



Modul Belajar Arduino Uno

Jati Widyo Leksono
Humaidillah K. W
Elly Indahwati
Nanndo Yanuansa
Imamatul Ummah

KATA PENGANTAR

Assalamu`alaikum Wr Wb

Rasa Syukur kami panjatkan kepada Allah SWT, atas selesainya buku yang berjudul “Modul Belajar Arduino Uno”. Penulisan buku ini merupakan hasil luaran dari pengabdian kepada masyarakat yang berjudul “Pelatihan Dasar Arduino Uno Untuk Meningkatkan Kemampuan SMK Dalam Pemrograman”. Isi buku ini menjelaskan tentang latar belakang penyusunan buku, penjelasan dasar terkait arduino uno dan beberapa kegiatan membuat *traffic light*.

Ucapan terima kasih kami sampaikan kepada

1. Dr.(HC). Ir. KH. Shalahudin Wahid, selaku Rektor Universitas Hasyim Asy’ari Tebuireng Jombang.
2. Prof. Dr. Haris Supratno, yang telah memberikan masukan dalam penyusunan buku.
3. Drs. Bambang Sujatmiko, M.T. selaku Ketua LPPM Universitas Hasyim Asy’ari Tebuireng Jombang, beserta staf.
4. Dr. Tri Rijanto, M.Pd, M.T, selaku Dekan Fakultas Teknik Universitas Hasyim Asy’ari Tebuireng Jombang, beserta staf.
5. Seluruh pihak terkait pengabdian kepada masyarakat, yang tidak dapat disebutkan satu-persatu.

Penulisan ini masih jauh dari sempurna, oleh karena itu kritik dan saran selalu kami harapkan demi kesempurnaan buku ini.

Jombang , 19 Februari 2019

Penullis

DAFTAR ISI

KATA PENGANTAR	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	iv
DAFTAR TABEL	vi
BAB I MENGENAL ARDUINO	1
A. JENIS-JENIS ARDUINO	1
B. MENGINSTAL SOFTWARE ARDUINO IDE	6
C. MENCoba <i>SKETCH</i> DI ARDUINO IDE	10
D. MEYIMPAN DAN MENGUNGGAH <i>SKETCH</i>	16
BAB II ARDUINO UNO	20
A. ARDUINO UNO	20
B. Fungsi Bagian-Bagian Pada Arduino	22
C. MACAM DAN FUNGSI INSTRUKSI BAHASA PEMROGRAMAN ARDUINO	24
D. Pemanfaatan Arduino UNO	32
BAB III KEGIATAN MEMBUAT <i>TRAFFIC LIGHT</i>	35
KEGIATAN I <i>ONE LED "ON"</i>	35
KEGIATAN II <i>FLIP -FLOP</i>	38
KEGIATAN III <i>TRAFFIC LIGHT</i>	40
KEGIATAN IV <i>TRAFFIC LIGHT WITH WARNING SYSTEM</i>	43
KEGIATAN V <i>TEXT ON LCD DISPLAY</i>	47
DAFTAR PUSTAKA	52

DAFTAR GAMBAR

Gambar 1.1. <i>Board</i> Arduino Mega2560.....	2
Gambar 1.2. <i>Board</i> Arduino Leonardo.....	3
Gambar 1.3. <i>Board</i> Arduino Nano	4
Gambar 1.4. <i>Board</i> Arduino Lilypad.....	5
Gambar 1.5. <i>Board</i> Arduino UNO	6
Gambar 1.6. Tampilan Laman Arduino IDE.....	7
Gambar 1.7. Arduino.Exe	7
Gambar 1.8. Persetujuan Instalasi	8
Gambar 1.9. Pilihan Instalasi.....	8
Gambar 1.10. Instalasi Folder	9
Gambar 1.11. Proses Instalasi.....	9
Gambar 1.12. Tampilan Awal arduino	10
Gambar 1.13. <i>Sketch di</i> Arduino IDE	10
Gambar 1.14. <i>Menu File</i>	11
Gambar 1.15. <i>Menu Edit</i>	12
Gambar 1.16. <i>Menu Sketch</i>	13
Gambar 1.17. <i>Menu Tools</i>	14
Gambar 1.18. <i>Menu Help</i>	15
Gambar 1.19. <i>Tool Bar Software</i> Arduino Ide.....	16
Gambar 1.20. Tampilan <i>Save as</i>	17
Gambar 1.21. Pengisian Nama <i>File</i>	18
Gambar 1.22. Penyesuaian <i>Board</i> Arduino.....	18
Gambar 1.23. Memilih <i>Port</i> Komunikasi Serial	19
Gambar 2.1. Bentuk Fisik Arduino UNO.....	20
Gambar 2.2. Tampilan Utama Arduino IDE.....	21
Gambar 2.3. Bagian-Bagian Pada Arduino UNO	22
Gambar 2.4. Letak Void Loop.....	28
Gambar 2.5. Penulisan pinMode dan menggunakan digitalWrite	31
Gambar 2.6. Project Board.....	32
Gambar 2.7. Robot Mobil Empat Roda	33
Gambar 2.8. Robot dengan Sesor Jarak.....	34
Gambar 2.9. Robot Menggambar	34
Gambar 3.1. Bagian-bagian LED.....	35

Gambar 3.2. Rangkaian 1 LED ke Arduino	36
Gambar 3.3. Rangkaian 2 LED ke Arduino	38
Gambar 3.4. Rangkaian 3 LED ke Arduino	40
Gambar 3.5. Contoh Buzzer.....	43
Gambar 3.6. Rangkaian 3 LED ke Arduino dan Buzzer	44
Gambar 3.7. Lcd 16 x 2 (16 kolom, 2 baris)	47
Gambar 3.8. Sambungan I2C ke LCD	49
Gambar 3.9. Rangkaian LCD 16x2 ke Arduino	49

DAFTARTABEL

Tabel 1.1 Pilihan pada Menu File	11
Tabel 1.2 Pilihan pada Menu Sketch.....	13
Tabel 1.3 Pilihan pada Menu Tools	14
Tabel 1.4 Penjelasan Tool Bar	16
Tabel 2.1 Penggunaan Symbol Pada Arduino Uno.....	28
Tabel 2.2 Penggunaan Operator Matematika Pada Arduino Uno	29

BAB I

MENGENAL ARDUINO

A. JENIS-JENIS ARDUINO

Arduino merupakan kata yang berasal dari kata Italia artinya teman yang berani. Arduino dikembangkan oleh sebuah tim yang beranggotakan orang-orang dari berbagai belahan dunia diantaranya:

1. Massimo Banzi Milano (Italia)
2. David Cuartielles Malmoe (Swedia)
3. Tom Igoe (New York USA)
4. Gianluca Martini Torini (Italia)
5. David A. Mellis Boston (USA)

Profil pembuat arduino bisa diakses melalui halaman web: <http://www.arduino.cc/playground/Main/People>. Tujuan awal dari pembuatan arduino adalah membantu siswa membuat perangkat desain dan interaksi dengan harga murah dibandingkan dengan perangkat yang lain yang tersedia pada saat itu.

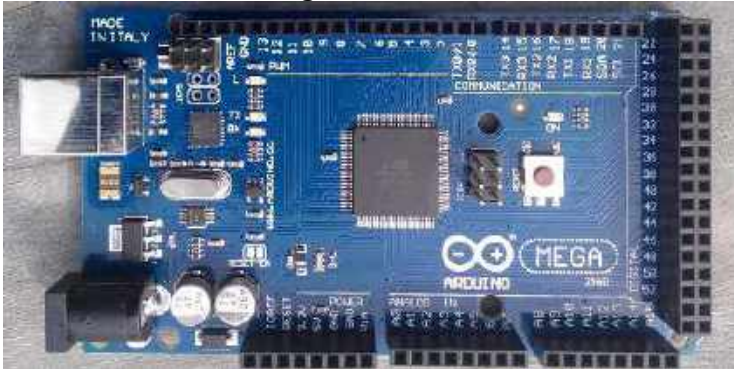
Arduino merupakan *board mikrokontroler* yang terdiri dari *hardware* dan *software* yang bersifat *open source*. *Board* arduino menggunakan IC mikrokontroler yaitu ATmega8 untuk arduino NG (Severino), ATmega 328 untuk arduino mega, nano, uno. *Software* IDE yang digunakan untuk membuat program, mengkompilasi dan mengupload program ke dalam IC ATmega. *Software* ini yang menghasilkan *file hex* dari baris kode instruksi program yang menggunakan bahasa C yang dinamakan *sketch* setelah dilakukan *compile* dengan perintah *verify/compile*. *Software* ini dapat diunduh pada halaman web: <http://www.arduino.cc>.

Macam-macam arduino yang berada dipasaran diantaranya:

1. Arduino Mega 2560

Arduino mega 2560 adalah sebuah *board* mikrokontroller yang berbasis ATmega2560. Arduino mega memiliki 54 pin input atau output yang mana 15 pin output untuk PWM, 16 analog input, 4 UARTS, sebuah crystal osilator

16 MHz, koneksi USB, *jack power*, soket ICSP dan tombol reset.
Gambar 1.1 *board* arduino Mega.



Gambar 1.1. Board Arduino Mega2560

Adapun ringkasan spesifikasi Arduino mega adalah sebagai berikut :

- | | | |
|----|---------------------------------------|--|
| a. | Mikrokontroler | : ATMEGA2560. |
| b. | Tegangan Operasi | : 5V |
| c. | Tegangan Input (<i>recommended</i>) | : 7 - 12 V |
| d. | Tegangan Input (limit) | : 6-20 V |
| e. | Pin digital I/O | : 54 (15 diantaranya pin PWM) |
| f. | Pin Analog input | : 16 |
| g. | Arus DC per pin I/O | : 40 mA |
| h. | Arus DC untuk pin 3.3 V | : 50 mA |
| i. | <i>Flash Memory</i> | : 256 KB dengan 8 KB digunakan untuk <i>Bootloader</i> |
| j. | EEPROM | : 4 KB |
| k. | SRAM | : 8 KB |
| l. | Kecepatan Pewaktuan | : 16 Mhz |

2. Arduino Leonardo

Arduino leonardo adalah sebuah papan mikrokontroler berbasis ATmega32u4. Arduino leonardo memiliki 20 digital input atau output dengan 7 pin digunakan sebagai output PWM dan 12 input analog, osilator kristal 16 MHz, koneksi micro

USB, *jack* listrik, header ICSP, dan tombol reset. Gambar 1.2 *board* arduino Leonardo.



