

LAPORAN PRAKTIKUM 1

Dosen pengampu: Erik Haritman, M.T.



Disusun oleh

Muhammad Ramdan

(NIM: 1904637)

TE02 2019

DEPARTEMEN PENDIDIKAN TEKNIK ELEKTRO

FAKULTAS PENDIDIKAN TEKNOLOGI DAN KEJURUAN

UNIVERSITAS PENDIDIKAN INDONESIA

2021

A. Tujuan Setelah Mengikuti Menyelesaikan Materi :

Setelah membaca modul diharapkan siswa dapat:

1. Peserta dapat menginstal program Arduino pada komputer/laptop masing- masing dengan benar

B. Indikator Pencapaian Kompetensi :

- a. Menenal mikrokontroler Arduino UNO
- b. Menginstal program mikrokontroller Arduino
- c. Menjalankan program mikrokontroler Arduino

C. Uraian Materi

Joobsheet ini dimaksudkan agar Anda yang masih pemula dalam dunia mikrokontroller dapat mengikuti dan mempelajari Arduino dengan mudah dan segera dapat mempraktekkannya. Oleh sebab itu, di sini akan dibahas tentang konsep elektronik, sensor, dan bahasa pemrograman secukupnya dengan harapan Anda bisa segera praktek tanpa memikirkan konsep elektronika yang relatif rumit.

Apa itu mikrokontroller?

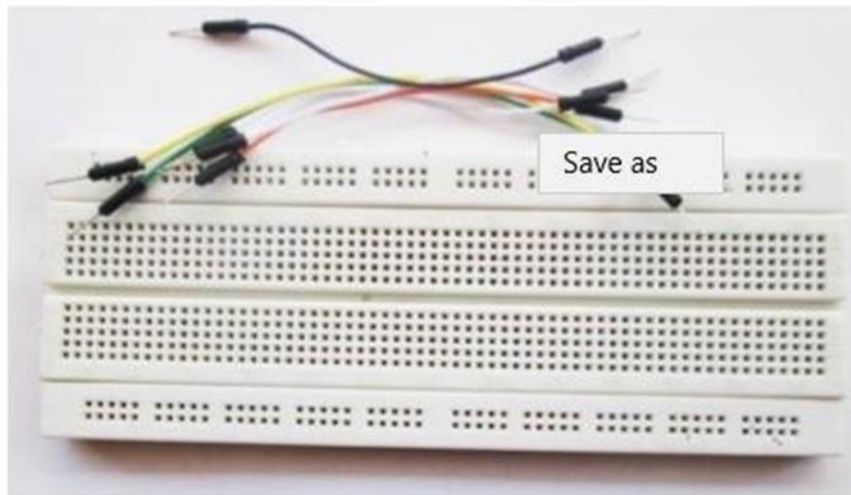
Dalam diskusi sehari-hari dan di forum internet, mikrokontroller sering dikenal dengan sebut μ C, uC, atau MCU. Terjemahan bebas dari pengertian tersebut, bisa dikatakan bahwa mikrokontroller adalah komputer yang berukuran mikro dalam satu chip IC (integrated circuit) yang terdiri dari processor, memory, dan antarmuka yang bisa diprogram. Jadi disebut komputer mikro karena dalam IC atau chip mikrokontroller terdiri dari CPU, memory, dan I/O yang bisa kita kontrol dengan memprogramnya. I/O juga sering disebut dengan GPIO (General Purpose Input Output Pins) yang berarti : pin yang bisa kita program sebagai input atau output sesuai kebutuhan.



Gambar 1 Arduino UNO

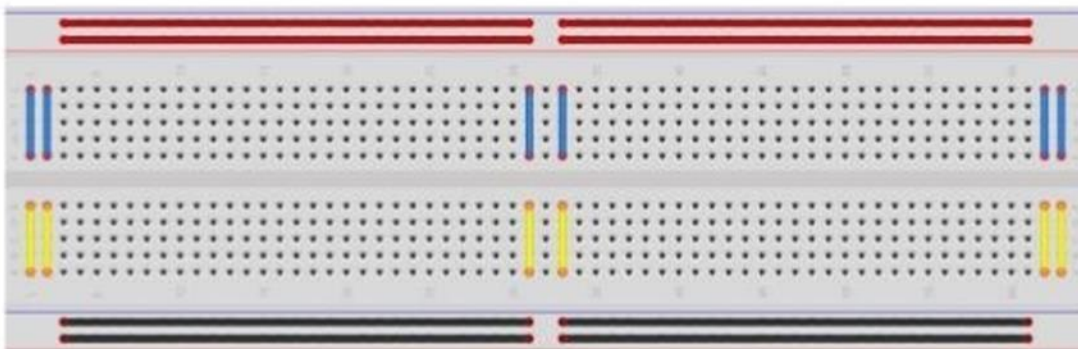
Dalam bahasan ini kita akan menggunakan board Arduino Uno (Gambar 1.1). Board Arduino terdiri dari hardware/modul mikrokontroler yang siap pakai dan software IDE yang digunakan untuk memprogram sehingga kita bisa belajar dengan mudah. Kelebihan dari Arduino yaitu kita tidak direpotkan dengan rangkaian minimum sistem dan programmer karena sudah built in dalam satu board. Oleh sebab itu kita bisa fokus ke pengembangan sistem.

