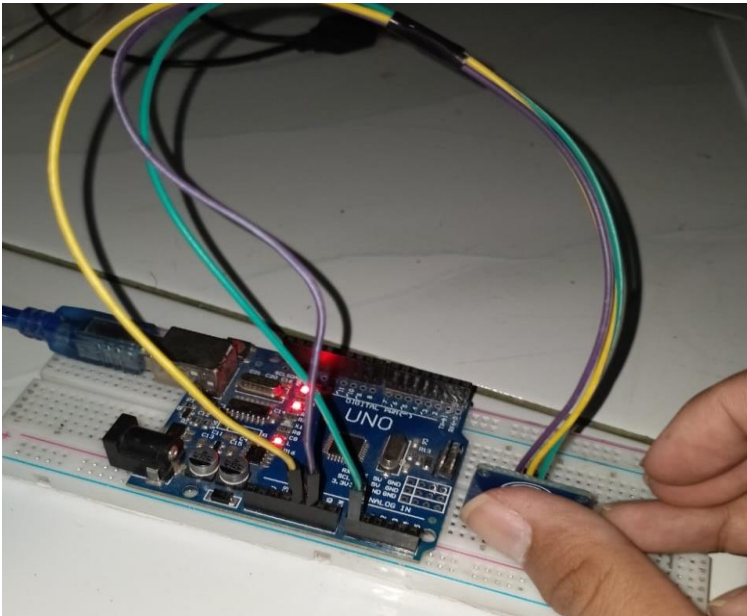
```
int value = 0; //membuat variabel untuk menyimpan nilai
dari sensor
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600); //mengatur baud rate ke 9600
  pinMode(A0, INPUT); //mengatur pin A0 sebagai input
}

void loop()
{
```

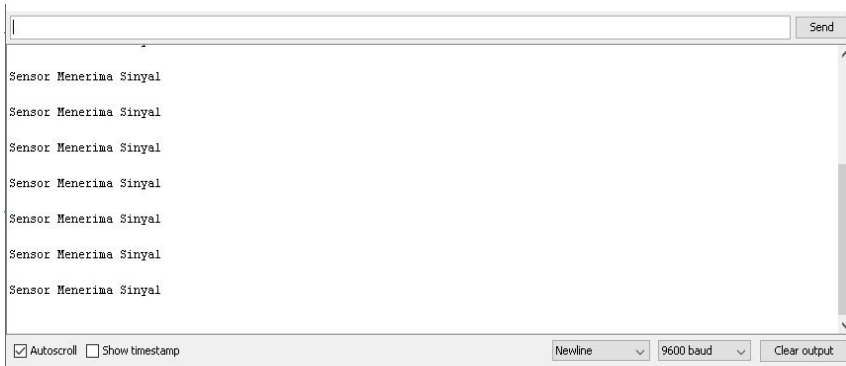
```
// put your main code here, to run repeatedly:  
value = digitalRead(A0); //membaca nilai sensor dan  
menyimpan ke vaiabel value  
if(value == 1) // jika nilai dari sensor adalah 1 maka/..  
{  
  Serial.print("Sensor Menerima Sinyal\n\n"); //mencetak  
kalimat pada serial monitor  
}  
else  
{  
  
}  
}
```

### Hasil Percobaan Sensor Sentuh (Touch Sensor)

Berikut hasil percobaan dari Sensor Sentuh (Touch Sensor)



**Gambar 4.20** Percobaan Sensor Sentuh



**Gambar 9.4** Hasil Serial Monitor Sensor Sentuh

## **6. Sensor Suara (Sound Voice)**

Sensor suara adalah sebuah alat yang mampu mengubah gelombang *Sinusioda* suara menjadi gelombang sinus energi listrik (Alternating Sinusioda Electric Current). Sensor suara berkerja berdasarkan besar/kecilnya kekuatan gelombang suara yang mengenai membran sensor yang menyebabkan Bergeraknya membran sensor yang juga terdapat sebuah kumparan kecil di balik membran tadi naik & turun. Oleh karena kumparan tersebut sebenarnya adalah ibarat sebuah pisau berlubang-lubang, maka pada saat ia bergerak naik-turun, ia juga telah membuat gelombang magnet yang mengalir melewatinya terpotong-potong. Kecepatan gerak kumparan menentukan kuat-lemahnya gelombang listrik yang dihasilkannya. Sensor suara adalah sensor yang cara kerjanya merubah besaran suara menjadi besaran listrik, dan dipasaran sudah begitu luas penggunaannya. Komponen yang termasuk dalam Sensor suara yaitu electric condenser microphone atau mic kondenser. Sensor suara memiliki 3 kaki yaitu Vcc, Gnd, dan Out. Bentuk fisik dari Sensor Suara seperti Gambar 10.1



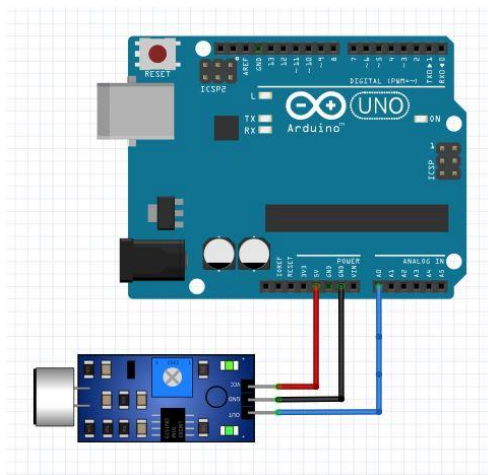
**Gambar 4.21** Sensor Suara

### Rangkaian Sensor Suara (*Sound Voice*)

Sensor Suara yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor Suara	1
<i>Breadboard</i>	1
Kabel Jumper	3

Untuk lebih jelasnya perhatikan rangkaian dari sensor suara pada Gambar 4.22



**Gambar 4.22** Rangkaian Sensor Suara

Buatlah rangkaian seperti pada **Gambar 4.22**

- Pasangkan Sensor Suara dan ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki out sensor ke pin A0 Arduino