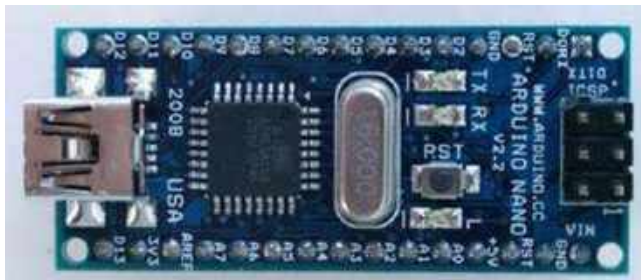
Gambar 1.2. Board Arduino Leonardo

Adapun ringkasan spesifikasi arduino leonardo adalah sebagai berikut :

- | | |
|--|---|
| a. Mikrokontroler | : ATMEGA32u4 |
| b. Tegangan Operasi | : 5V |
| c. Tegangan Input (<i>recommended</i>) | : 7 - 12 V |
| d. Tegangan Input (limit) | : 6-20 V |
| e. Pin digital I/O | : 20 (7 diantaranya pin PWM) |
| f. Pin Analog input | : 12 |
| g. Arus DC per pin I/O | : 40 mA |
| h. Arus DC untuk pin 3.3 V | : 50 mA |
| i. <i>Flash Memory</i> | : 32 KB dengan 4 KB digunakan untuk <i>bootloader</i> |
| j. EEPROM | : 1 KB |
| k. SRAM | : 2,5 KB |
| l. Kecepatan Pewaktuan | : 16 Mhz |

3. Arduino Nano

Arduino nano adalah *board* arduino terkecil dengan menggunakan mikrokontroler ATmega168 atau ATmega328. Gambar 1.3 *board* arduino nano.



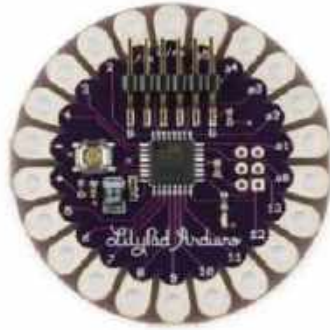
Gambar 1.3. Board Arduino Nano

Adapun ringkasan spesifikasi arduino nano adalah sebagai berikut :

- | | |
|--|---|
| a. Mikrokontroler | : ATMEGA168 atau ATMEGA328 |
| b. Tegangan Operasi | : 5V |
| c. Tegangan Input (<i>recommended</i>) | : 7 - 12 V |
| d. Tegangan Input (limit) | : 6-20 V |
| e. Pin digital I/O | : 14 (6 diantaranya pin PWM) |
| f. Pin Analog input | : 8 |
| g. Arus DC per pin I/O | : 40 mA |
| h. Arus DC untuk pin 3.3 V | : 50 mA |
| i. <i>Flash Memory</i> | : 32 KB dengan 4 KB digunakan untuk <i>bootloader</i> |
| j. EEPROM | : 512 Byte (ATMEGA168) atau 1 KB (ATMEGA328) |
| k. SRAM | : 1 KB (ATMEGA168) atau 2 KB (ATMEGA328) |
| l. Kecepatan Pewaktuan | : 16 Mhz |

4. Arduino Lilypad

Arduino Lilypad adalah sebuah *board* mikrokontroler yang berbasis ATmega32u4. Arduino lilypad memiliki 9 pin input atau output digital yang mana 4 pin sebagai output PWM dan 4 input analog. Gambar 1.4 *board* arduino lilypad.



Gambar 1.4. Board Arduino Lilypad

Adapun ringkasan spesifikasi arduino lilypad adalah sebagai berikut :

- | | |
|--|--|
| a. Mikrokontroler | : ATMEGA32u4 |
| b. Tegangan Operasi | : 3.3V |
| c. Tegangan Input (<i>recommended</i>) | : 3.8 - 5 V |
| d. Tegangan Input (limit) | : 6-20 V |
| e. Pin digital I/O | : 9 (4 diantaranya pin PWM) |
| f. Pin Analog input | : 4 |
| g. Arus DC per pin I/O | : 40 mA |
| h. Arus DC untuk pin 3.3 V | : 50 mA |
| i. <i>Flash Memory</i> | : 32 KB dengan 4 KB
digunakan untuk <i>Bootloader</i> |
| j. EEPROM | : 1 KB Byte (ATMEGA32u4) |
| k. SRAM | : 2.5 KB (ATMEGA32u4) |
| l. Kecepatan Pewaktuan | : 8 Mhz |

5. Arduino UNO

Arduino uno adalah sebuah *board* mikrokontroller yang berbasis ATmega328. Arduino uno memiliki 14 pin input atau output yang mana 6 pin dapat digunakan sebagai output PWM, 6 analog input, sebuah crystal osilator 16 MHz, koneksi USB, *jack power*, kepala ICSP, dan tombol reset. Arduino uno mampu *men-support* mikrokontroller; dapat dikoneksikan dengan komputer menggunakan kabel USB dan bisa disuplai dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk memulainya. Gambar 1.5 *board* arduino uno.



Gambar 1.5. Board Arduino UNO

Adapun ringkasan spesifikasi Arduino uno adalah sebagai berikut:

- | | |
|---------------------------------|--|
| a. Mikrokontroler | : ATMEGA328b. |
| b. Tegangan Operasi | : 5V |
| c. Tegangan Input (recommended) | : 7 - 12 V |
| d. Tegangan Input (limit) | : 6-20 V |
| e. Pin digital I/O | : 14 (6 diantaranya pin PWM) |
| f. Pin Analog input | : 6 |
| g. Arus DC per pin I/O | : 40 mA |
| h. Arus DC untuk pin 3.3 V | : 150 mA |
| i. <i>Flash Memory</i> | : 32 KB dengan 0.5KB digunakan untuk <i>bootloader</i> |
| j. EEPROM | : 1 KB |
| k. SRAM | : 2 KB |
| l. Kecepatan Pewaktuan | : 16 Mhz |

B. MENGINSTAL SOFTWARE ARDUINO IDE

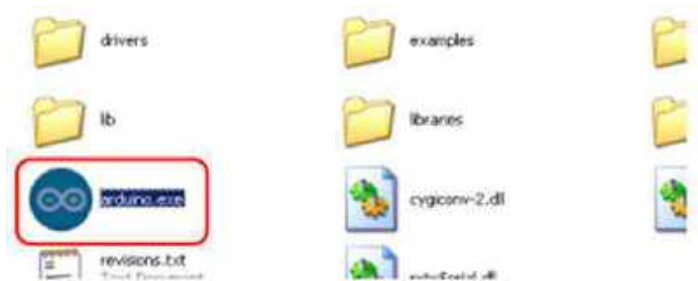
Software yang digunakan untuk mengembangkan dan memprogram ke dalam *board* arduino dinamakan arduino IDE. Arduino IDE dapat diunduh secara gratis pada alamat URL berikut: <http://arduino.cc/en/Main/Software>. *Software* ini dapat digunakan pada OS Windows, Mac OS dan Linux. Tampilan laman untuk mengunduh *software* arduino IDE seperti pada Gambar 1.6.



Gambar 1.6. Tampilan Laman Arduino IDE

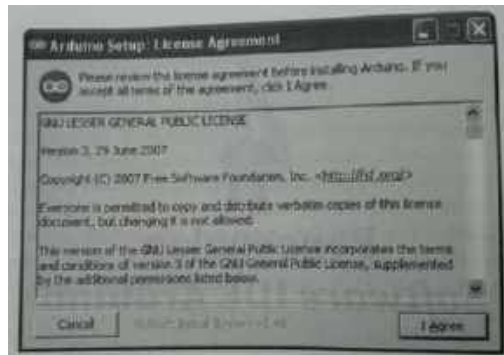
Berikut langkah-langkah menginstal *software* arduino IDE:

1. Setelah mengunduh *software* arduino IDE yang sesuai dengan OS yang digunakan, unzip arduino IDE hasil unduhan dengan menggunakan aplikasi winzip atau aplikasi ekstraksi lainnya.
2. Klik dua kali *file arduino.exe* pada hasil ekstraksi seperti Gambar 1.7 berikut ini.



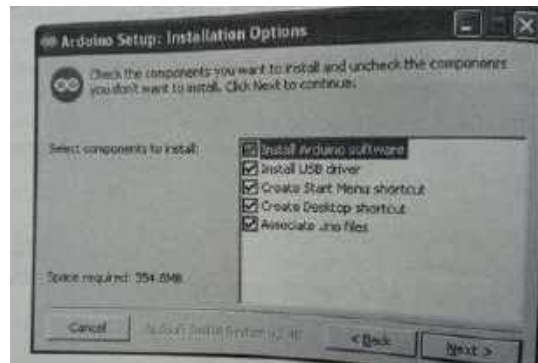
Gambar 1.7. Arduino.Exe

3. Klik tombol **I Agree** untuk melakukan persetujuan menginstal *software* arduino seperti terlihat pada Gambar 1.8.



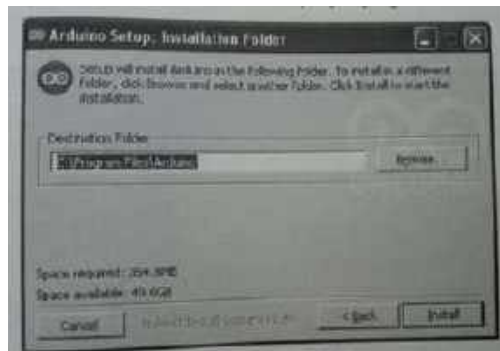
Gambar 1.8. Persetujuan Instalasi

4. Lalu muncul **Installation Option**, klik semua pilihan seperti pada Gambar 1.9.



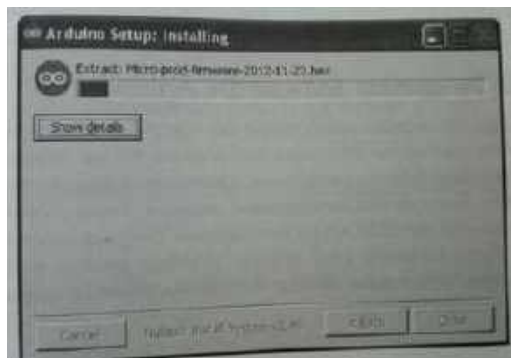
Gambar 1.9. Pilihan Instalasi

5. Klik **Next**, lalu pilih **Folder** untuk menyimpan program arduino.



Gambar 1.10. Instalasi Folder

6. Klik **Install**, untuk melakukan proses instalasi



Gambar 1.11. Proses Instalasi

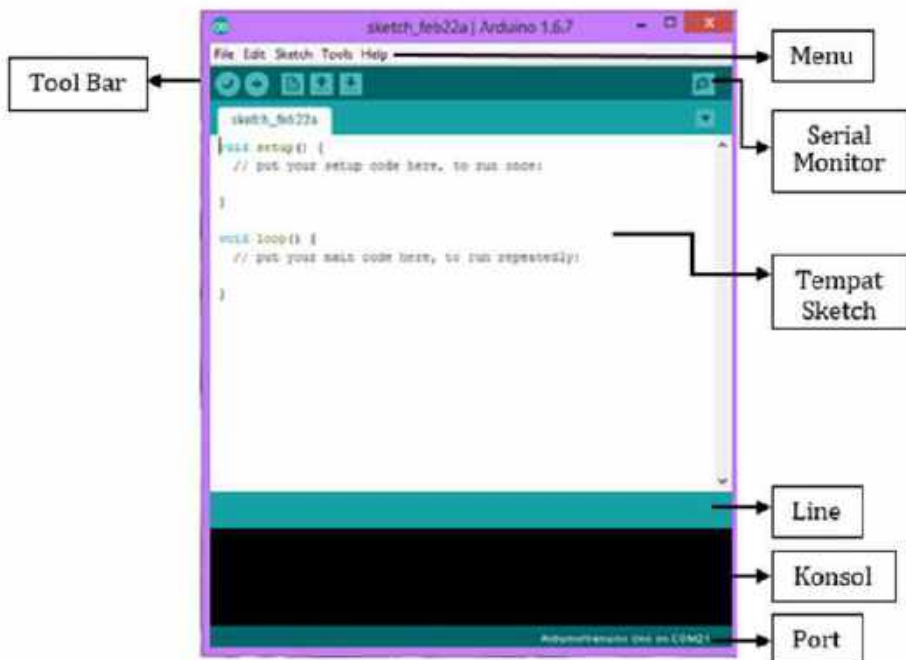
7. Klik icon **arduino** yang ada di desktop, maka tampilan awal seperti pada Gambar 1.12 di bawah ini.