Untuk praktek, kita akan menggunakan project board (ada yang menyebutnya dengan istilah bread board) dan beberapa kabel jumper untuk menghubungkan antara komponen dan Arduino (Gambar 1.2). Dengan project board kita tidak perlu menyolder rangkaian sehingga relatif mudah dan cepat dalam merangkai. Project board memungkinkan kita untuk membangun dan membongkar rangkaian dengan cepat sehingga sangat cocok untuk eksperimen. Tapi jika kita ingin membuat rangkaian yang permanen, maka kita harus menggunakan PCB.



Gambar 2 Project Board dan Kabel Jumper

Yang terpenting adalah, kita harus memahami jalur-jalur pada project board. Project board yang akan diulas di sini terdiri dari jalur vertikal dan jalur horisontal. Jalur vertikal ada di bagian tengah yang terdiri dari 2 x 64 jalur. Masing-masing jalur terdiri dari 5 titik vertikal, misal jalur 1A-1B-1C-1D-1E dan jalur 1F-1G-1H-1I-1J yang kedua tidak saling tersambung. Jalur horisontal sebanyak 8 jalur, 4 jalur ada di bagian atas dan 4 jalur lagi di bagian bawah. Jalur ini bisa digunakan untuk power supply (VCC dan GND) untuk rangkaian. Untuk lebih jelasnya, silakan perhatikan Gambar 1.3. Garis-garis yang ada menunjukkan bahwa lubang tersebut terhubung secara fisik. Ada beberapa macam model project board, ada yang besar/panjang, ada yang pendek dan ada pula yang kecil. Semua model sama dalam penggunaannya dan cara pemasangan kabel jumper, prinsipnya seperti gambar 1.3 di bawah.



Gambar 3 Peta Jalur pada Project Board

Instalasi Arduino IDE Anda bisa mendownload Arduino IDE di website Arduino, yaitu di alamat : <https://www.arduino.cc/en/Main/Software>. Pada saat tulisan ini dibuat (12/01/2017), Arduino IDE sudah versi 1.8.1. Software Arduino ada yang versi installer (hanya untuk Windows) dan versi terkompres dalam zip. Jika memilih versi tanpa install (format .zip), maka Anda hanya perlu mengekstraknya di folder mana saja dan Anda bisa langsung menjalankannya. Jika Anda pengguna Linux, maka sedikit tantangan untuk Anda karena proses instalasi tidak semudah instalasi di Windows dan Mac. Panduan untuk menginstall di Linux bisa Anda pelajari di bagian instalasi Linux.

Sedangkan untuk pengguna Windows dan Mac, Anda bisa menginstall dengan mengikuti instruksi dalam modul ini.

Instalasi di Windows

Praktik 1

1. Pasang board Arduino Anda ke port USB pada komputer atau laptop, kemudian tunggu hingga Windows mencoba untuk menginstall driver sendiri. Biasanya dia gagal menginstall driver jika belum memiliki driver tersebut. (Silakan lanjutkan ke step berikutnya)
2. Jika berhasil, berarti instalasi selesai. Tapi jika gagal, lanjutkan ke step selanjutnya.
3. Anda harus install dari device manager. Untuk masuk ke device manager, Anda bisa melakukannya dengan dua cara:

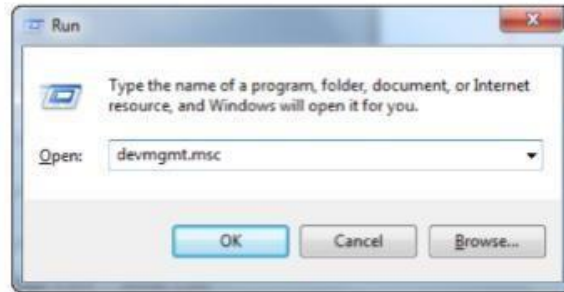


Gambar 4 Posisi tombol Windows

Tekan tombol ("Windows" + R) secara bersamaan. Tombol "Windows" adalah tombol pada keyboard dengan logo Windows (gambar logo windows, biasanya terletak di sebelah kiri atau kanan spasi, lihat Gambar 1.4). Setelah Anda menekan tombol

"Windows" + R, maka akan muncul "Run", ketikkan "devmgmt.msc" (tanpa tanda petik), kemudian tekan tombol ENTER.

Jika benar, maka akan muncul window Device Manager.



Gambar 5 Window yang muncul setelah menekan (Windows + R)

4. Jika Device Manager Anda sudah keluar, Anda bisa lanjut ke point 4, jika tidak, coba cara berikut untuk menampilkan device manager
5. Klik Start - pilih Control Panel. Di dalam Control Panel, pilih System and Security, lalu pilih System. Selanjutnya pilih Device Manager.