### Program Sensor Suara (Sound Voice)

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

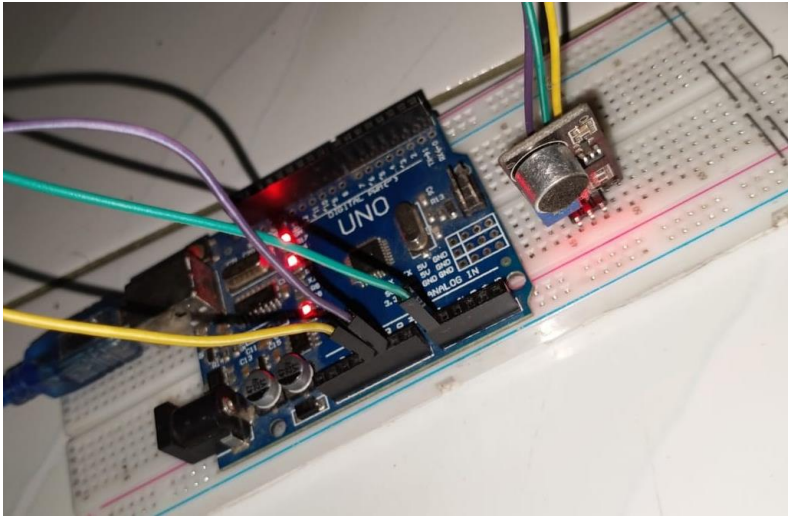
```
int value = 0; //membuat variabel untuk menyimpan nilai
dari sensor
void setup()
{
  // put your setup code here, to run once:
  Serial.begin(9600); //mengatur baud rate ke 9600
  pinMode(A0, INPUT); //mengatur pin A0 sebagai input
}

void loop()
{
  // put your main code here, to run repeatedly:
  value = analogRead(A0); //membaca nilai sensor dan
menyimpan ke vaiabel value
  Serial.println(value); //mencetak nilai dari variable value
}
```

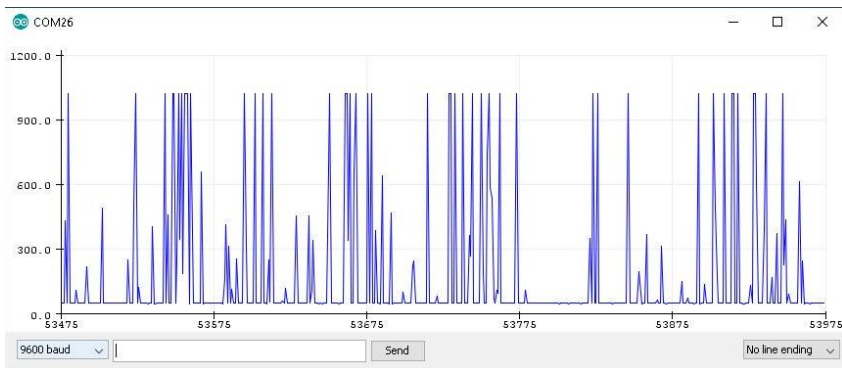
### Hasil Percobaan Sensor Suara (Sound Voice)

Berikut hasil percobaan dari Sensor Suara





**Gambar 4.23** Percobaan Sensor Suara



**Gambar 4.24** Hasil Serial Monitor Sensor Suara

## **7. Sensor Warna (TCS34725)**

Modul sensor warna GY-33 TCS 34725 merupakan sensor penginderaan warna yang memiliki elemen penginderaan cahaya RGB dan Clear. Sensor ini dilengkapi dengan filter blok IR, on-chip terintegrasi dan dilokalisasi ke foto sensor warna, meminimalkan komponen spektrum IR dari cahaya yang masuk dan memungkinkan pengukuran warna dilakukan secara akurat. Sensor ini juga memiliki kisaran dinamis 3.800.000: 1 yang luar biasa

dengan waktu dan penguatan integrasi yang disesuaikan sehingga cocok untuk digunakan di belakang kaca yang gelap. Sensor ini bekerja dari 3.3 Volt sampai 5 Volt, dan sensor ini memiliki 7 kaki/pin yaitu LED, INT, SDA, SCL, 3V3, GND, dan VIN. Bentuk fisik dari sensor warna TCS34725 seperti Gambar 4.25



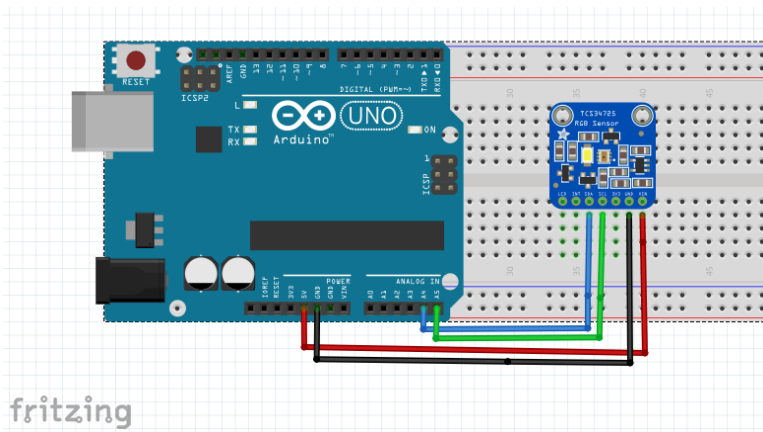
Gambar 4.25 Sensor Warna TCS34725

### Rangkaian Sensor Warna (TCS34725)

Sensor Warna TCS34725 yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor Warna TCS34725	1
Breadboard	1
Kabel Jumper	4

Untuk lebih jelasnya perhatikan rangkaian dari sensor Warna TCS34725 pada Gambar 11.2



**Gambar 4.26** Rangkaian Sensor Warna TCS34725

Buatlah rangkaian seperti pada **Gambar 4.26**.

- Pasangkan Sensor Warna TCS34725 dan ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki SCL sensor ke pin A5 Arduino
- Pasangkan kaki SDA sensor ke pin A4 Arduino

### Program Sensor Warna (TCS34725)

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```
#include <Wire.h> // penggunaan library wire pada arduino
#include "Adafruit_TCS34725.h" // penggunaan library dari sensor warna
#define redpin 3
#define greenpin 5
#define bluepin 6
#define commonAnode true
byte gammatable[256];
```

```

Adafruit_TCS34725 tcs =
Adafruit_TCS34725(TCS34725_INTEGRATIONTIME_50MS,
TCS34725_GAIN_4X);

void setup()
{
  Serial.begin(9600); //memulai baudrate pada 9600
  Serial.println("Color View Test!");

  if (tcs.begin())
  {
    Serial.println("Found sensor");
  }
  else
  {
    Serial.println("No TCS34725 found ... check your
connections");
    while (1); // halt!
  }
#ifdef ARDUINO_ARCH_ESP32
  ledcAttachPin(redpin, 1);
  ledcSetup(1, 12000, 8);
  ledcAttachPin(greenpin, 2);
  ledcSetup(2, 12000, 8);
  ledcAttachPin(bluepin, 3);
  ledcSetup(3, 12000, 8);
#else
  pinMode(redpin, OUTPUT);
  pinMode(greenpin, OUTPUT);
  pinMode(bluepin, OUTPUT);
#endif
  for (int i = 0; i < 256; i++) {
    float x = i;
    x /= 255;
    x = pow(x, 2.5);
  }

```