Gambar 1.12. Tampilan Awal arduino

C. MENCOBA *SKETCH* DI ARDUINO IDE

Tampilan awal *sketch* di arduino IDE diperlihatkan di Gambar 1.13.

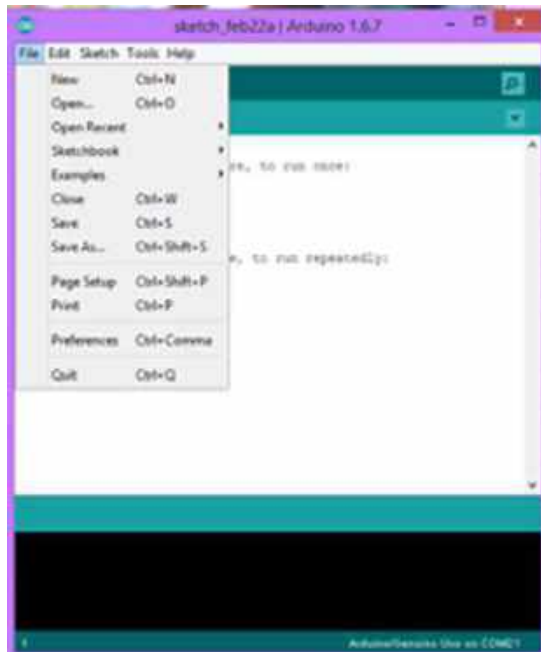


Gambar 1.13. Sketch di Arduino IDE

Beberapa tombol menu yang terdapat pada *software* arduino IDE pada Gambar 1.13 adalah *File*, *Edit*, *Sketch*, *Tools* dan *Help*.

1. *File*

Pilihan pada menu *File* seperti pada Gambar 1.14.



Gambar 1.14. Menu File

Bagian-bagian menu *file* dan penjelasannya seperti pada Tabel 1.1.

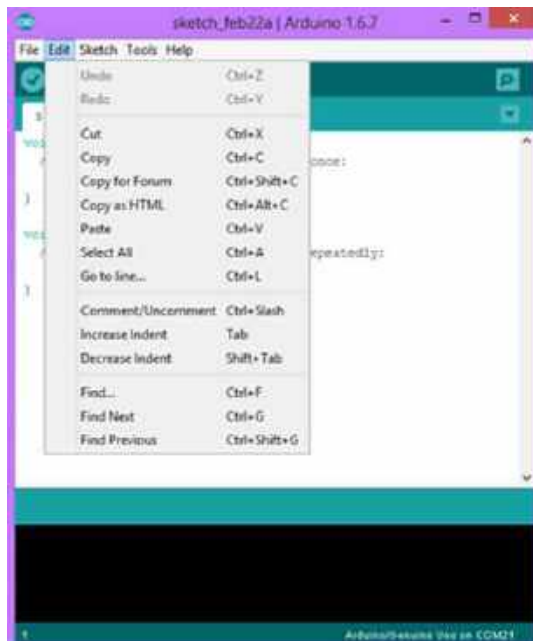
Tabel 1.1 Pilihan pada Menu File

<i>New</i>	Membuat <i>sketch</i> * baru
<i>Open</i>	Membuka <i>sketch</i> yang sudah disimpan
<i>Sketchbook</i>	Membuka <i>sketch</i> yang pernah dibuat
<i>Examples</i>	Membuka contoh-contoh <i>file sketch</i> yang berisi berbagai macam aplikasi yang

	disediakan oleh arduino
<i>Close</i>	Menutup <i>sketch</i>
<i>Save</i>	Menyimpan <i>sketch</i>
<i>Save As</i>	Menyimpan <i>sketch</i> dengan nama lain
<i>Upload to I/O Board</i>	Mengunggah program ke <i>board</i>
<i>Page Setup</i>	Mengatur ukuran halaman pada pencetak
<i>Print</i>	Mencetak <i>sketch</i>
<i>Preferences</i>	Mengatur <i>setting</i> IDE arduino
<i>Quit</i>	Keluar dari arduino

2. Edit

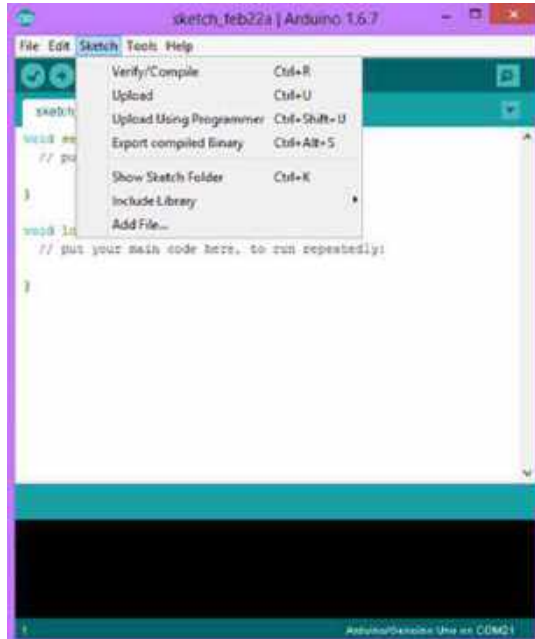
Pilihan pada menu *Edit* seperti pada Gambar 1.15.



Gambar 1.15. Menu Edit

3. Sketch

Pilihan pada menu *Sketch* seperti pada Gambar 1.16.



Gambar 1.16. Menu Sketch

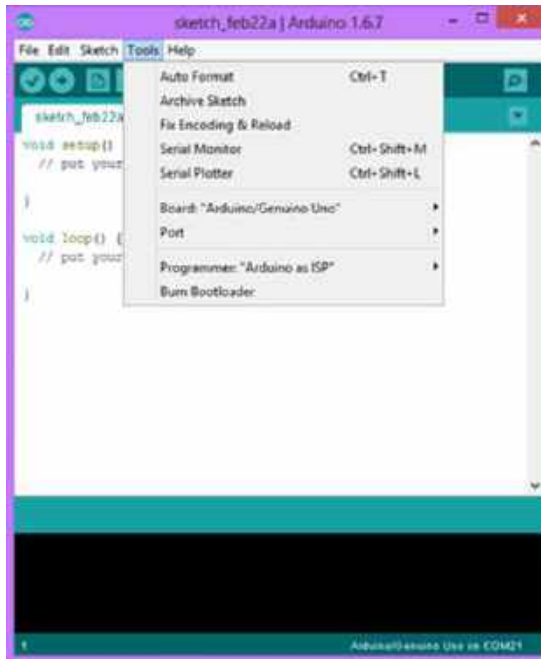
Bagian-bagian menu *sketch* dan penjelasannya seperti pada Tabel 1.2.

Tabel 1.2 Pilihan pada Menu Sketch

<i>Verify/Compile</i>	Mengompilasi program
<i>Stop</i>	Menghentikan kompilasi (apabila 'Hang')
<i>Show sketch folder</i>	Menampilkan <i>folder</i> dari <i>sketch</i> yang sedang dibuka
<i>Import Library</i>	Mengambil <i>header</i> librari dari fungsi-fungsi tambahan
<i>Add File</i>	Menambah buka <i>file sketch</i> pada jendela yang sama

4. Tools

Pilihan pada menu *Tools* seperti pada Gambar 1.17.



Gambar 1.17. Menu

Bagian-bagian menu *tools* dan penjelasannya seperti pada Tabel 1.3.

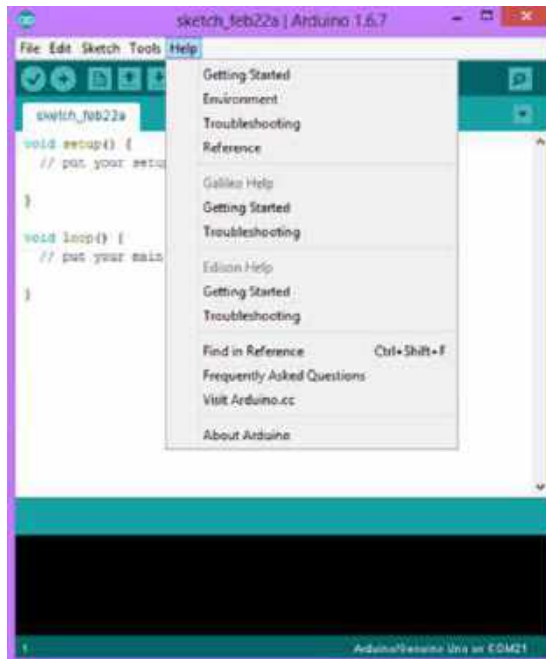
Tabel 1.3 Pilihan pada Menu Tools

<i>Auto Format</i>	Mengatur format <i>sketch</i> secara otomatis
<i>Archive Sketch</i>	Menyimpan <i>sketch</i> dalam bentuk <i>zip file</i> (kompresi)
<i>Fix Encoding & Reload</i>	Membatalkan perubahan <i>sketch</i> sebelumnya yang telah disimpan
<i>Serial Monitor</i>	Mengaktifkan jendela tampilan komunikasi serial komputer

<i>Board</i>	Menentukan jenis <i>board</i> arduino yang digunakan
<i>Serial Port</i>	Menentukan <i>port serial</i> yang digunakan untuk mengunggah program dan tersambung pada <i>board</i> arduino
<i>Burn Bootloader</i>	Memasukkan <i>bootloader</i> pada mikrokontroler yang ada pada <i>board</i> arduino melalui ICSP

5. Help

Pilihan pada menu *Help* seperti pada Gambar 1.18.



Gambar 1.18. Menu *Help*

Beberapa tombol *tool bar* yang terdapat pada *software* arduino IDE pada Gambar 1.19 adalah *Verify*, *Upload*, *New*, *Open*, *Save*, *Serial Monitor*.



Gambar 1.19. Tool Bar Software

Arduino Ide

Bagian-bagian *tool bar* dan penjelasannya seperti pada Tabel 1.4.