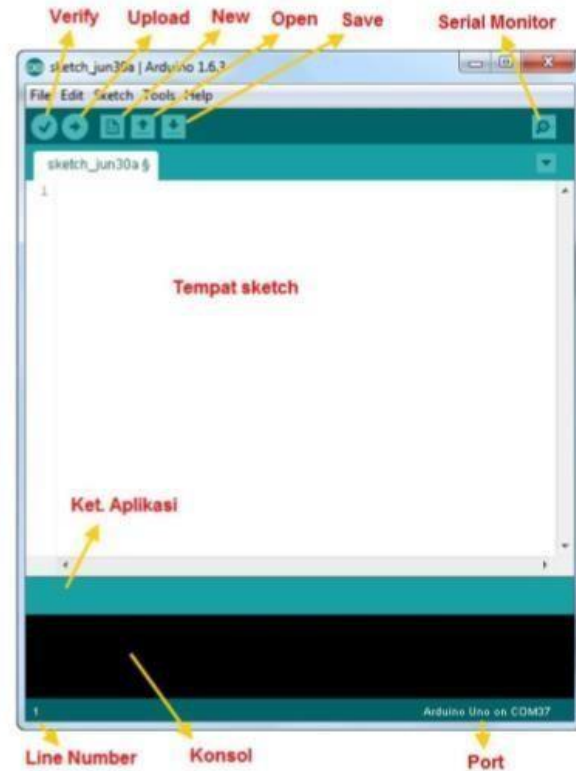
Gambar 6 Tampilan Device Manager

6. Pada Device Manager, perhatikan bagian Ports (COM & LPT), akan muncul device baru dengan nama "Arduino UNO (COMxx)"
7. Klik kanan pada "Arduino UNO (COMxx)", kemudian pilih "Update Driver Software".
8. Selanjutnya pilih "Browse my computer for Driver software".
9. Cari folder software Arduino Anda, kemudian cari file arduino.inf (khusus untuk Arduino UNO REF.3) pada folder Drivers.
10. Jika Anda menggunakan versi IDE di bawah 1.0.3, Anda bisa memilih driver dengan nama file ArduinoUNO.inf

11. Jika berhasil, berarti instalasi driver sudah selesai. Jika belum, silakan Anda mencari caranya, bisa tanya ke teman-teman ataupun mencari jawabannya di internet.
12. Selanjut mari kita coba untuk mengupload sampel code yang ada pada software Arduino
13. Jalankan Aplikasi Arduino (arduino.exe), pada pojok kanan bawah akan ada tulisan "Arduino UNO on COMxx". Berarti port yang digunakan Arduino adalah COMxx, jika tulisan tersebut tidak muncul, berarti instalasi driver belum berhasil atau board Arduino belum disambungkan ke komputer. Selanjutnya, silakan buka sampel led flipflop dengan cara Klik menu File > Examples > 1.Basic > Blink
14. Setting board Arduino dengan cara : Klik menu Tools > Board > Arduino UNO
15. Pilih port yang digunakan Arduino dengan cara mengklik menu Tools > Ports > (pilih yang ada Arduino-nya)
16. Klik tombol upload (tombol dengan panah ke kanan)
17. Setelah berhasil diupload, akan muncul tulisan "Done uploading" di bagian bawah.
18. Jika berhasil, maka LED dengan tulisan "L" pada board Arduino akan berkedip.

Arduino IDE

Untuk memprogram board Arduino, kita butuh aplikasi IDE (Integrated Development Environment) bawaan dari Arduino. Aplikasi ini berguna untuk membuat, membuka, dan mengedit source code Arduino (Sketches, para programmer menyebut source code arduino dengan istilah "sketches"). Selanjutnya, jika kita menyebut source code yang ditulis untuk Arduino, kita sebut "sketch" juga. Sketch merupakan source code yang berisi logika dan algoritma yang akan diupload ke dalam IC mikrokontroler (Arduino).



Gambar 7 Interface IDE Arduino

Interface Arduino IDE tampak seperti gambar 1.7. Dari kiri ke kanan dan atas ke bawah, bagian-bagian IDE Arduino terdiri dari:

1. Verify : pada versi sebelumnya dikenal dengan istilah Compile. Sebelum aplikasi diupload ke board Arduino, biasakan untuk memverifikasi terlebih dahulu sketch yang dibuat. Jika ada kesalahan pada sketch, nanti akan muncul error. Proses Verify/Compile mengubah sketch ke binary code untuk diupload ke mikrokontroller.
2. Upload : tombol ini berfungsi untuk mengupload sketch ke board Arduino. Walaupun kita tidak mengklik tombol verify, maka sketch akan di-compile, kemudian langsung diupload ke board. Berbeda dengan tombol verify yang hanya berfungsi untuk memverifikasi source code saja.
3. New Sketch : Membuka window dan membuat sketch baru
4. Open Sketch : Membuka sketch yang sudah pernah dibuat. Sketch yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file .ino
5. Save Sketch : menyimpan sketch, tapi tidak disertai mengcompile.

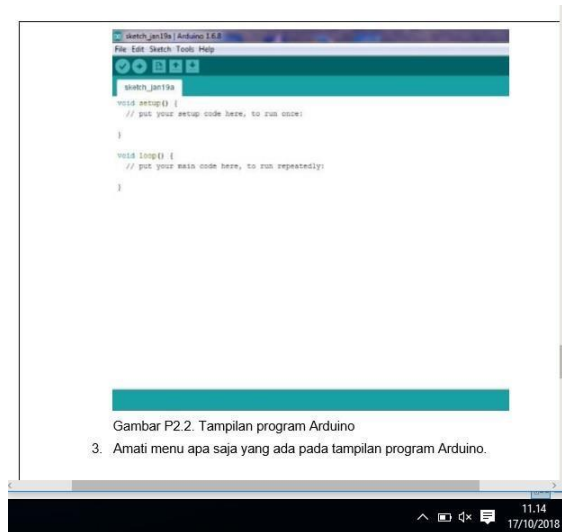
6. Serial Monitor : Membuka interface untuk komunikasi serial.
7. Keterangan Aplikasi : pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal "Compiling" dan "Done Uploading" ketika kita mengcompile dan mengupload sketch ke board Arduino
8. Konsol : Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang sketch akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada sketch yang kita buat, maka informasi error dan baris akan diinformasikan di bagian ini.
9. Baris Sketch : bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada sketch.
10. Informasi Port : bagian ini menginformasikan port yang dipakai oleh board Arduino.