```

x *= 255;

if (commonAnode)
{
    gammatable[i] = 255 - x;
}
else
{
    gammatable[i] = x;
}
//Serial.println(gammatable[i]);
}
}
void loop()
{
    float red, green, blue;

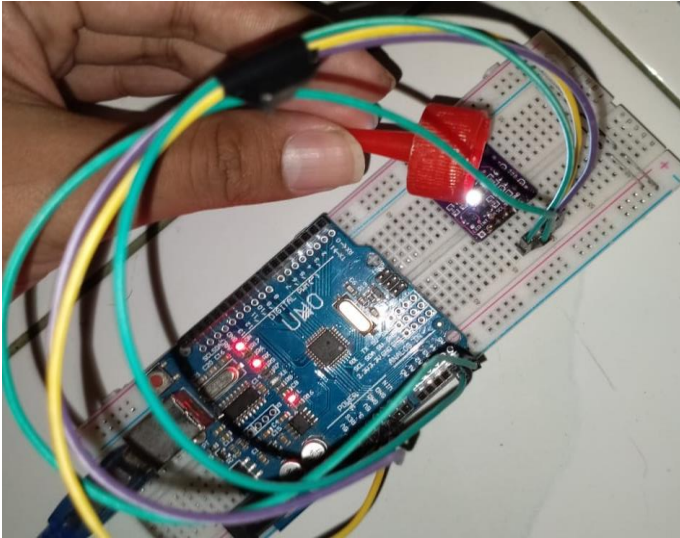
    tcs.setInterrupt(false);
    delay(60);
    tcs.getRGB(&red, &green, &blue);
    tcs.setInterrupt(true);

    Serial.print("R:\t"); Serial.print(int(red));
    Serial.print("\tG:\t"); Serial.print(int(green));
    Serial.print("\tB:\t"); Serial.print(int(blue));
    Serial.print("\n");
#ifdef ARDUINO_ARCH_ESP32
    ledcWrite(1, gammatable[(int)red]);
    ledcWrite(2, gammatable[(int)green]);
    ledcWrite(3, gammatable[(int)blue]);
#else
    analogWrite(redpin, gammatable[(int)red]);
    analogWrite(greenpin, gammatable[(int)green]);
    analogWrite(bluepin, gammatable[(int)blue]);
#endif
}

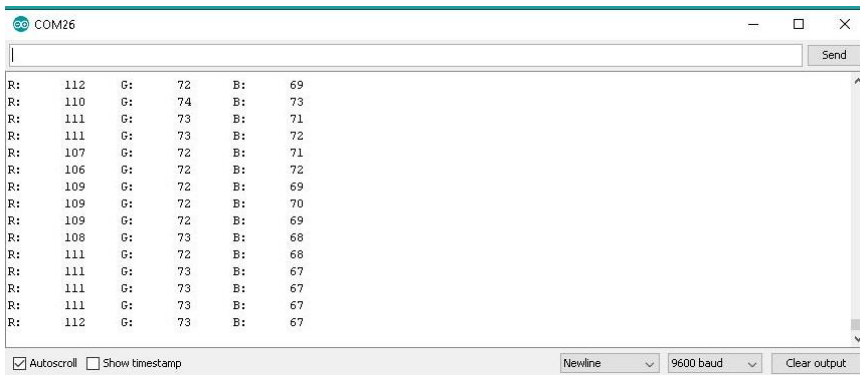
```

## Hasil Percobaan Sensor Warna (TCS34725)

Berikut hasil percobaan dari Sensor Warna (TCS34725)



**Gambar 4.27** Percobaan Sensor Warna (TCS34725)



**Gambar 4.28** Hasil Serial Monitor Sensor Warna (TCS34725)

## 8. Sensor Gyroscope (MPU-6050)

*Gyroscope* merupakan sensor yang akan mengukur kecepatan sudut dalam  $\alpha$ ,  $\beta$ , dan  $\gamma$ . *Gyroscope* sering digunakan untuk membantu penstabil dan memperbaiki perubahan arah dan

pengarahan. Walaupun tidak seperti *accelerometer*, *gyroscope* tidak memiliki nilai referensi yang pasti, dan hanya akan mengukur perubahannya saja. Untuk melakukan pengaturan *gyroscope* sesuai dengan yang diinginkan, maka gambaran nilai rata-rata perubahan sudut maksimumnya (dalam derajat per detik) sangat diperlukan. Sensor ini bekerja pada tegangan 3.3 Volt sampai 5 Volt, dan sensor *gyroscope* memiliki 8 kaki/pin yaitu VCC, GND, SCL, SDA, XDA, XCL, ADO, dan INT. Bentuk fisik dari sensor *gyroscope* seperti Gambar 12.1



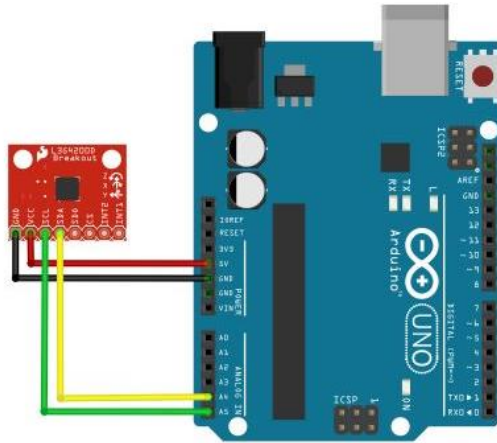
Gambar 4.29 Sensor Gyroscope MPU-6050

### Rangkaian Sensor Gyroscope (MPU-6050)

Sensor *Gyroscope* MPU-6050 yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor <i>Gyroscope</i> MPU-6050	1
<i>Breadboard</i>	1
Kabel Jumper	4

Untuk lebih jelasnya perhatikan rangkaian dari sensor *Gyroscope* MPU-6050 pada Gambar 4.30



**Gambar 4.30** Rangkaian Sensor *Gyroscope* MPU-6050

Buatlah rangkaian seperti pada **Gambar 4.30**..

- Pasangkan Sensor *Gyroscope* MPU-6050 dan ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki SCL sensor ke pin A5 Arduino.
- Pasangkan kaki SDA sensor ke pin A4 Arduino.

### Program Sensor Gyroscope (MPU-6050)

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```
#include "I2Cdev.h" //pemanggilan library i2c dari sensor
mpu6050
#include "MPU6050.h" //library dari sensor mpu6050
#if I2CDEV_IMPLEMENTATION ==
I2CDEV_ARDUINO_WIRE
#include "Wire.h" //library wire dari arduino
#endif
MPU6050 accelgyro;
int16_t ax, ay, az;
int16_t gx, gy, gz;
```



```

#define OUTPUT_READABLE_ACCELYGYRO
bool blinkState = false;

void setup() {
  #if I2CDEV_IMPLEMENTATION ==
  I2CDEV_ARDUINO_WIRE
    Wire.begin();
  #elif I2CDEV_IMPLEMENTATION ==
  I2CDEV_BUILTIN_FASTWIRE
    Fastwire::setup(400, true);
  #endif
  Serial.begin(38400); //mengatur baud rate ke 38400
  Serial.println("Initializing I2C devices...");
  accelgyro.initialize();
  Serial.println("Testing device connections...");
  Serial.println(accelgyro.testConnection() ? "MPU6050
connection successful" : "MPU6050 connection failed");
}

void loop()
{
  accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
  #ifdef OUTPUT_READABLE_ACCELYGYRO
  Serial.print("a/g:\t");
  Serial.print(map(ax, -17000, 17000, 0, 180)); Serial.print("\t");
  Serial.print(map(ay, -17000, 17000, 0, 180)); Serial.print("\n");
  #endif

  #ifdef OUTPUT_BINARY_ACCELYGYRO
  Serial.write((uint8_t)(ax >> 8)); Serial.write((uint8_t)(ax &
0xFF));
  Serial.write((uint8_t)(ay >> 8)); Serial.write((uint8_t)(ay &
0xFF));
  Serial.write((uint8_t)(az >> 8)); Serial.write((uint8_t)(az &
0xFF));