Tabel 1.4 Penjelasan Tool Bar

<i>Verify</i>	Untuk mengkompilasi program artinya mengkonversi program pada arduino menjadi informasi/data yang dapat dieksekusi/dibaca oleh mikrokontroler
<i>Upload</i>	Untuk meng-unggah program ke dalam <i>Board</i> Arduino
<i>New</i>	Untuk membuat <i>file sketch</i> baru
<i>Open</i>	Untuk membuka <i>file sketch</i> yang sudah pernah dibuat
<i>Save</i>	Untuk menyimpan <i>sketch (list program)</i> yang sedang dibuat
<i>Serial Monitor</i>	Untuk mengaktifkan jendela komunikasi serial, dan transfer data (kirim/terima) antara <i>Board Arduino</i> dan komputer.

D. MEYIMPAN DAN MENGUNGGAH *SKETCH*

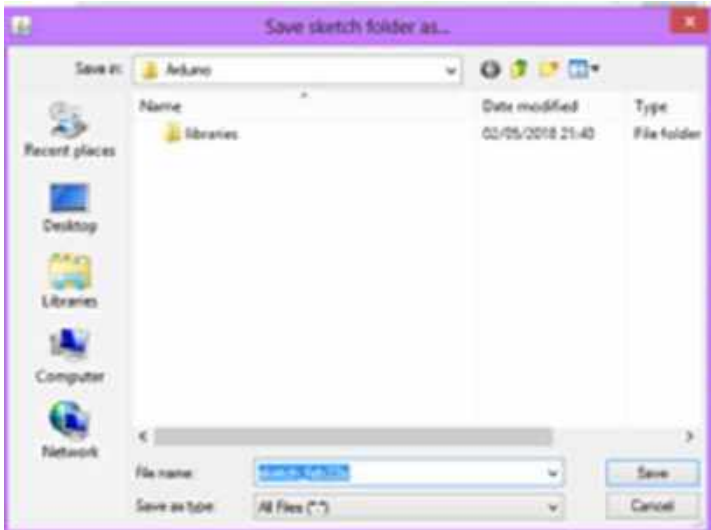
Sketch yang baru saja ditulis dapat disimpan ke dalam file dengan cara sebagai berikut.

1. Klik pada menu **File**.

2. Klik pada **Save As...** pada Gambar 1.20 untuk menyimpan. Langkah ini menampilkan kotak dialog berjudul “*Save sketch folder as ...*” seperti pada Gambar1.21.

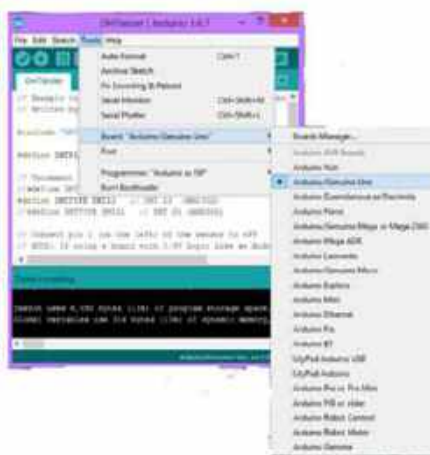


Gambar 1.20. Tampilan Save as



Gambar 1.21. Pengisian Nama File

3. Pilih folder arduino di bagian **Save in**.
4. Ketikkan **nama file** di kotak edit berlabel **File name**.
5. Klik tombol **Save** untuk menyimpan.
6. Untuk mengunggah program yang dibuat ke dalam *board* arduino pilih terlebih dahulu *board* arduino yang akan digunakan melalui *tool* dan *board* seperti pada Gambar 1.22.



Gambar 1.22. Penyesuaian Board Arduino

- Pilih *port* komunikasi serial (*comm port*) yang langsung tersambung pada *board* arduino dengan memilih menu *tool* dan *serialport* pada Gambar 1.23.



Gambar 1.23. Memilih Port Komunikasi Serial

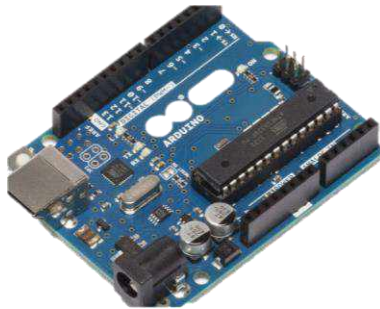
- Klik **Verify** pada *tool bar* untuk mengkompilasi kode agar dijalankan oleh *board* arduino. Hasil kompilasi berupa *binary sketch*. Tujuan dari verifikasi adalah pemeriksaan terhadap kode program yang ditulis sudah benar atau masih mengandung kesalahan.
- Apabila tidak ada kesalahan atau *error*, unggah program yang dibuat dengan memilih tombol **upload**.

BAB II

ARDUINO UNO

A. ARDUINO UNO

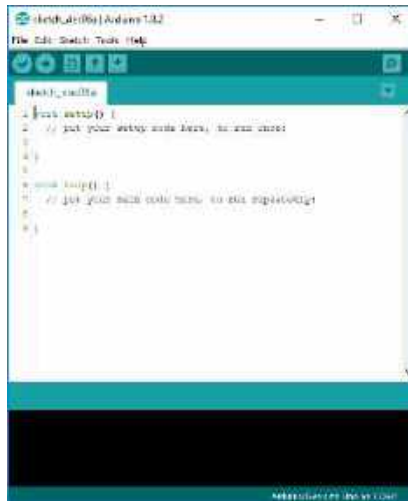
Arduino adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari *Wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware* nya memiliki prosesor Atmel AVR dan *software* nya memiliki bahasa pemrograman sendiri. Bahasa yang dipakai dalam Arduino bukan assembler yang relatif sulit, tetapi bahasa C yang disederhanakan dengan bantuan pustaka-pustaka (*libraries*) Arduino.



Gambar 2.1. Bentuk Fisik Arduino UNO

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

Arduino merupakan perangkat keras sekaligus perangkat lunak yang memungkinkan siapa saja melakukan pembuatan prototipe suatu rangkaian elektronika yang berbasis mikrokontroler dengan mudah dan cepat. Dari sisi perangkat lunak, Arduino IDE adalah *tool* yang bermanfaat untuk menuliskan program (yang secara khusus dinamakan sketsa di Arduino), mengompilasi, dan sekaligus mengunggahnya ke papan Arduino.



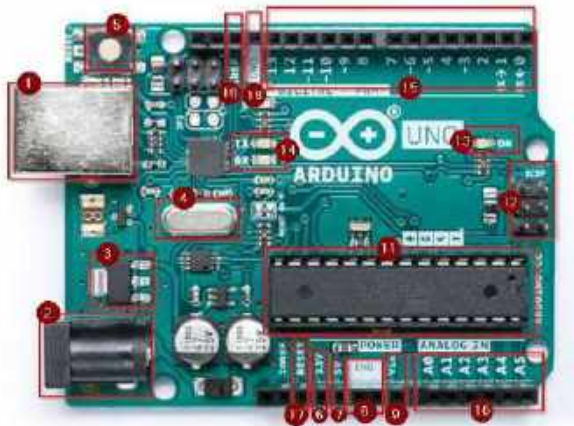
Gambar 2.2. Tampilan Utama Arduino IDE

Papan Arduino sering disebut dengan nama Arduino Uno. Arduino Uno merupakan *board* yang menggunakan mikrokontroler ATmega328P. Arduino merupakan sebuah *board* minimum sistem mikrokontroler yang bersifat *open source*. Arduino memiliki kelebihan tersendiri dibandingkan board mikrokontroler yang lain, yaitu mempunyai bahasa pemrograman sendiri yang berupa bahasa C. Selain itu, dalam *board* Arduino terdapat *loader* yang berupa USB sehingga memudahkan untuk mentransfer program dari Arduino IDE ke Arduino Uno. *Board* Arduino Uno bekerja dengan tegangan masukan 7-12V. Adapun tegangan kerja yang digunakan adalah 5V. *Board* Arduino mengandung 14 pin digital dan 6 di antara

pin-pin tersebut dapat bertindak sebagai pin-pin PWM (*Pulse Width Modulation*). Untuk mengubah pin analog menjadi digital cukup dengan mengubah konfigurasi pin pada program.

B. Fungsi Bagian-Bagian Pada Arduino

Meskipun Arduino Uno memiliki ukuran yang tidak terlalu besar, namun pada *board* tersebut terdapat mikrokontroler dan sejumlah *input-output* (I/O) yang memudahkan pemakai dalam menggunakannya.



Gambar 2.3. Bagian-Bagian Pada Arduino UNO

Bagian-bagian dari Arduino UNO seperti pada gambar 2.3 yang perlu diketahui adalah sebagai berikut:

1. Port USB tipe B
Port ini digunakan untuk sumber Arduino dari USB, digunakan untuk upload kode program dari PC ke Arduino, dan untuk komunikasi serial dari Arduino ke PC. Tipe kabel yang digunakan adalah USB type A to type B.
2. Jack Power
Jack Power ini digunakan untuk memberi tegangan arduino menggunakan arus DC dari baterai maupun power adaptor/rectifier. Tegangan kerja yang digunakan oleh Arduino Uno adalah 5V. Dengan Tegangan input dari 7V – 12V. Sedangkan limitnya adalah 6V – 20V.

3. **Regulator Tegangan**
Fungsi dari komponen ini adalah untuk mengontrol tegangan yang masuk ke papan Arduino dan menstabilkan tegangan DC yang digunakan oleh seluruh komponen yang ada di papan Arduino.
4. **Crytal Oscillator**
Komponen ini menghasilkan Frekuensi kerja Arduino. Terdapat nomor 16.000H9H yang dicetak di atas komponen, ini berarti frekuensinya adalah 16.000.000 Hertz atau 16 MHz.
5. **17 Reset**
Kita bisa me-reset papan Arduino untuk menjalankan program dari awal. Kita bisa me-reset Arduino Uno dengan 2 cara. Yang pertama adalah dengan menggunakan tombol Reset (5) yang berada di papan. Kedua adalah dengan menghubungkan tombol reset eksternal ke Pin Arduino yang berlabel Reset (17)
6. **Pin 3.3V**
Mengeluarkan tegangan output sebesar 3.3V.
7. **Pin 5V**
Mengeluarkan tegangan output sebesar 5V. Pin 3.3V dan 5V sering digunakan untuk mencatu komponen eksternal yang digunakan Arduino, seperti LCD.
8. **18 GND**
Merupakan pin yang digunakan untuk meng-ground-kan sirkuit.
9. **Vin**
Digunakan untuk mencatu arduino dari sumber power eksternal.
10. **Pin analog**
Arduino Uno memiliki 6 pin input analog, dari A0 sampai A5. Pin ini digunakan untuk membaca sinyal dari sensor analog seperti sensor kelembapan atau sensor suhu lalu mengubahnya ke digital sehingga bisa dibaca oleh mikroprosesor.
11. **Mikrokontroler**
Setiap Arduino memiliki mikrokontrolernya masing-masing. Komponen ini merupakan otaknya papan Arduino.

Mikrokontroler yang digunakan merupakan buatan dari ATMEL.