Praktik 2

1. Buka program Arduino di komputer/laptop anda. Ada 2 cara untuk membukanya, dapat lewat layar desktop ataupun lewat Start > Arduino. Pertama akan muncul gambar seperti di bawah. Ini menunjukkan program sedang dijalankan. Setelah selesai akan terbuka program Arduino seperti gambar P2.2



Gambar 8 Tampilan awal



Gambar 9 Tampilan program Arduino

D. Aktivitas Pembelajaran

1. Lakukan pekerjaan install sesuai dengan langkah-langkah yang sudah tercantum di modul.
2. Setelah itu buka program dan pelajarilah aplikasi tersebut.
3. Amati menu apa saja yang ada pada tampilan program Arduino.

E. Latihan/Tugas Pertanyaan

Sebutkan tugas masing masing bagian sendiri

1. Verify : Untuk memverifikasi terlebih dahulu sketch yang dibuat. Jika ada kesalahan pada sketch, nanti akan muncul error. Proses Verify/Compile mengubah sketch ke binary code untuk diupload ke mikrokontroller.
2. Upload : tombol ini berfungsi untuk mengupload sketch ke board Arduino. Walaupun kita tidak mengklik tombol verify, maka sketch akan di-compile, kemudian langsung diupload ke board. Berbeda dengan tombol verify yang hanya berfungsi untuk memverifikasi source code saja.
3. New Sketch : Membuka window dan membuat sketch baru
4. Open Sketch : Membuka sketch yang sudah pernah dibuat. Sketch yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file .ino
5. Save Sketch : menyimpan sketch, tapi tidak disertai mengcompile.
6. Serial Monitor : Membuka interface untuk komunikasi serial.

7. Keterangan Aplikasi : pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal "Compiling" dan "Done Uploading" ketika kita mengcompile dan mengupload sketch ke board Arduino
8. Konsol : Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang sketch akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada sketch yang kita buat, maka informasi error dan baris akan diinformasikan di bagian ini.
9. Baris Sketch : bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada sketch.
10. Informasi Port : bagian ini menginformasikan port yang dipakai oleh board Arduino.

JOBSHEET 2

Program LED Berkedip

A. Tujuan Setelah Mengikuti Menyelesaikan Materi :

Setelah membaca modul diharapkan siswa dapat:

1. Peserta dapat menginstal program Arduino pada komputer/laptop masing-masing dengan benar

B. Indikator Pencapaian Kompetensi

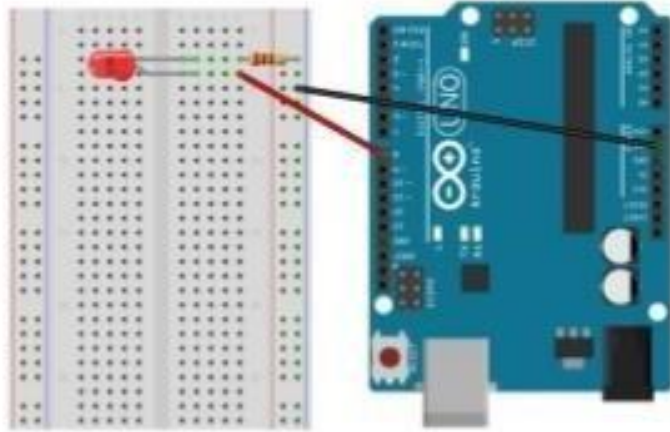
1. Membuat Sketch program lampu LED berkedip
2. Merangkai lampu LED berkedip dengan Arduino
3. Menjalankan Sketch program LED berkedip dengan Arduino

C. Uraian Materi

Rangkaian LED

Ketika belajar pemrograman, program pertama yang harus dicoba pertama kali adalah memunculkan pesan "Hello World!". Dalam belalajar mikrokontroller ternyata juga ada, yang pertama kali harus dibuat adalah membuat lampu LED berkedip, LED berkedip maksudnya adalah flip-flop. LED merupakan kependekan dari Light Emitting Diode, yaitu diode yang mampu mengubah listrik menjadi cahaya. Sebagaimana sifat diode, lampu LED memiliki kaki positif dan negatif. Sehingga pemasangannya tidak boleh terbalik, jika dipasang terbalik maka tidak akan ada arus yang mengalir dan LED pun tidak akan menyala. Arduino bekerja pada tegangan 5-12 volt dengan arus yang relatif besar yang sanggup memutuskan LED. Sehingga jika kita ingin menyambungkan LED, maka kita butuh tahanan (resistor) untuk membatasi arus yang masuk ke LED. LED memiliki tegangan kerja yang disebut dengan forward voltage (fv) yang mana tegangan ini adalah tegangan yang dibutuhkan LED untuk bisa menyala dengan baik dan aman. Ukuran resistor yang bisa dipakai adalah 100 Ω hingga 1 K Ω (Ω dibaca ohm, satuan dari resistansi/hambatan), makin besar nilai resistor maka nyala LED akan semakin redup. Pada Arduino, tegangan yang keluar dari pin-pinnya adalah 0-5 volt. Sementara catu daya untuk Arduino antara 5-12 volt. Oleh sebab itu, pemilihan resistor tergantung tegangan mana yang akan kita gunakan.