```

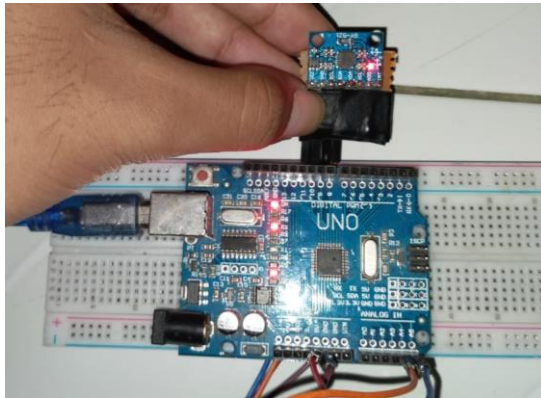
```

Serial.write((uint8_t)(gx >> 8)); Serial.write((uint8_t)(gx &
0xFF));
Serial.write((uint8_t)(gy >> 8)); Serial.write((uint8_t)(gy &
0xFF));
Serial.write((uint8_t)(gz >> 8)); Serial.write((uint8_t)(gz &
0xFF));
#endif
blinkState = !blinkState;
}

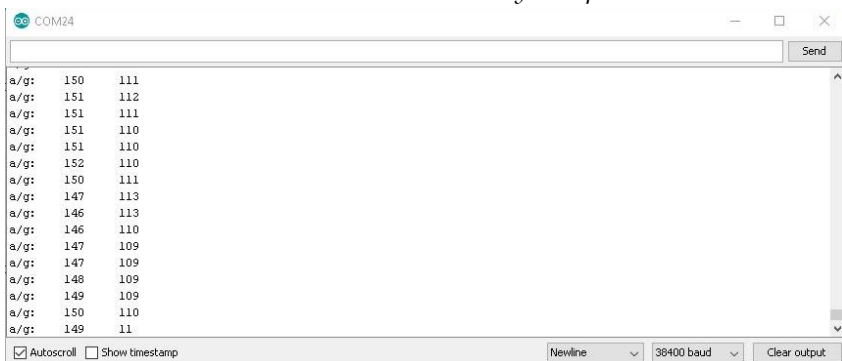
```

### Hasil Percobaan Sensor Gyroscope (MPU-6050)

Berikut hasil percobaan dari Sensor *Gyroscope* MPU-6050



**Gambar 4.31** Percobaan Sensor *Gyroscope* MPU-6050



**Gambar 4.32** Hasil Serial Monitor Sensor *Gyroscope* MPU-6050

## Sensor Gyroscope (MPU-6050) dengan Motor Servo

Pada *project* ini akan melanjutkan bahasan pada *project* sebelumnya. Jika pada bahasan sebelumnya hasil pengukuran sensor *Gyroscope* MPU-6050 ditampilkan di Serial Monitor IDE Arduino, maka pada bahasan ini akan diimplementasikan dengan motor Servo sebagai Outputnya.

### Motor Servo

Motor servo adalah sebuah perangkat atau aktuator putar (motor) yang dirancang dengan sistem kontrol umpan balik loop tertutup (servo), sehingga dapat di set-up atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer. Serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas posisi putaran poros motor servo. Bentuk fisik motor Servo seperti Gambar 13.1



Gambar 4.33 Motor Servo

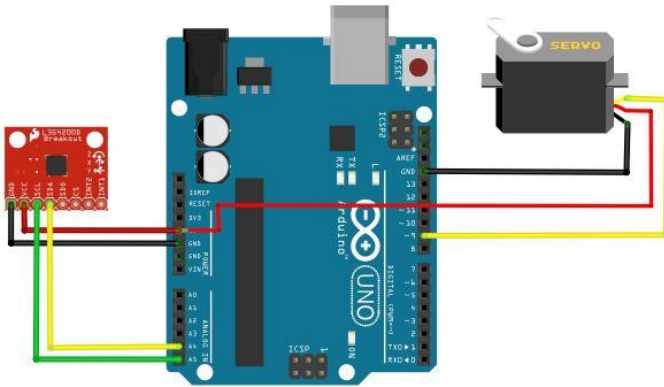
### Rangkaian Sensor Gyroscope (MPU-6050)

Sensor *Gyroscope* MPU-6050 yang akan di buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
Arduino UNO	1
Sensor <i>Gyroscope</i> MPU-6050	1
Motor Servo	1

Breadboard	1
Kabel Jumper	7

Untuk lebih jelasnya perhatikan rangkaian dari sensor *Gyroscope* MPU-6050 pada Gambar 4.34



**Gambar 4.34** Rangkaian Sensor *Gyroscope* MPU-6050

Buatlah rangkaian seperti pada **Gambar 4.34**.

- Pasangkan Sensor *Gyroscope* MPU-6050 dan ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin 5 Volt arduino dan gnd *Breadboard* ke gnd arduino.
- Pasangkan kaki SCL sensor ke pin A5 Arduino
- Pasangkan kaki SDA sensor ke pin A4 Arduino.
- Pasangkan kaki vcc dan gnd Motor Servo ke pin vcc dan gnd Arduino dan kaki tengah Motor Servo ke pin 8 Arduino.