12. Pin ICSP
Biasanya digunakan untuk memprogram mikrokontroler tanpa melalui bootloader. Pin ini jarang sekali digunakan meskipun tersedia.
13. LED Indikator Power
LED ini akan menyala ketika kita menghubungkan sumber power ke papan Arduino. LED ini menandakan bahwa papan Arduino telah dicatu dengan benar. Jika LED tidak menyala maka bisa jadi terdapat masalah pada power.
14. LED TX dan RX
Pada papan yang digunakan, terdapat dua label yaitu TX (transmit) dan RX (receive). Label tersebut muncul di 2 tempat, pertama di pin digital 0 dan 1 yang digunakan untuk komunikasi serial, kedua di TX dan RX led (14). TX LED akan berkedip dengan kecepatan yang berbeda ketika mengirimkan data serial. Kecepatan berkedip bergantung pada baud rate yang digunakan. RX LED berkedip ketika proses penerimaan.
15. Pin Digital Input/Output
Papan Arduino memiliki 14 pin input output digital, 6 diantaranya dapat digunakan sebagai PWM output. Pin ini dapat dikonfigurasi agar bekerja sebagai pin input digital untuk membaca nilai logika 0 dan 1 atau sebagai output digital untuk menjalankan beberapa komponen seperti LED, Relay, dll. Pin yang terdapat label “~” merupakan pin yang dapat digunakan sebagai PWM.
16. AREF
Sebagai Analog Reference. Digunakan untuk menyetel tegangan reference eksternal (bernilai 0 sampai 5V) sebagai batas atas untuk pin input analog. Menggunakannya dengan fungsi analog Reference().

C. MACAM DAN FUNGSI INSTRUKSI BAHASA PEMROGRAMAN ARDUINO

Suatu program Arduino umumnya terdiri atas instruksi void setup () dan void loop (). Instruksi void setup() digunakan untuk menginisialisasi variabel-variabel yang akan digunakan, dan

hanya dijalankan satu kali saat Arduino mulai menyala. Sedangkan instruksi `void loop()` digunakan untuk menjalankan suatu siklus program, yang akan dilakukan terus-menerus hingga Arduino mati/reset. Berikut ini adalah beberapa fungsi dasar pada Arduino.

1. `pinMode(pin, SET)`

Fungsi ini digunakan untuk menginisialisasi sebuah pin, dan menentukan pin tersebut akan digunakan sebagai input ataupun output. Nilai SET dapat diisi OUTPUT atau INPUT, tergantung dari kebutuhan. Sedangkan nilai pin adalah nomor pin pada mikrokontroler yang akan diset sebagai input atau output. Contoh: `pinMode(13, OUTPUT)` artinya kita menentukan pin digital 13 pada Arduino berfungsi sebagai output.

2. `digitalWrite(pin, VAL)`

Fungsi ini digunakan untuk menuliskan nilai secara digital pada suatu pin. Nilai VAL dapat berupa HIGH (ON) atau LOW (OFF) dan nilai pin adalah nomor pin pada Arduino yang akan diset. Contoh: `digitalWrite(13, HIGH)` artinya pin digital 13 diset pada kondisi menyala.

3. `digitalRead(pin)`

Fungsi ini digunakan untuk membaca nilai input/masukan yang diberikan ke Arduino. Nilai yang terbaca oleh Arduino melalui `digitalRead()` bergantung pada voltase pada pin yang diatur. Kebergantungan pada nilai voltase ini disebut *Logic Level*. Pada Arduino, batasan nilai yang mencukupi untuk mencapai HIGH adalah di antara 5-3 volt, sedangkan batasan nilai yang mencapai nilai LOW adalah di antara 0 1,5 volt. Contoh: `digitalRead(13)` artinya Arduino akan membaca input yang diberikan melalui pin 13, hasilnya HIGH atau LOW.

4. `analogWrite(pin, VAL)`

Fungsi `analogWrite()` adalah fungsi yang digunakan untuk menuliskan suatu nilai berupa angka pada sebuah komponen, misalnya LED. Pengguna dapat mengatur seberapa terang cahaya dari lampu LED saat menyala, tergantung pada nilai yang dituliskan. Fungsi ini akan berguna ketika kita mulai bermain dengan sensor, di mana nilai yang terbaca seringkali berupa analog (memiliki banyak nilai, misal 0-1023), bukan

digital (hanya memiliki 2 nilai, 0 (LOW) dan 1(HIGH)). Contoh: `analogWrite(13, 1023)` artinya Arduino akan memberikan instruksi nilai maksimal pada pin 13, sehingga komponen yang terkoneksi di pin 13 akan menyala maksimal (1023).

5. `analogRead(pin)`
Fungsi ini mirip dengan fungsi `digitalRead()`, yaitu membaca nilai masukan pada suatu pin. Bedanya adalah fungsi `analogRead()` akan menghasilkan nilai dari 0 hingga 1023, yang merepresentasikan voltase 0 v hingga 5 v. Contoh: `analogRead(0)` artinya Arduino akan membaca nilai input dari pin 0.
6. `delay(time)`
Fungsi ini digunakan untuk memberikan jeda antar fungsi. Nilai time adalah waktu lamanya jeda dalam satuan ms (milisekon), di mana 1 detik setara dengan 1.000 milisekon.
7. `Serial.begin(baudrate)`
Pengguna dapat melakukan komunikasi serial antara Arduino dengan PC, dengan menggunakan Serial Monitor yang disediakan pada Arduino IDE. Pada Serial Monitor, kita bisa melihat data yang dikirim dari Arduino ke PC. Selain itu, kita juga bisa mengirim data ke Arduino dengan cara mengetikkannya pada textbox di bagian atas Arduino IDE. Untuk memakai serial, yang pertama harus dilakukan adalah melakukan inisiasi dengan menggunakan fungsi `Serial.begin(baudrate)`. Variabel baudrate merupakan rasio modulasi, dan harus dicocokkan dengan baudrate hardware yang akan dikomunikasikan. Contoh: `Serial.begin(9600)` artinya komunikasi akan berjalan pada rasio modulasi/baudrate 9600.
8. `Serial.available()`
Fungsi ini digunakan untuk mengetes apakah ada input data dari hardware yang disambungkan ke serial port, misalnya dari PC. Fungsi ini akan menghasilkan 1 apabila ada masukan, dan 0 apabila tidak ada masukan.
9. `Serial.read()`
Fungsi ini berfungsi untuk membaca karakter pada serial port. Karakter yang dibaca akan disimpan dalam bentuk ASCII (misalnya karakter '0' memiliki representasi ASCII yaitu 48).

10. Serial.print() dan Serial.println()

Fungsi ini digunakan untuk menuliskan suatu kalimat ke Serial Monitor, tetapi tidak mengirimkan data apapun, alias hanya digunakan untuk memberikan teks visual pada pengguna. Serial.print("text") digunakan untuk menulis "text", sedangkan Serial.println("text") dipakai untuk menuliskan kata "text" dan diakhiri dengan enter (kalimat selanjutnya ada di baris berikutnya).

11. Serial.write(VAL)

Untuk mengirimkan data dari arduino ke PC, kita bisa menggunakan fungsi Serial.write(VAL). Nilai VAL adalah data yang ingin dikirimkan dari arduino ke PC, dengan ukuran 1 byte.

Bahasa pemrograman Arduino merupakan salah satu bahasa pemrograman yang menggunakan bahasa C, dan ketika sebuah perintah atau coding sedang dibuat maka hasilnya akan langsung terlihat di arduino tersebut. Berikut adalah fungsi serta intruksi pada bahasa pemrograman arduino.

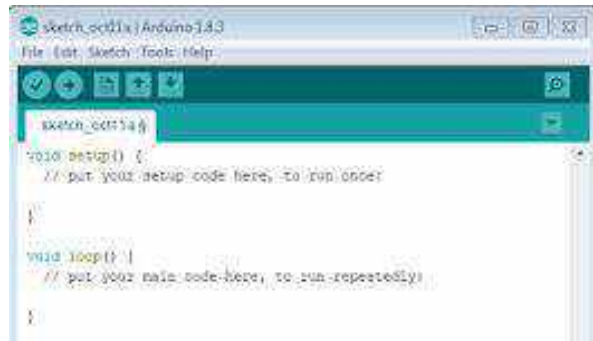
1. Struktur dari Pemograman Arduino

a. Void Setup

Tempat untuk menuliskan kode atau perintah kita di sini untuk bisa di jalan kan hanya sekali

b. Void Loop

Perintah void loop ini tempat di mana menulis kode untuk untuk menjalankan program itu berulang kali agar tidak berhenti sendiri, seperti gambar di bawah ini.



Gambar 2.4. Letak Void Loop

c. Variabel

Sebuah program membutuhkan variabel untuk mendefinisikan suatu perintah untuk memindahkan angka.

Tabel 2.1 Penggunaan Symbol Pada Arduino Uno

Simbol/Variabel	Keterangan
//	Diperlukan untuk memberi catatan. Cukup menuliskan dua buah garis miring dan apapun yang kita tulis dibelakangnya akan diabaikan oleh program. Variabel ini digunakan hanya untuk menuliskan komentar satu baris.
/* */	Diperlukan untuk menuliskan banyak catatan, maka hal itu dapat dituliskan pada beberapa baris sebagai komentar. Semua hal yang terletak di antara dua simbol tersebut akan diabaikan oleh program.
{ }	Digunakan untuk mendefinisikan kapan blok program mulai dan berakhir (digunakan juga pada

	fungsi dan pengulangan).
;	Setiap baris kode harus diakhiri dengan tanda titik koma (jika ada titik koma yang hilang maka program tidak akan bisa dijalankan)
int	<i>(integer)</i> Digunakan untuk menyimpan angka dalam 2 byte (16 bit). Tidak mempunyai angka desimal dan menyimpan nilai dari -32,768 dan 32,767.
long	<i>(long)</i> Digunakan ketika integer tidak mencukupi lagi. Memakai 4 byte (32 bit) dari memori (RAM) dan mempunyai rentang dari -2,147,483,648 dan 2,147,483,647.
boolean	<i>(boolean)</i> Variabel sederhana yang digunakan untuk menyimpan nilai TRUE (benar) atau FALSE (salah). Sangat berguna karena hanya menggunakan 1 bit dari RAM.
float	<i>(float)</i> Digunakan untuk angka desimal (floating point). Memakai 4 byte (32 bit) dari RAM dan mempunyai rentang dari -3.4028235E+38 dan 3.4028235E+38.
char	<i>(character)</i> Menyimpan 1 karakter menggunakan kode ASCII (misalnya 'A' = 65). Hanya memakai 1 byte (8 bit) dari RAM.

d. Operator Matematika dan Operator Perbandingan

Tabel 2.2 Penggunaan Operator Matematika Pada Arduino

Uno

Operator Matematika	Keterangan
=	Membuat sesuatu menjadi sama dengan nilai yang lain.