PERCOBAAN 1



Gambar 10 Rangkaian percobaan dengan menggunakan 1 LED dan 1 Resistor

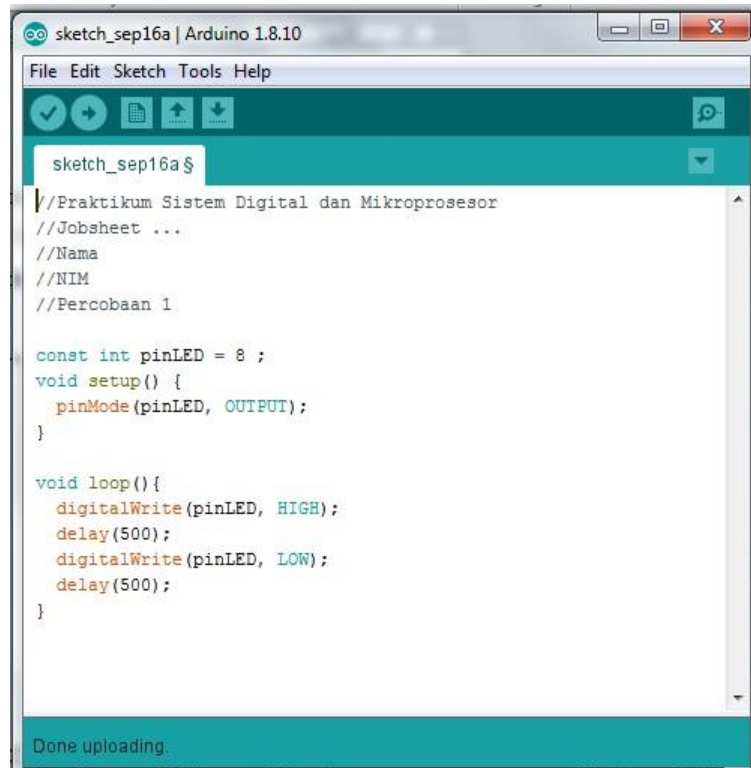
1. Hubungkan kaki anoda (+) LED ke pin 8 di board Arduino dan kaki katoda (-) LED ke resistor 220 ohm lalu ke pin GND pada board Arduino. Hubungkan board Arduino ke komputer/laptop dengan kabel USB downloader.
2. Buka program Arduino, dan ketiklah sketch program berikut!

```
*new 1 - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plug
change.log x license.dat x Genesys2007.lic x FORMAT LAPRAK PTE17.bt x
1 //Praktikum Sistem Digital dan Mikroprosesor
2 //Jobsheet ...
3 //Nama
4 //NIM
5 //Percobaan 1
6
7 const int pinLED = 8 ;
8 void setup() {
9     pinMode(pinLED, OUTPUT);
10 }
11
12 void loop() {
13     digitalWrite(pinLED, HIGH);
14     delay(500);
15     digitalWrite(pinLED, LOW);
16     delay(500);
17 }
18
```

Gambar 11 Sketch Percobaan 1

Setelah selesai membuat Sketch maka akan tampak seperti gambar di bawah. Selanjutnya tekan tombol upload untuk mengirim Sketch program ke board Arduino untuk dijalankan.

Tombol upload adalah menu panah arah ke kanan di bawahnya menu Edit



Gambar 12 Interface IDE Arduino dan Sketch Percobaan 1

Tunggu beberapa saat untuk proses mengirimkan sketch program ke board Arduino. Ditandai tulisan “Compiling sketch” pada pojok kiri bawah layar program Arduino. Setelah selesai tulisan menjadi “Done uploading”. Lihat apa yang terjadi pada rangkaian Arduino dan jelaskan apa yang Anda dapat dari pengamatan tersebut. Tuliskan hasil pengamatan anda.

PERCOBAAN 2

Memodifikasi Time Delay menggunakan command IF 1.

1. Buatlah rangkaian seperti gambar 1. Hubungkan kaki anoda (+) LED ke pin 8 di board arduino dan kaki katoda (-) LED ke resistor 220 ohm lalu ke pin GND pada board arduino. Hubungkan board Arduino ke komputer/laptop dengan kabel USB downloader.
2. Buka program Arduino, dan ketiklah sketch program berikut!

```
1 //Praktikum Sistem Digital dan Mikroprosesor
2 //Jobsheet ...
3 //Nama
4 //NIM
5 //Percobaan 2
6
7 const int pinLED = 8 ;
8
9 void setup() {
10     pinMode(pinLED, OUTPUT);
11 }
12
13 //awal time delay 1000 | 1 detik
14
15 int timeDelay=1000;
16 void loop () {
17     //setiap looping, nilai timeDelay dikurangi 100
18     timeDelay=timeDelay-100;
19     /*jika timeDelay bernilai 0 atau negatif
20     maka nilai timeDelay direset ke 1000
21     */
22     if(timeDelay <=0){
23         timeDelay=1000;
24     }
25
26     //nyalakan dan matikan LED selama timeDelay
27     digitalWrite(pinLED, HIGH);
28     delay(timeDelay);
29     digitalWrite(pinLED, LOW);
30     delay(timeDelay);
31 }
32
```

Gambar 13 Sketch Percobaan 2

Setelah selesai membuat Sketch maka selanjutnya tekan tombol upload untuk mengirim Sketch program ke board Arduino untuk dijalankan. Tombol upload adalah menu panah arah ke kanan di bawahnya menu Edit. Kalau tidak ada kesalahan pasti Sketch bisa dijalankan di Arduino. Jika ada kesalahan (error), maka carilah apa penyebabnya dan temukan pemecahannya. Tunggu beberapa saat untuk proses mengirimkan sketch program ke board Arduino. Ditandai tulisan

“Compiling sketch” pada pojok kiri bawah layar program Arduino. Setelah selesai tulisan menjadi “Done uploading”. Lihat apa yang terjadi pada rangkaian Arduino dan jelaskan apa yang Anda dapat dari pengamatan tersebut. Tuliskan hasil pengamatan anda dan bandingkan dengan hasil pengamatan percobaan 1.