### Program Sensor Gyroscope (MPU-6050)

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke Arduino UNO menggunakan kabel USB Downloader.

```

#include<Servo.h> //pemanggilan library dari servo
Servo myservo1; //inisialisasi servo
#include "I2Cdev.h" //pemanggilan library i2c dari sensor
mpu6050
#include "MPU6050.h" //library dari sensor mpu6050
#if I2CDEV_IMPLEMENTATION ==
I2CDEV_ARDUINO_WIRE
#include "Wire.h" //library wire dari arduino
#endif
MPU6050 accelgyro;
int16_t ax, ay, az;
int16_t gx, gy, gz;
#define OUTPUT_READABLE_ACCELGYRO
bool blinkState = false;

void setup() {
#if I2CDEV_IMPLEMENTATION ==
I2CDEV_ARDUINO_WIRE
Wire.begin();
#elif I2CDEV_IMPLEMENTATION ==
I2CDEV_BUILTIN_FASTWIRE
Fastwire::setup(400, true);
#endif
Serial.begin(38400); //mengatur baud rate ke 38400
Serial.println("Initializing I2C devices...");
accelgyro.initialize();
Serial.println("Testing device connections...");
Serial.println(accelgyro.testConnection() ? "MPU6050
connection successful" : "MPU6050 connection failed");
myservo1.attach(9); //inisialisasi pin servo
}

void loop() {
accelgyro.getMotion6(&ax, &ay, &az, &gx, &gy, &gz);
#ifdef OUTPUT_READABLE_ACCELGYRO

```

```

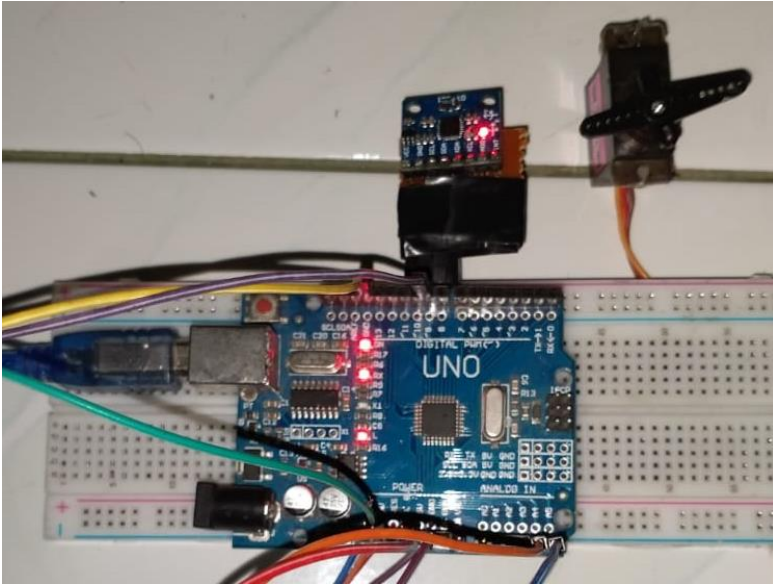
Serial.print("a/g:\t");
Serial.print(map(ax, -17000, 17000, 0, 180)); Serial.print("\t");
Serial.print(map(ay, -17000, 17000, 0, 180)); Serial.print("\n");
myservo1.write(map(ax, -17000, 17000, 0, 180)); // mengatur
derajat servo sesuai nilai dari sensor gyro
#endif

#ifdef OUTPUT_BINARY_ACCELYRO
  Serial.write((uint8_t)(ax >> 8)); Serial.write((uint8_t)(ax &
0xFF));
  Serial.write((uint8_t)(ay >> 8)); Serial.write((uint8_t)(ay &
0xFF));
  Serial.write((uint8_t)(az >> 8)); Serial.write((uint8_t)(az &
0xFF));
  Serial.write((uint8_t)(gx >> 8)); Serial.write((uint8_t)(gx &
0xFF));
  Serial.write((uint8_t)(gy >> 8)); Serial.write((uint8_t)(gy &
0xFF));
  Serial.write((uint8_t)(gz >> 8)); Serial.write((uint8_t)(gz &
0xFF));
#endif
  blinkState = !blinkState;
}

```

### Hasil Percobaan Sensor Gyroscope (MPU-6050)

Berikut hasil percobaan dari Sensor *Gyroscope* MPU-6050



Gambar 4.35 Percobaan Sensor Gyroscope MPU-6050 dengan Motor Servo

## 9. NodeMCU ESP8266

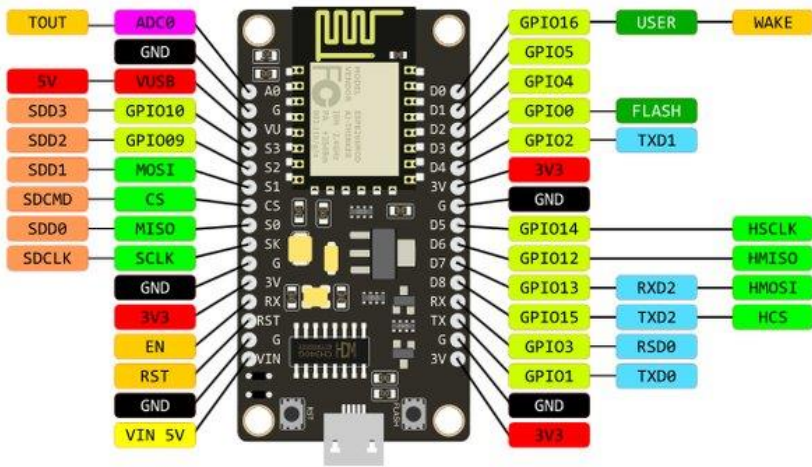
### PENGENALAN NODEMCU ESP8266

#### NODEMCU ESP8266

Sebelum nya NodeMCU ini sama halnya seperti mikrokontroler atau Arduino, hanya saja NodeMCU ini sudah terkoneksi atau dapat terhubung ke internet. Dengan arti lain NodeMCU adalah sebuah *board* elektronik yang berbasis chip ESP8266 dengan kemampuan menjalankan fungsi mikrokontroler dan juga koneksi internet (WiFi). Terdapat beberapa pin I/O sehingga dapat dikembangkan menjadi sebuah aplikasi monitoring maupun *controlling* pada proyek IOT. NodeMCU ESP8266 dapat diprogram dengan *compiler*-nya Arduino, menggunakan Arduino IDE. Bentuk fisik dari NodeMCU ESP 8266, terdapat *port* USB (mini USB) sehingga akan memudahkan dalam pemrogramannya.