%	Menghasilkan sisa dari hasil pembagian suatu angka dengan angka yang lain (misalnya: 12 % 10, ini akan menghasilkan angka 2).
+	Digunakan sebagai perintah penjumlah
-	Digunakan sebagai perintah pengurang
*	Digunakan sebagai perintah pengali
/	Digunakan sebagai perintah pembagi
==	Sama dengan (misalnya: 12 == 10 adalah FALSE (salah) atau 12 == 12 adalah TRUE (benar))
!=	Tidak sama dengan (misalnya: 12 != 10 adalah TRUE (benar) atau 12 != 12 adalah FALSE (salah))
<	Lebih kecil dari (misalnya: 12 < 10 adalah FALSE (salah) atau 12 < 12 adalah FALSE (salah) atau 12 < 14 adalah TRUE (benar))
>	Lebih besar dari (misalnya: 12 > 10 adalah TRUE (benar) atau 12 > 12 adalah FALSE (salah) atau 12 > 14 adalah FALSE (salah))

2. Struktur Pengaturan Program

Untuk menjalankan sebuah program di butuhkan beberapa struktur dengan format seperti berikut ini:

- a. *if... else*
`if(kondisi) { }`
`else if(kondisi) { }`
`else { }`

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

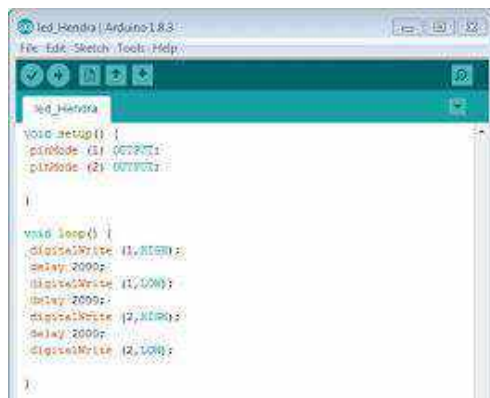
- b. *for*
`for (int i = 0; i < #pengulangan; i++) { }`

Digunakan bila anda ingin melakukan pengulangan kode di dalam kurung kurawal beberapa kali, ganti #pengulangan dengan jumlah pengulangan yang diinginkan. Melakukan penghitungan ke atas dengan i++ atau ke bawah dengan i--.

3. Digital

a. pinMode(pin, mode)

Digunakan untuk menetapkan mode dari suatu pin, pin adalah nomor pin yang akan digunakan dari 0-19 (pin analog 0-5 adalah 14-19). Mode yang bisa digunakan adalah INPUT atau OUTPUT.



Gambar 2.5. Penulisan pinMode dan penggunaan digitalWrite

Dari gambar di atas dapat dilihat pin 1 dan pin 2 digunakan sebagai output atau keluaran, Dan di Void loop nya menggunakan digitalWrite untuk mengulang-ngulang kode tersebut.

b. digitalWrite(pin, value)

Ketika sebuah pin ditetapkan sebagai OUTPUT, pin tersebut dapat dijadikan HIGH (ditarik menjadi 5 volts) atau LOW (diturunkan menjadi ground).

c. digitalRead(pin)

Ketika sebuah pin ditetapkan sebagai INPUT maka andadapat menggunakan kode ini untuk mendapatkan

nilai pin tersebut apakah HIGH (ditarik menjadi 5 volts) atau LOW (diturunkan menjadi ground).

Setelah mampu mengaplikasikan bahasa pemrograman tersebut dengan baik, kita hubungkan arduino ke dalam Project Board seperti gambar di bawah ini:



Gambar 2.6. Project Board

Dari Gambar di atas, kita dapat menghubungkan kabel dari konektor pin 13 Arduino menuju bagian titik PIN+ pada project board. Kabel GND dari Arduino dihubungkan menuju bagian titik GROUND pada project board.

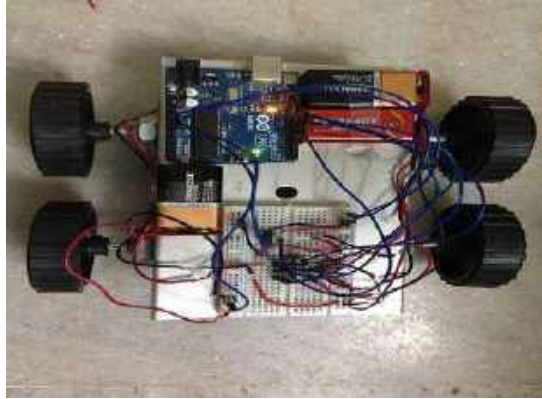
D. Pemanfaatan Arduino UNO

Pemanfaatan Arduino UNO dalam kehidupan sehari-hari beraneka ragam, antara lain:

1. SMS alarm system with Arduino

SMS alarm System menggunakan Arduino, modul GSM, sensor, resistor, dan kabel, SMS alarm system dapat digunakan untuk sistem keamanan rumah dengan melalui SMS. Proyek ini mencakup smartphone Android dan aplikasi Android dengan tombol untuk sistem.

2. Remote controlled car using Arduino and T.V. remote



Gambar 2.7. Robot Mobil Empat Roda

Proyek ini adalah robot mobil empat roda yang dikendalikan remote vi remote TV. Proyek ini meliputi Arduino UNO, dua motor DC, IR penerima, driver L293D bermotor IC, baterai, kabel, chassis sederhana, dan roda.

3. Sorting Machine

Proyek ini menunjukkan betapa sederhananya untuk membangun mesin cerdas dengan mikrokontroler Arduino. Mesin sortasi ini dirancang untuk memisahkan permen dengan berbagai warna. Mesin sortasi dapat menangani hingga 500 gram permen berwarna dengan kecepatan 80 lembar per menit.

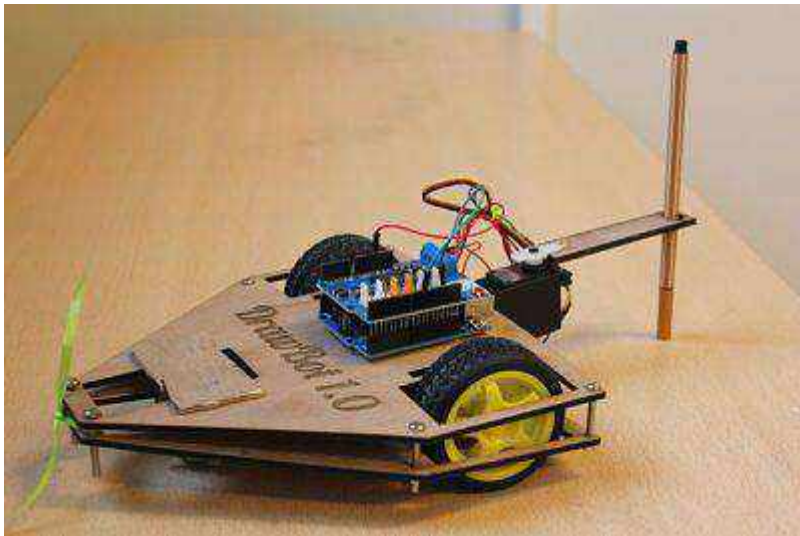
4. Robot “Cheaper”

Robot beroda murah ini direkayasa dengan sensor IR untuk mengukur jarak antara robot dan objek. Dua *servomotors* yang melekat pada roda dan mendorong robot.



Gambar 2.8. Robot dengan Sensor Jarak

5. Drawbot



Gambar 2.9. Robot Menggambar

Robot sederhana dengan pena pada lengan yang bergerak kiri dan kanan untuk menggambar bentuk.

BAB III

KEGIATAN MEMBUAT TRAFFIC LIGHT

Pada bab ini, akan di berikan beberapa proyek untuk membuat *traffic light* dengan dasar cara menyalakan satu LED.

KEGIATAN I ONE LED "ON"

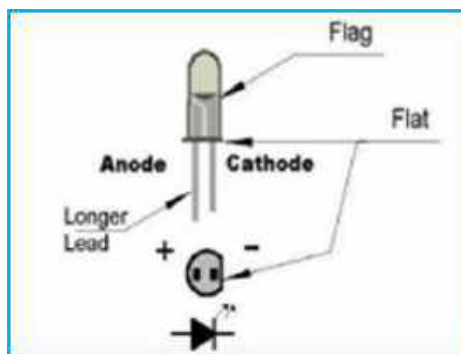
A. TUJUAN

Peserta didik dapat menganalisis penggunaan bahasa pemrograman deklarasi Output dan perintah High dalam menyalakan lampu LED (*Light Emitting Diode*)

B. DASAR TEORI LED

LED adalah salah satu jenis diode yang dapat menghasilkan cahaya. Led dapat menghasilkan cahaya berbagai macam warna tergantung dari jenisnya. Warna yang umum dihasilkan: merah, kuning, hijau, biru atau ungu, dan putih. Arus maksimum yang dapat diterima oleh Led adalah sekitar 20 mA (miliampere).

Bagian-bagian led seperti pada Gambar 3.1.



Gambar 3.1. Bagian-bagian LED

KNAP singkatan dari katoda negatif anoda positif. Prinsip dasar kerja dari led yaitu:

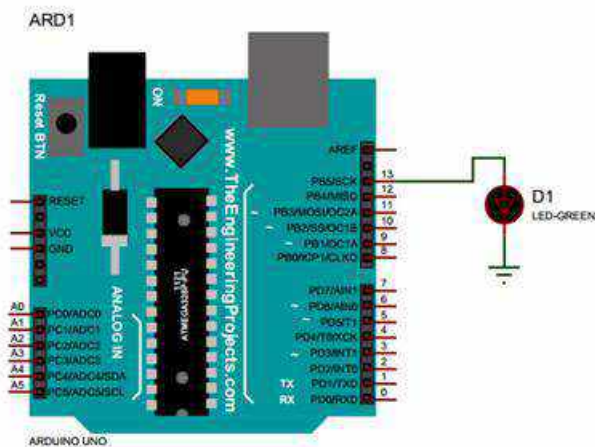
1. Lampu LED berkerja berdasarkan prinsip polarisasi. *Chip* LED mempunyai kutub positif dan negatif (p-n)
2. Hanya akan menyala bila diberikan arus maju. Ini dikarenakan LED terbuat dari bahan semikonduktor yang hanya akan mengizinkan arus listrik mengalir ke satu arah dan tidak ke arah sebaliknya.
3. Bila LED diberikan arus terbalik, hanya akan ada sedikit arus yang melewati *chip* LED. Ini menyebabkan chip LED tidak akan mengeluarkan emisi cahaya.

C. ALAT DAN BAHAN

Komputer / Laptop	1
Board Arduino Uno	1
Kabel USB A-B	1
Kabel Jumper	5
Whiteboard	1
LED	5

D. RANGKAIAN PERCOBAAN

Buatlah rangkaian seperti pada Gambar 3.2. dibawah ini.



Gambar 3.2. Rangkaian 1 LED ke Arduino

E. LANGKAH KERJA

1. Hubungkan Arduino Uno ke komputer/laptop dengan menggunakan kabel USB.
2. Hubungkan 1 lampu LED pada *white board* dengan menggunakan kabel jumper dihubungkan pada pin 13 arduino.
3. Aktifkan *software* Arduino dalam komputer / laptop.
4. Tuliskan bahasa pemrograman di bawah ini:

```
void setup() {  
  // put your setup code here, to run once:  
  pinMode(13,OUTPUT);  
}  
void loop() {  
  // put your main code here, to run repeatedly:  
  digitalWrite(13,HIGH);  
}
```

5. Klik Tools kemudian pilih Port ,lalu klik COM5 (Arduino/Genuino Uno).
6. Setelah selesai semuanya, klik Upload untuk mengunggah program data yang kita buat.