D. Aktivitas Pembelajaran

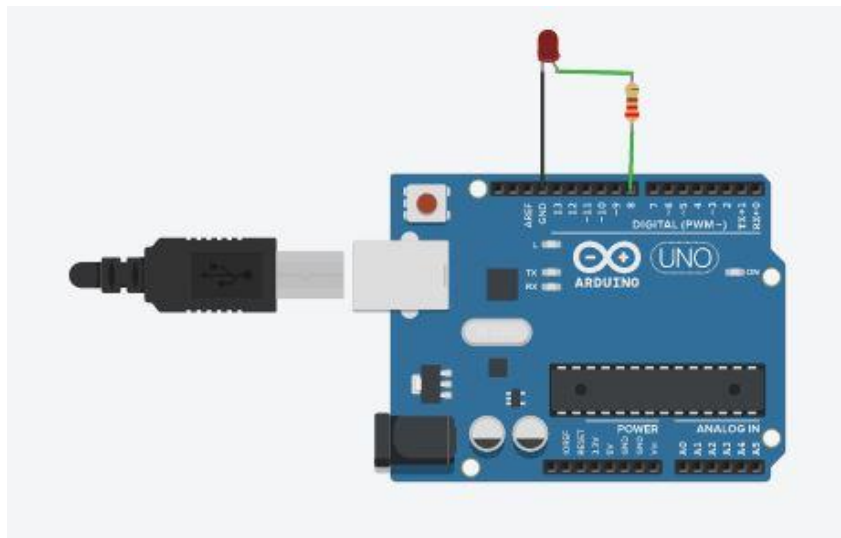
1. Selama proses pembelajaran, Peserta hendaknya mengidentifikasi dan mengamati cara menggunakan modul arduino Uno sesuai dengan manual book.

E. Tugas

1. Ubah delay pada percobaan 1 menjadi 1000 lalu amati dan analisis
2. Ubah parameter-parameter delay pada percobaan delay lalu amati dan analisis
3. Apa perbedaan dari percobaan 1 yang memasukan langsung nilai delay dengan percobaan 2 yang menggunakan command IF?

F. Jawaban

Hasil Percobaan 1



Gambar 14 Board Hasil Percobaan 1 (Delay 500 ms)

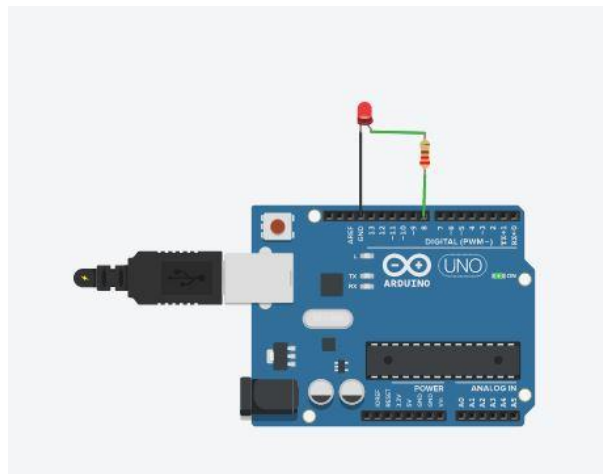

```
Text [Download] [Save] 1 (Arduino Uno R3)
1 //Praktikum Sistem Digital dan Mikroprosesor
2 //jobsheet 2
3 //Rafly Electrica Nurillah
4 //NIM 1905336
5 //Percobaan 1
6
7 const int pinLED = 8 ;
8 void setup() {
9     pinMode (8, OUTPUT);
10 }
11
12 void loop() {
13     digitalWrite (8, HIGH);
14     delay(500);
15     digitalWrite(8, LOW);
16     delay(500);
17 }
```

Gambar 15 Sketch Hasil Percobaan 1 (Delay 500 ms)