NodeMCU ESP8266 merupakan modul turunan pengembangan dari modul platform IoT (Internet of Things)

keluarga ESP8266 tipe ESP-12. Secara fungsi modul ini hampir menyerupai dengan platform modul arduino, tetapi yang membedakan yaitu dikhususkan untuk “Connected to Internet”. Bentuk fisik dari NodeMCU seperti Gambar 4.36



Gambar 4.36 NodeMCU ESP8266

## APLIKASI BLYNK

**BLYNK** adalah platform untuk aplikasi OS Mobile (iOS dan Android) yang bertujuan untuk kendali module *Arduino*, *Raspberry Pi*, *ESP8266*, *WEMOS D1*, dan module sejenisnya melalui Internet. Logo Blynk dapat di lihat pada Gambar 14.2 dan dapat juga di download pada smartphone pribadi.



Gambar 4.37 Logo Blynk



Aplikasi ini merupakan wadah kreatifitas untuk membuat antarmuka grafis untuk proyek yang akan diimplementasikan hanya dengan metode drag and drop widget.

Penggunaannya sangat mudah untuk mengatur semuanya dan dapat dikerjakan dalam waktu kurang dari 5 menit. **Blynk** tidak terikat pada papan atau module tertentu. Dari platform aplikasi inilah dapat mengontrol apapun dari jarak jauh, dimanapun kita berada dan waktu kapanpun. Dengan catatan terhubung dengan internet dengan koneksi yang stabil dan inilah yang dinamakan dengan sistem Internet of Things (**IOT**)

Cara menggunakan Blynk :

Pada kesempatan kali ini saya menggunakan platform dari aplikasi ANDROID, langkah-langkah awal penggunaannya yaitu :

1. Download dan install aplikasi melalui "PlayStore"
2. Buka aplikasi, dan silahkan sign up new account atau login menggunakan "Facebook"
3. Buat new project, dan pilihlah salah satu module yang akan Anda gunakan maupun aksesoris module yang berfungsi sebagai sarana terhubung ke Internet.
4. Setelah itu drag and drop rancangan proyek Anda
5. Kemudian klik Blynk untuk mengirimkan Token Auth melalui email
6. Dan terakhir cek inbox email Anda dan temukan Auth Token yang dimana ini akan digunakan untuk program yang di downloadkan ke module.

## **PROJECT NODEMCU SP8266, DHT11 dan BLYNK SENSOR DHT11**

Sensor DHT11 merupakan sensor suhu dan kelembaban, data suhu dan kelembaban ruang dideteksi oleh sensor DHT11. Keluaran sensor DHT11 berupa sinyal digital yang sudah terkalibrasi. Jangkauan pengukuran temperatur dari sensor ini adalah 0-50°C dan jangkauan pengukuran kelembaban relatif sebesar 20-90%. Sensor DHT11 memiliki 3 pin yaitu vcc, data, dan gnd. Sensor DHT11 membutuhkan catu daya sebesar 3.3 sampai 5

Volt DC. Keakuratan untuk kelembaban relatifnya sebesar  $\pm 4\%$  dan keakuratan untuk temperatur sebesar  $\pm 2^{\circ}\text{C}$ . Bentuk fisik dari sensor DHT11 ditunjukkan pada Gambar 15.1.



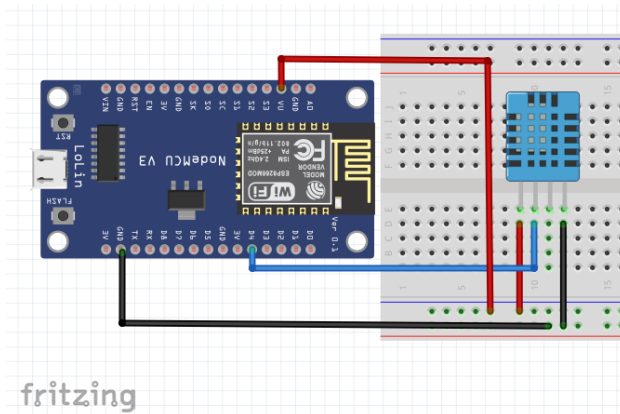
**Gambar 4.37** Sensor DHT11

### Rangkaian Sensor DHT11 dengan NodeMCU8266

Sensor DHT11 yang akan kita buat terdiri dari beberapa komponen diantaranya yaitu:

Komponen	Jumlah
NodeMCU8266	1
Sensor DHT11	1
<i>Breadboard</i>	1
Kabel Jumper	5

Untuk lebih jelasnya perhatikan rangkaian dari sensor DHT11 pada Gambar 4.38



**Gambar 4.38** Rangkaian sensor DHT11 dengan NodeMCU8266

Buatlah rangkaian seperti pada **Gambar 4.38**.

- Pasangkan Sensor DHT11 ke *Breadboard*
- Pasangkan kaki vcc sensor dan gnd sensor ke *Breadboard*, dan vcc *Breadboard* ke pin VU NodeMCU8266 dan gnd *Breadboard* ke gnd NodeMCU8266.
- Pasangkan kaki out sensor DHT11 ke pin D4 NodeMCU8266.

### Program Sensor DHT11 NodeMCU8266

Masukkan program berikut ini kedalam aplikasi IDE Arduino kemudian upload ke NodeMCU8266 menggunakan kabel USB Downloader.

```
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);
char auth[] = "TOKEN";
char ssid[] = "Nama wifi";
char pass[] = "Password wifi";
void setup()
{
  Blynk.begin(auth, ssid, pass);
```

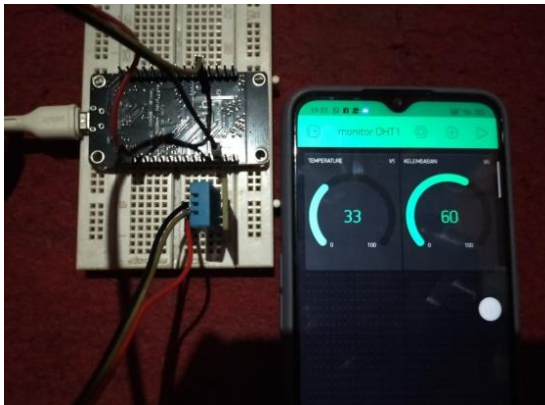
```

Serial.begin(9600);
Serial.println(F("DHTxx test!"));
dht.begin();
}
void baca()
{
float h = dht.readHumidity();
float t = dht.readTemperature();
float f = dht.readTemperature(true);
if (isnan(h) || isnan(t) || isnan(f)) {
Serial.println(F("Failed to read from DHT sensor!"));
return;
}
float hif = dht.computeHeatIndex(f, h);
float hic = dht.computeHeatIndex(t, h, false);
Blynk.virtualWrite(V0, h);
Blynk.virtualWrite(V1, t);
}
void loop()
{
Blynk.run();
baca();
}

```

### Hasil Percobaan

Berikut hasil percobaan dari sensor DHT11 dengan NodeMCU8266 menggunakan Aplikasi BLYNK.