F. TUGAS

Buatlah rangkaian 4 Lampu LED yang dapat menyala secara bersamaan



KEGIATAN II

FLIP -FLOP

A. TUJUAN

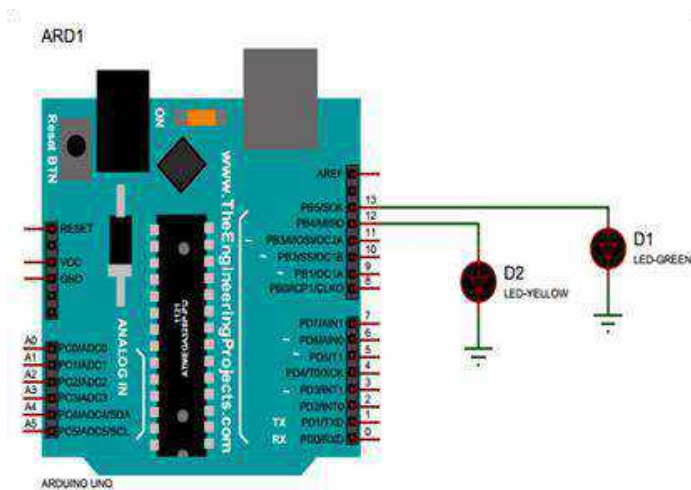
Peserta didik dapat menganalisis penggunaan perintah delay dalam menyalakan beberapa lampu LED (*Light Emiting Diode*)

B. ALAT DAN BAHAN

Komputer / Laptop	1
Board Arduino Uno	1
Kabel USB A-B	1
Kabel Jumper	5
Whiteboard Project	1
LED	5

C. RANGKAIAN PERCOBAAN

Buatlah rangkaian seperti pada Gambar 3.3. dibawah ini.



Gambar 3.3. Rangkaian 2 LED ke Arduino

D. LANGKAH KERJA

1. Hubungkan Arduino Uno ke komputer/laptop dengan menggunakan kabel USB.
2. Hubungkan 2 lampu LED yang berbeda warna pada *white board* dengan menggunakan kabel jumper dihubungkan pada pin 13 dan pin 12 arduino.
3. Aktifkan *software* Arduino dalam komputer / laptop
4. Tuliskan bahasa pemrograman di bawah ini:

```
void setup() {
    // put your setup code here, to run once:
    pinMode(13,OUTPUT);
    pinMode(12,OUTPUT);
}
void loop() {
    // put your main code here, to run repeatedly:
    digitalWrite(13,HIGH);
    delay(2000);
    digitalWrite(12, HIGH);
}
```

5. Klik Tools kemudian pilih Port ,lalu klik COM5 (Arduino/Genuino Uno).
6. Setelah selesai semuanya, klik Upload untuk mengunggah program data yang kita buat.

4. TUGAS

Buatlah rangkaian 6 lampu LED yang dapat menyala 2 lampu secara bergantian



KEGIATAN III TRAFFIC LIGHT

A. TUJUAN

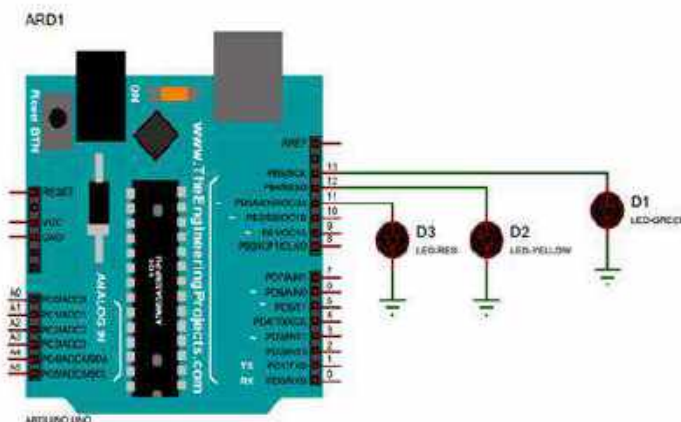
Peserta didik dapat menganalisis penggunaan perintah *delay*, *high*, *low* dalam *traffic light* satu jalur

B. ALAT DAN BAHAN

Komputer / Laptop	1
Board Arduino Uno	1
Kabel USB A-B	1
Kabel Jumper	5
Whiteboard Project	1
LED	5

C. RANGKAIAN PERCOBAAN

Buatlah rangkaian seperti pada Gambar 3.4. dibawah ini.



Gambar 3.4. Rangkaian 3 LED ke Arduino

D. PERINTAH PROGRAM

1. Hubungkan Arduino Uno ke komputer/laptop dengan menggunakan kabel USB.
2. Hubungkan 3 lampu LED yang berbeda warna pada *white board* dengan menggunakan kabel jumper dihubungkan pada pin 13, pin 12, dan pin 11 arduino.

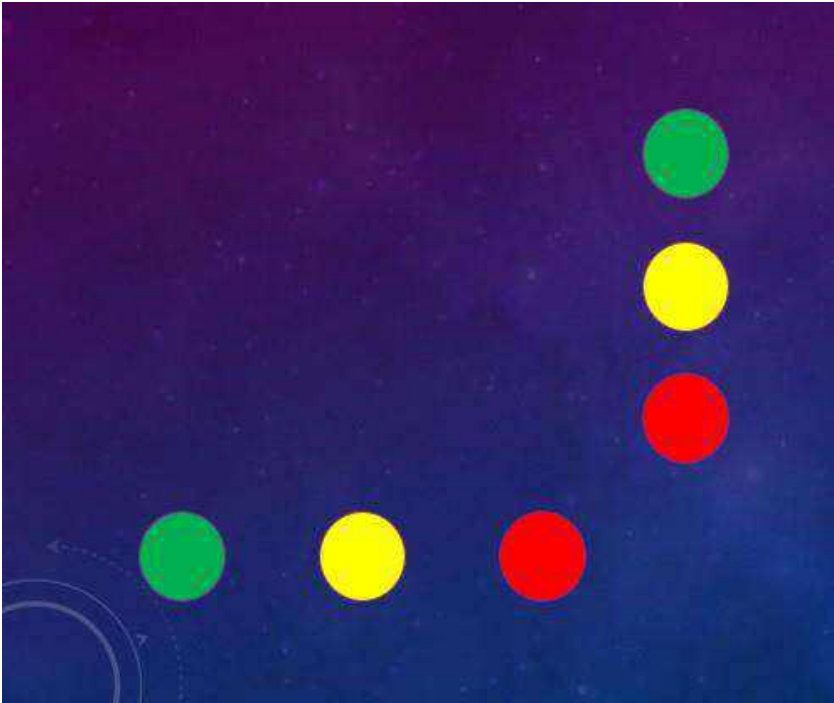
3. Aktifkan *software* Arduino dalam komputer / laptop
4. Tuliskan bahasa pemrograman di bawah ini:

```
void setup() {  
    // put your setup code here, to run once:  
    pinMode(13,OUTPUT);  
    pinMode(12,OUTPUT);  
    pinMode(11,OUTPUT);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    digitalWrite(13,HIGH);  
    digitalWrite(12, LOW);  
    digitalWrite(11, LOW);  
    delay(2000);  
  
    digitalWrite(13,LOW);  
    digitalWrite(12, HIGH);  
    digitalWrite(11, LOW);  
    delay(2000);  
  
    digitalWrite(13,LOW);  
    digitalWrite(12, LOW);  
    digitalWrite(11, HIGH);  
    delay(2000);  
}
```

5. Klik Tools kemudian pilih Port ,lalu klik COM5 (Arduino/Genuino Uno).
6. Setelah selesai semuanya, klik Upload untuk mengunggah program data yang kita buat.

E. TUGAS

Buatlah Traffic Light 2 jalur



KEGIATAN IV

TRAFFIC LIGHT WITH WARNING SYSTEM

A. TUJUAN

Peserta didik dapat menganalisis penggunaan perintah delay, high, low dan for dalam penerapan traffic light dengan sistem peringatan

B. MATERI BUZZER

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja buzzer hampir sama dengan loud speaker. Jadi buzzer juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar. Tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. Buzzer biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (alarm). Contoh buzzer dapat dilihat pada Gambar 3.5.



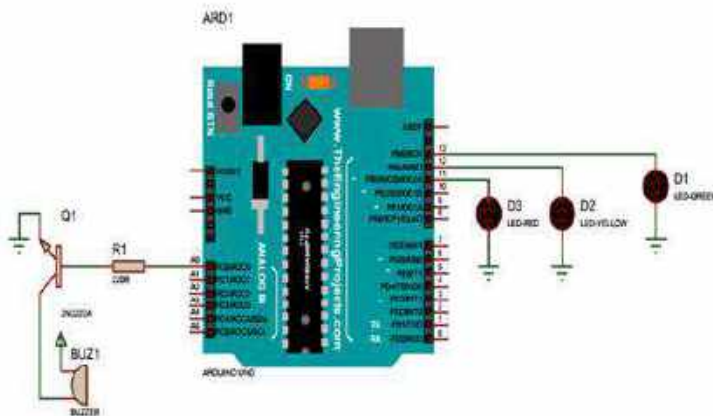
Gambar 3.5. Contoh Buzzer

C. ALAT DAN BAHAN

Komputer / Laptop	1
Board Arduino Uno	1
Kabel USB A-B	1
Kabel Jumper	5
<i>Whiteboard Project</i>	1
Buzzer	1
LED	5

D. RANGKAIAN PERCOBAAN

Buatlah rangkaian seperti pada Gambar 3.6. dibawah ini.