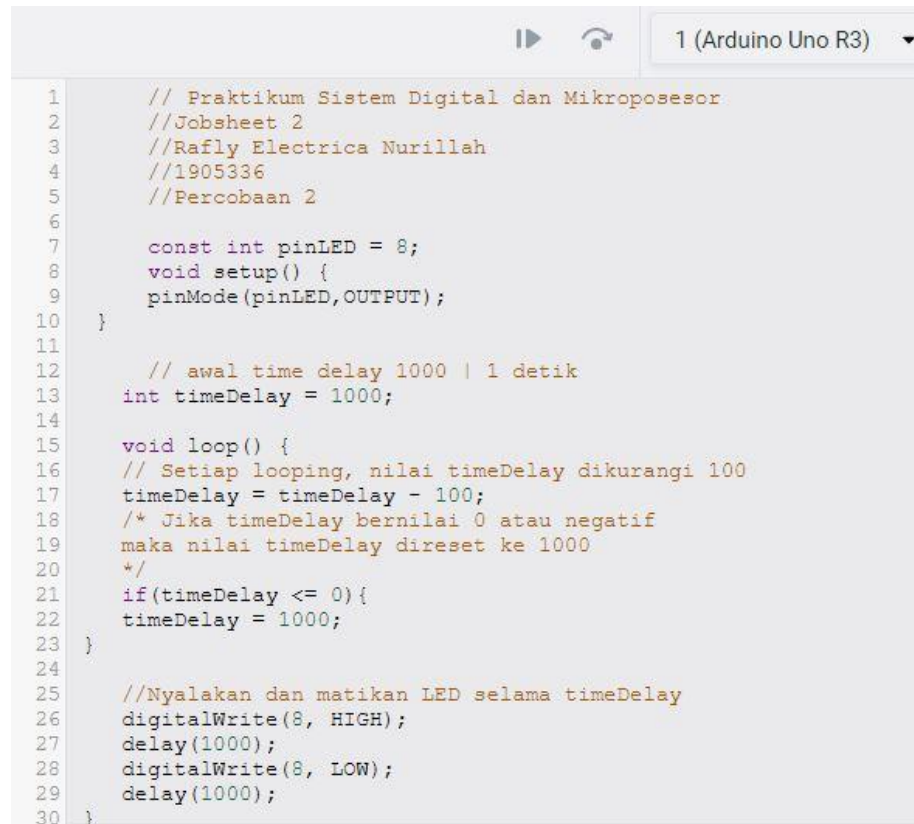
Analisis

Ketika simulasi dijalankan, lampu menyala berkedip setiap selang waktu 500 ms sesuai dengan kode yang ditulis.

HASIL PERCOBAAN 2



Gambar 16 Board Hasil Percobaan 2



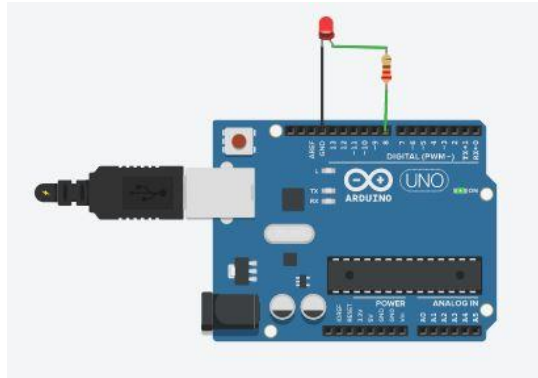
```
1 // Praktikum Sistem Digital dan Mikroposecor
2 //Jobsheet 2
3 //Rafly Electrica Nurillah
4 //1905336
5 //Percobaan 2
6
7 const int pinLED = 8;
8 void setup() {
9   pinMode(pinLED,OUTPUT);
10 }
11
12 // awal time delay 1000 | 1 detik
13 int timeDelay = 1000;
14
15 void loop() {
16   // Setiap looping, nilai timeDelay dikurangi 100
17   timeDelay = timeDelay - 100;
18   /* Jika timeDelay bernilai 0 atau negatif
19   maka nilai timeDelay direset ke 1000
20   */
21   if(timeDelay <= 0){
22     timeDelay = 1000;
23   }
24
25   //Nyalakan dan matikan LED selama timeDelay
26   digitalWrite(8, HIGH);
27   delay(1000);
28   digitalWrite(8, LOW);
29   delay(1000);
30 }
```

Gambar 17 Sketch Hasil Percobaan 2

Analisis

Ketika simulasi dijalankan, lampu menyala berkedip setiap selang waktu 1000 ms pada awalnya sesuai dengan kode yang ditulis dan timeDelaynya berkurang 1 s setiap looping. Ketika timeDelay bernilai 0 atau negatif maka nilai timeDelay akan direset kembali ke 1000 ms dan begitu seterusnya.

1. Percobaan 1 (Delay 1000)



Gambar 18 Board Hasil Percobaan 1 (Delay 1000 ms)

```

1      // Praktikum Sistem Digital dan Mikroposeor
2      //Jobsheet 2
3      //Rafly Electrica Nurillah
4      //1905336
5      //Percobaan 2
6
7      const int pinLED = 8;
8      void setup() {
9          pinMode(pinLED, OUTPUT);
10     }
11
12     void loop() {
13         digitalWrite(8, HIGH);
14         delay(1000);
15         digitalWrite(8, LOW);
16         delay(1000);
17     }

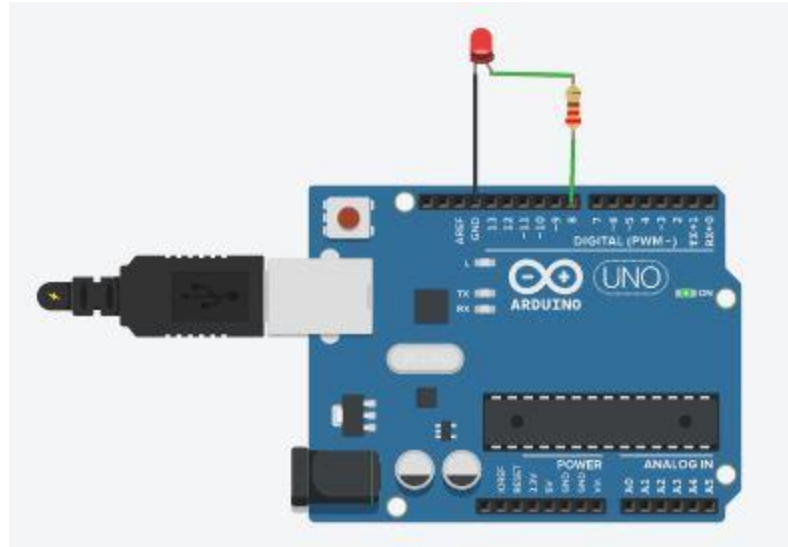
```

Gambar 19 Sketch Hasil Percobaan 1 (Delay 1000 ms)

Analisis

Ketika simulasi dijalankan, lampu menyala berkedip setiap selang waktu 1000 ms sesuai dengan kode yang ditulis.