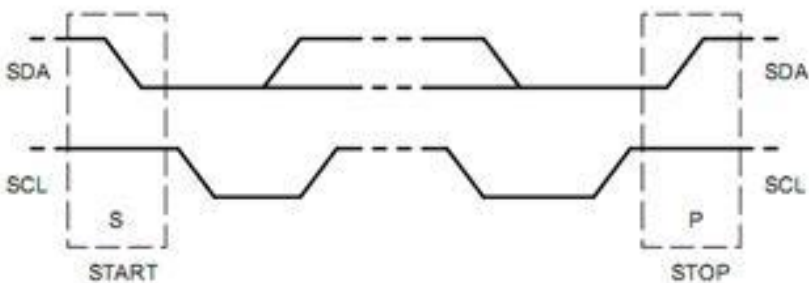
**Gambar 4.39** Hasil Percobaan dengan Blynk

# BAB 5

## INTERCAFE SERIAL I2C

### 1. Pengenalan I2C

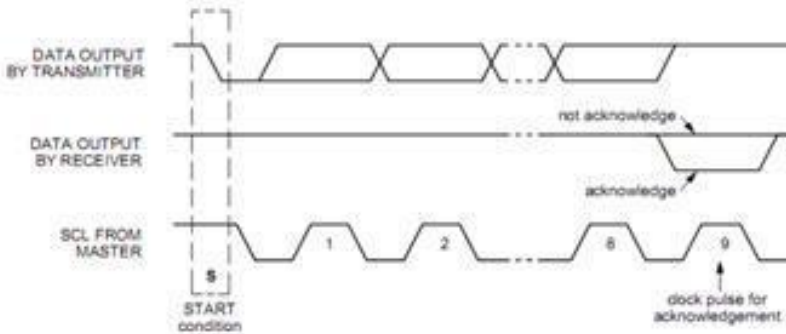
Inter Integrated Circuit atau sering disebut I2C adalah standar komunikasi serial dua arah menggunakan dua saluran yang didisain khusus untuk mengirim maupun menerima data. Sistem I2C terdiri dari saluran SCL (Serial Clock) dan SDA (Serial Data) yang membawa informasi data antara I2C dengan pengontrolnya. Piranti yang dihubungkan dengan sistem I2C Bus dapat dioperasikan sebagai Master dan Slave. Master adalah piranti yang memulai transfer data pada I2C Bus dengan membentuk sinyal Start, mengakhiri transfer data dengan membentuk sinyal Stop, dan membangkitkan sinyal clock. Slave adalah piranti yang dialamati master.



Gambar 5.1: Kondisi sinyal start dan stop

Sinyal Start merupakan sinyal untuk memulai semua perintah, didefinisikan sebagai perubahan tegangan SDA dari 1 menjadi 0 pada saat SCL 1. Sinyal Stop merupakan sinyal untuk mengakhiri semua perintah, didefinisikan sebagai perubahan

tegangan SDA dari 0 menjadi 1 pada saat SCL 1. Kondisi sinyal Start dan sinyal Stop seperti tampak pada Gambar 9.1

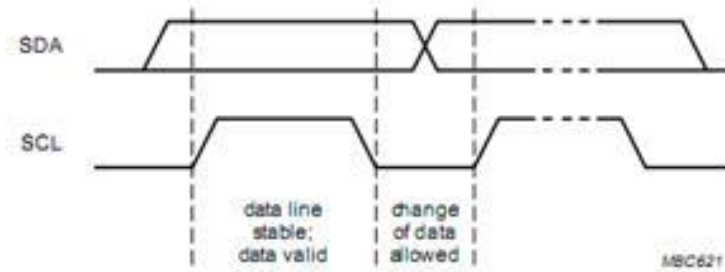


Gambar 5.2: Sinyal ACK dan NACK

Sinyal dasar yang lain dalam I2C Bus adalah sinyal acknowledge yang disimbolkan dengan ACK Setelah transfer data oleh master berhasil diterima slave, slave akan menjawabnya dengan mengirim sinyal acknowledge, yaitu dengan membuat SDA menjadi 0 selama siklus clock ke 9. Ini menunjukkan bahwa Slave telah menerima 8 bit data dari Master. Kondisi sinyal acknowledge seperti tampak pada Gambar 5.2

Dalam melakukan transfer data pada I2C Bus, kita harus mengikuti tata cara yang telah ditetapkan yaitu:

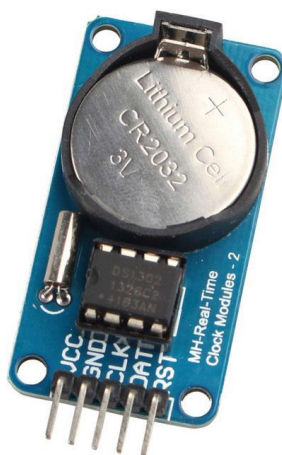
- Transfer data hanya dapat dilakukan ketika Bus tidak dalam keadaan sibuk.
- Selama proses transfer data, keadaan data pada SDA harus stabil selama SCL dalam keadaan tinggi. Keadaan perubahan 1 atau 0 pada SDA hanya dapat dilakukan selama SCL dalam keadaan rendah. Jika terjadi perubahan keadaan SDA pada saat SCL dalam keadaan tinggi, maka perubahan itu dianggap sebagai sinyal Start atau sinyal Stop.



Gambar 5.3: Trasfer Bit pada I2C bus

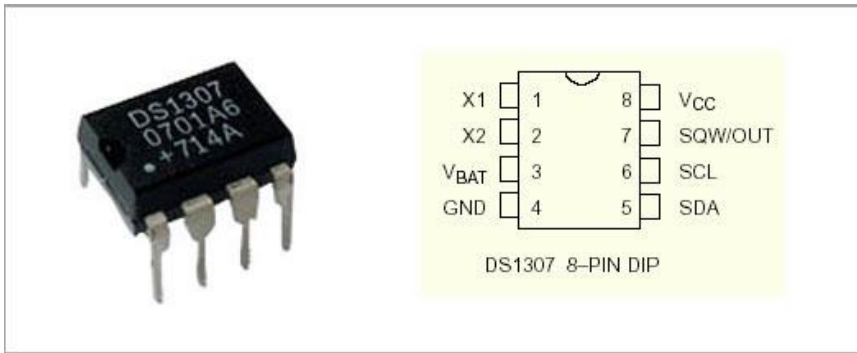
## 2. RTC sebagai Interface I2C

Real Time Clock merupakan suatu IC yang memiliki fungsi sebagai penyimpan waktu dan tanggal. RTC (Real time clock) adalah jam elektronik berupa chip yang dapat menghitung waktu dan menyimpan data waktu tersebut secara real time. RTC dilengkapi dengan baterai sebagai penyedia daya pada IC. RTC cukup akurat sebagai pewaktu (timer) karena menggunakan osilator kristal Beberapa. contoh IC RTC yang ada di pasaran seperti D812C887, DS1307, DS1302, DS3234. Salah satu IC RTC yang banyak digunakan yaitu adalah DS1307. DS1307 merupakan Real-time clock (RTC) yang memiliki antarmuka serial Two-wire 12C (Inter Integrated Circuit). Beberapa keistimewaan RTC.