Gambar 3.6. Rangkaian 3 LED ke Arduino dan Buzzer

E. PERINTAH PROGRAM

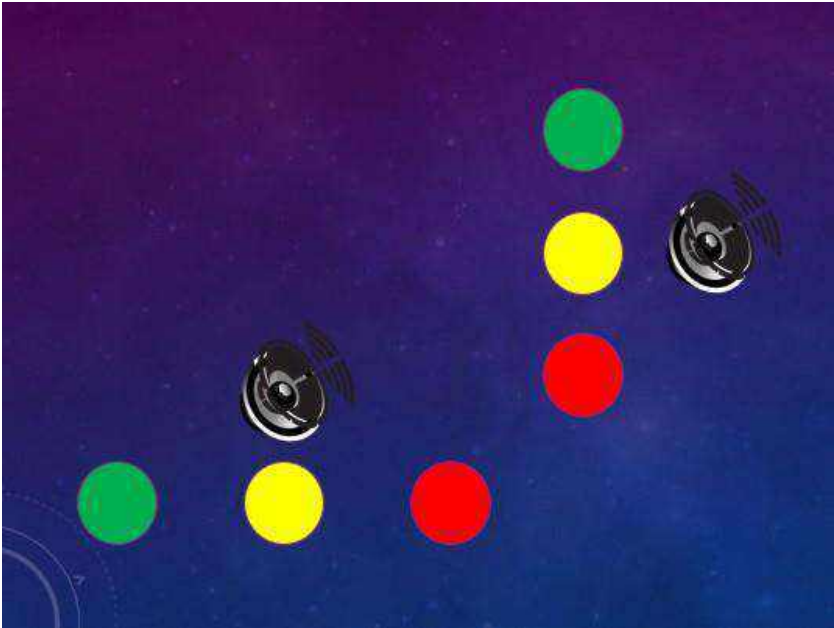
1. Hubungkan Arduino Uno ke komputer/laptop dengan menggunakan kabel USB.
2. Hubungkan 3 lampu LED yang berbeda warna pada *white board* dengan menggunakan kabel jumper dihubungkan pada pin 13, pin 12, dan pin 11 arduino. Kemudian penambahan buzzer yang dihubungkan ke pin...
3. Aktifkan *software* Arduino dalam komputer / laptop
4. Tuliskan bahasa pemrograman di bawah ini:

```
void setup() {  
    // put your setup code here, to run once:  
    pinMode(13,OUTPUT);  
    pinMode(12,OUTPUT);  
    pinMode(11,OUTPUT);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    digitalWrite(13,HIGH);  
    digitalWrite(12, LOW);  
    digitalWrite(11, LOW);  
    delay(2000);  
  
    for (int i=1; i<=3; i++)  
        digitalWrite(13,LOW);  
        digitalWrite(12, HIGH);  
        digitalWrite(11, LOW);  
        delay(2000);  
  
    digitalWrite(13,LOW);  
    digitalWrite(12, LOW);  
    digitalWrite(11, HIGH);  
    delay(2000);  
}
```

5. Klik Tools kemudian pilih Port ,lalu klik COM5 (Arduino/Genuino Uno).
6. Setelah selesai semuanya, klik Upload untuk mengunggah program data yang kita buat.

F. TUGAS

Buatlah *Traffic Light* 2 jalur dengan menggunakan peringatan bunyi buzzer



KEGIATAN V

TEXT ON LCD DISPLAY

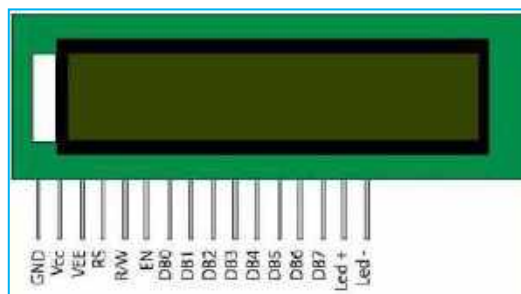
A. TUJUAN

Peserta didik dapat menganalisis penggunaan perintah `include` pada library dan Perintah `blacklight` & `print` dalam menuliskan pesan pada LCD Display

B. MATERI LCD DAN I2C

Liquid Crystal Display (LCD) merupakan tampilan visual yang menggunakan daya rendah dengan pengontrol kontras dan kecerahan, biasanya digunakan untuk jam tangan, kalkulator, multimeter, layar di hp, laptop, dan lain sebagainya. LCD juga dapat digunakan untuk menampilkan output dari mikrokontroler sehingga dapat dibaca oleh pengguna dengan mudah.

Perangkat ini memberikan tampilan angka-angka secara jelas dengan pemakaian daya yang sangat rendah. LCD dibentuk oleh suatu jenis cairan khusus yang ditempatkan di antara dua buah lempengan kaca, terdapat sebuah lempengan kaca bagian belakang, dengan sisi dalam yang ditutupi oleh elektroda transparan. Konfigurasi lcd 16x2 seperti pada Gambar 3.7.



Gambar 3.7. Lcd 16 x 2 (16 kolom, 2 baris)

Deskripsi konfigurasi pin LCD 2x16 yaitu,

1. VSS (pin 1) : ground
2. VDD (pin 2) : tegangan suplai (+5V)
3. VO (pin 3) : mengendalikan kontras dari tampilan LCD
4. RS (pin 4) : sinyal seleksi register, yaitu *high* (1) untuk input data (tulis dan baca) dan *low* (0) untuk instruksi (tulis)
5. R/W (pin 5) : sinyal seleksi instruksi, yaitu *high* (1) untuk instruksi baca dan *low* (0) untuk instruksi tulis.
6. E (pin 6) : *enable*, sinyal operasi awal yang akan mengaktifkan data tulis atau baca
7. DB0...DB7 (pin 7...pin 14) : *data bus line* (input data)
8. LEDA (pin 15) : tegangan supply LED (+5V)
9. LEDK (pin 16) : tegangan supply LED (0V).

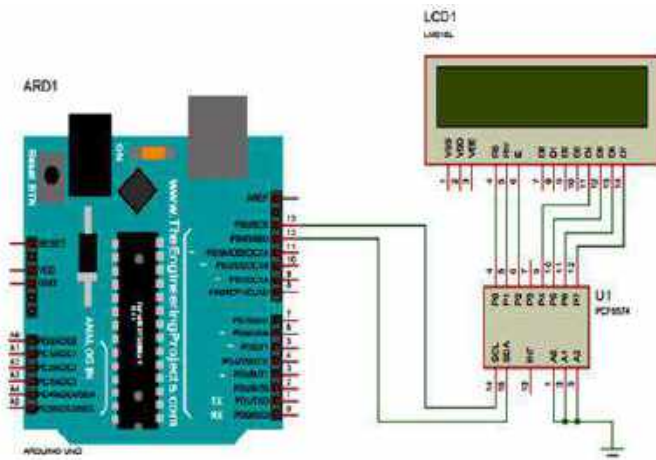
I2C adalah modul LCD yang dikendalikan secara serial sinkron dengan protokol I2C/IIC (*Inter Integrated Circuit*) atau TWI (*Two Wire Interface*). Arduino sendiri sudah mendukung protokol I2C/IIC. Di papan Arduino UNO, Port I2C terletak pada pin A4 untuk jalur SDA (Serial Data) dan pin A5 untuk SCL (Serial Clock), dan tidak lupa menghubungkan GND dan VCC. Modul I2C konverter menggunakan chip IC PCF8574 produk dari NXP sebagai kontroler. IC yang digunakan ini adalah 8 bit I/O expander untuk I2C bus yang berkerja seperti shift register. I2C LCD mempunyai 16 pin Output yang bisa dihubungkan dengan pin LCD 16x2 secara langsung (disolder permanen) dan memiliki 4 pin input (VCC, GND, SDA, SCL). Sambungan I2C ke LCD dapat dilihat pada Gambar 3.8.



Gambar 3.8. Sambungan I2C ke LCD

C. RANGKAIAN PERCOBAAN

Buatlah rangkaian seperti pada Gambar 3.9. dibawah ini.



Gambar 3.9. Rangkaian LCD 16x2 ke Arduino

D. ALAT DAN BAHAN

Komputer / Laptop	1
Board Arduino Uno	1
Kabel USB A-B	1
Kabel Jumper	5
Whiteboard Project	1
Buzzer	1
LCD	1
I2C	1
LED	5

E. PERINTAH PROGRAM

1. Hubungkan Arduino Uno ke komputer/laptop dengan menggunakan kabel USB.
2. Hubungkan lcd yang sudah dirangkai dengan i2c ke *board* arduino sesuai Gambar 3.9.
3. Aktifkan *software* Arduino dalam komputer / laptop
4. Tuliskan bahasa pemrograman di bawah ini:

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x3f, 16, 2);
void setup()
{
    lcd.begin();
    lcd.backlight();
    lcd.print("BISMILLAH");
}
void loop()
{
}
}
```

5. Klik Tools kemudian pilih Port ,lalu klik COM5 (Arduino/Genuino Uno).
6. Setelah selesai semuanya, klik Upload untuk mengunggah program data yang kita buat.

F. TUGAS

Buatlah beberapa Tulisan Pesan yang dapat berubah bergantian



DAFTAR PUSTAKA

- Andrianto, Heri dan Darmawan, Aan. 2016. *Arduino Belajar Cepat dan Pemograman*. Bandung: Informatika.
- Kadir, Abdul. 2013. *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemogramannya menggunakan Arduino*. Yogyakarta: Andi Offset.
- Wicaksono, Mochamad Fajar dan Hidayat. 2017. *Mudah Belajar Mikrokontroler Arduino Disertai 23 Proyek, Termasuk Proyek Ethernet dan Wireless Client Server*. Bandung: Informatika.
- Saftari, Firmansyah. 2015. *Proyek Robotik Keren dengan Arduino*. Jakarta: Elex Media Komputindo.
- Blum, Jeremy. 2013. *Exploring Arduino Tools and Techniques for Engineering wizardry (ebook)*. Indianapolis: John Wiley & Sons, Inc.
- Boloor, Adith Jagadish. 2015. *Arduino by Example Design and Build Fantastic Projects and Devices Using The Arduino Platform (ebook)*. Mumbai: Packt Publishing.
- Notash, Yaghoubi. 2014. *Cooperative Learning for Oral Accuracy: Investigating Inflectional Ending Pronunciation in EFL situation*. Journal of Language and Literature Education. Vol 1, no 10.pp 41-57.
- Turmuzi, Imam. 2016. *Modul Workshop ARC (Arduino Robotic Control) 2016 Implementasi Sistem Kontrol Berbasis Arduino (ebook)*. Madura: Universitas Trunojoyo Madura
- Santoso, Hari. 2015. *Panduan Praktis Arduino untuk Pemula (ebook)*. Trenggalek: ebook gratis (<http://www.elangsakti.com>).
- Skemp,R.Richard. 2001. *The Psychology Of Learning Mahematic*. London: Routledge.

- Dani, Puput. 2016. *30 Arduino Tutorials Projects Smart Arduino (ebook)*. Malang: Universitas Merdeka Malang.
- Laboratorium Mikrokontroler. *Teori Mikrokontroler Arduino Uno (ebook)*. Surabaya: Universitas Narotama Surabaya.
- Yusro, Muhammad. 2017. *Modul Teori dan Praktikum Mikrokontroler Platform Arduino (ebook)*. Jakarta: Universitas Negeri Jakarta.
- Winanda, Mukti. *Modul Praktikum EL2142 Praktikum Sistem Digital dan Mikroprosesor Beginners Guide To The Arduino (ebook)*.
- Djuandji, Feri. 2011. *Pengenalan Arduino (ebook)*. (www.tobuku.com)
- Yudhana, Anton. 2018. *Otomasi dan Instrumentasi untuk Proyek Smart Farming dan Smart Glove (ebook)*. Yogyakarta: CV Mine.
- Rinanto, Noorman dan Kautsar, Syamsiar. 2015. *Modul Ajar Praktikum Otomasi dan Robotika Kode: 609321A (ebook)*. Surabaya: Politeknik Perkapalan Negeri Surabaya.

NEW EXPERIENCE

Buku ini membahas secara lengkap cara penggunaan arduino uno dalam pembuatan suatu *prototype*.



Penerbit:
LPPM UNHAS Y Tebuireng, Jombang,
Gedung B.I.I. Jl. Irian Jaya No. 55 Tebuireng
Jombang, 61471 - Indonesia
Telp : (0322) 861719
Mail : lppm.unhasy@gmail.com / lppm@unhasy.ac.id
Site : <http://www.lppm.unhasy.ac.id>

ISBN 978-623-90696-1-2