Gambar 5.4: Modul Real Time Clock (RTC)

DS1307 yaitu sinyal keluaran berupa gelombangkotak terprogram (Programmable squarewave), deteksi otomatis kegagalan-daya (power-fail), konsumsi daya kurang dari 500mA menggunakan mode baterai cadangan dengan operasional osilator. Tersedia untuk industri dengan ketahanan suhu: -40C hingga +85C. Tersedia dalam kemasan 8 pin DIP atau SOIC.



Gambar 5.5: IC RTC DS1307

Penjelasan dari masing-masing kaki adalah sebagai berikut:

- X1 dan X2 adalah pin yang dihubungkan dengan kristal 32.768 KHz.
- VBAT adalah pin yang dihubungkan masukan baterai +3V.
- GND adalah pin yang dihubungkan Ground.
- SDA adalah pin yang difungsikan sebagai jalur data.
- SCL adalah pin yang fungsikan sebagai jalur clock.
- SQW/OUT adalah pin yang digunakan sebagai keluaran sinyal kotak.
- VCC adalah pin untuk mencatu tegangan 5V.

DS1307 memiliki akurasi hingga tahun 2100. Sistem RTC DS1307 memerlukan baterai eksternal 3 volt yang terhubung ke pin Vbat dan ground. Pin X1 dan X2 dihubungkan dengan kristal osilator 32,768 KHz. Sedangkan pin SCL, SDA, dan SQW/OUT di-pull-up dengan resistor (nilainya 1k s.d 10k) ke Vcc. Untuk membaca data tanggal dan waktu yang tersimpan di memori RTC DS1307 dapat dilakukan melalui komunikasi serial I2C.

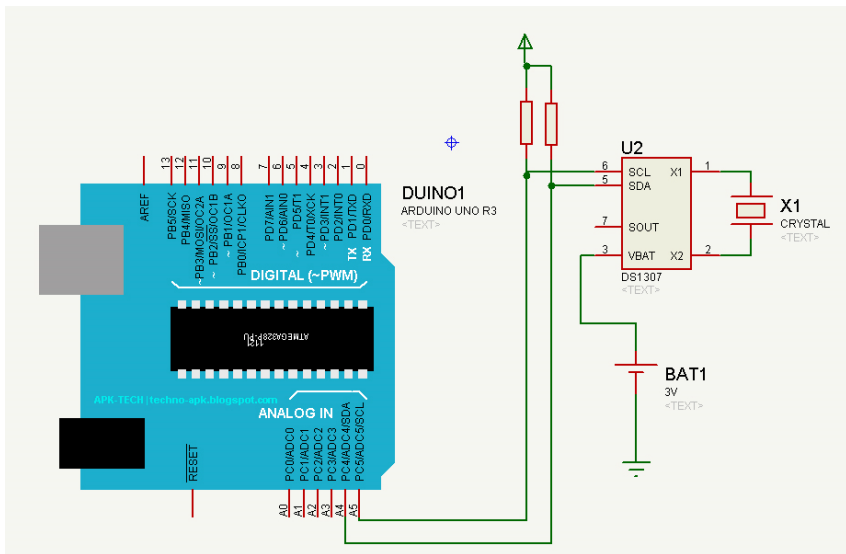


### 3. Interface I2C pada Arduino

Arduino sudah menyediakan pin dengan `_tur` SDA dan SCL, pada mikrokontroler jenis Arduino UNO pin SDA/SCL berada pada pin Analog 4 dan 5. Berikut adalah contoh pengaplikasian interface I2C pada perangkat RTC.

#### Rangkaian

Pada Gambar 5.6, Arduino UNO diberikan interface berupa RTC dan interface LCD untuk menampilkan data berupa waktu. Prinsip kerjanya adalah mikrokontroler akan melakukan request data setiap waktu ke RTC dan akan mengumpukan ke LCD untuk menampilkan waktu yang telah diterima dari RTC.



Gambar 5.6: Rangkaian Interface RTC pada Arduino UNO

#### Pemrograman

Pada Pemrograman RTC Arduino telah tersedia library "`<Wire.h>`" untuk mengaktifkan jalur komunikasi data I2C dan library "`RTCLib.h`" untuk mengaktifkan fitur RTC.

```
// Date and time functions using a DS1307 RTC
// and connected via I2C and Wire lib
#include <Wire.h>
#include "RTClib.h"

RTC_DS1307 rtc;

char daysOfTheWeek[7][12] = {
  "Sunday",
  "Monday",
  "Tuesday",
  "Wednesday",
  "Thursday",
  "Friday",
  "Saturday"
};

void setup () {
  while (!Serial); // for Leonardo/Micro/Zero

  Serial.begin(57600);
  if (! rtc.begin()) {
    Serial.println("Couldn't find RTC");
    while (1);
  }

  if (! rtc.isrunning()) {
    Serial.println("RTC is NOT running!");
    // following line sets the RTC to the date & time this sketch was compiled
    // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
    // This line sets the RTC with an explicit date & time, for example to set
    // January 21, 2014 at 3am you would call:
    // rtc.adjust(DateTime(2014, 1, 21, 3, 0, 0));
  }
}

void loop () {
  DateTime now = rtc.now();

  Serial.print(now.year(), DEC);
  Serial.print('/');
  Serial.print(now.month(), DEC);
  Serial.print('/');
  Serial.print(now.day(), DEC);
  Serial.print(" (");
  Serial.print(daysOfTheWeek[now.dayOfTheWeek()]);
```

# DAFTAR PUSTAKA

Andrianto, H. dan Darmawan, A. 2016. *Arduino Belajar Cepat dan Pemrograman*. Informatika Bandung, Bandung.

McRoberts Mike. 2009. *Arduino Starter Kit Manual*. Earthshine Design.

Monk, Simon. 2010. *30 Arduino Projects for the Ecil Genius*. McGraw-Hill Companies, Inc. United State.

Santoso, Hari. 2015. *Panduan Praktis Arduino untuk Pemula*.

E. Knuth, D.: 2011, *The Art of Computer Programing*, Addison Wesley, United States.