2. Percobaan 1 (Delay 1500 ms)



Gambar 20 Board Hasil Percobaan 1 (Delay 1500 ms)

```

1      // Praktikum Sistem Digital dan Mikropotesor
2      //Jobsheet 2
3      //Rafly Electrica Nurillah
4      //1905336
5      //Percobaan 2
6
7      const int pinLED = 8;
8      void setup() {
9          pinMode(pinLED,OUTPUT);
10     }
11
12     void loop() {
13         digitalWrite(8, HIGH);
14         delay(1500);
15         digitalWrite(8, LOW);
16         delay(1500);
17     }

```

Gambar 21 Sketch Hasil Percobaan 1 (Delay 1500 ms)

Analisis

Ketika simulasi dijalankan, lampu menyala berkedip setiap selang waktu 1500 ms sesuai dengan kode yang ditulis.

3. Perbedaan Percobaan 1 dan 2

Perbedaannya ada pada time delay, pada percobaan 1 dengan memasukan langsung nilai delay maka hasil yang didapatkan berupa interval yang konsta. Sementara pada percobaan 2 yang memiliki command IF maka nilai delay akan selalu berkurang pada tiap interval dan akan kembali ke semula ketika nilai tersebut mencapai kurang dari atau sama dengan 0.

JOBSHEET 3

Program LED Berkedip

A. Tujuan Setelah mengikuti menyelesaikan materi

Setelah membaca modul diharapkan siswa dapat :

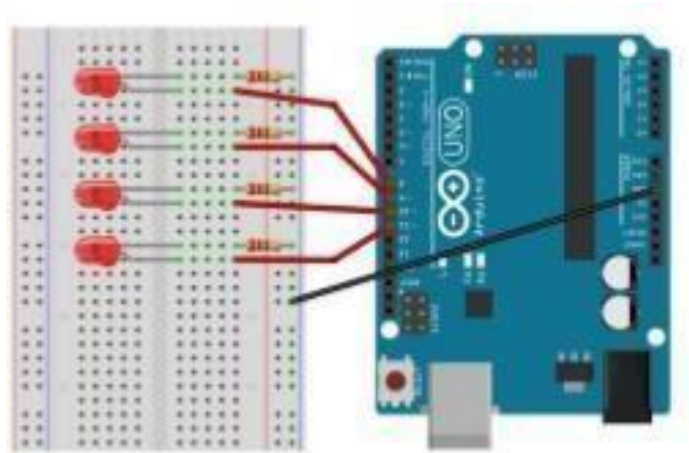
1. Menguji program Arduino untuk menyalakan lampu LED berderet dengan benar.

B. Indikator Pencapaian Kompetensi

1. Membuat Sketch program lampu LED berderet
2. Merangkai lampu LED berderet dengan Arduino
3. Menjalankan Sketch program LED berderet dengan Arduino

C. Uraian Materi

Jobsheet ini akan mempraktikkan pemrograman LED berderet sebagai pengembangan dari jobsheet sebelumnya.



Gambar 22 Percobaan 1

PERCOBAAN 1

1. Siapkan 4 buah resistor dan 4 buah LED. Siapkan kabel jumper untuk menyuplai GND pada project board.
2. Masing-masing kaki negatif LED dihubungkan ke GND dengan resistor. Sedangkan keempat LED tersebut dihubungkan berturut-turut dengan pin 8, 9, 10, dan 11 pada board Arduino.
3. Buka program Arduino, dan ketiklah sketch program berikut!

```

1 //Praktikum Sistem Digital dan Mikroprosesor
2 //Jobsheet 3
3 //Nama
4 //NIM
5 //Percobaan 1
6
7 //inisialisasi pinLED
8 const int pinLED1=8;
9 const int pinLED2=9;
10 const int pinLED3=10;
11 const int pinLED4=11;
12
13 void setup() {
14 //pinLED sebagai output
15 pinMode(pinLED1, OUTPUT);
16 pinMode(pinLED2, OUTPUT);
17 pinMode(pinLED3, OUTPUT);
18 pinMode(pinLED4, OUTPUT);
19 }
20
21 void loop () {
22 //perulangan sebanyak 5 kali
23 //dari i=0 hingga i=4 atau (i<5)
24 for(int i=0;i<5;i++){
25 if(i==1){
26 //jika i=1, hidupkan led 1, led yang lain matik
27 digitalWrite(pinLED1, HIGH);
28 digitalWrite(pinLED2, LOW);
29 digitalWrite(pinLED3, LOW);
30 digitalWrite(pinLED4, LOW);
31 }else if(i==3){
32 //jika i=3, hidupkan led 1, 2 dan 3, led 4 mati
33 digitalWrite(pinLED1, HIGH);
34 digitalWrite(pinLED2, HIGH);
35 digitalWrite(pinLED3, HIGH);
36 digitalWrite(pinLED4, LOW);
37 }else if (i==4){
38 //jika i=4, hidupkan semua LED
39 digitalWrite(pinLED1, HIGH);
40 digitalWrite(pinLED2, HIGH);
41 digitalWrite(pinLED3, HIGH);
42 digitalWrite(pinLED4, HIGH);
43 }else {
44 //jika tidak ada, matikan semua led
45 digitalWrite(pinLED1, LOW);
46 digitalWrite(pinLED2, LOW);
47 digitalWrite(pinLED3, LOW);
48 digitalWrite(pinLED4, LOW);
49 }
50 //delay selama 5 detik
51 delay(1000);
52 }
53

```

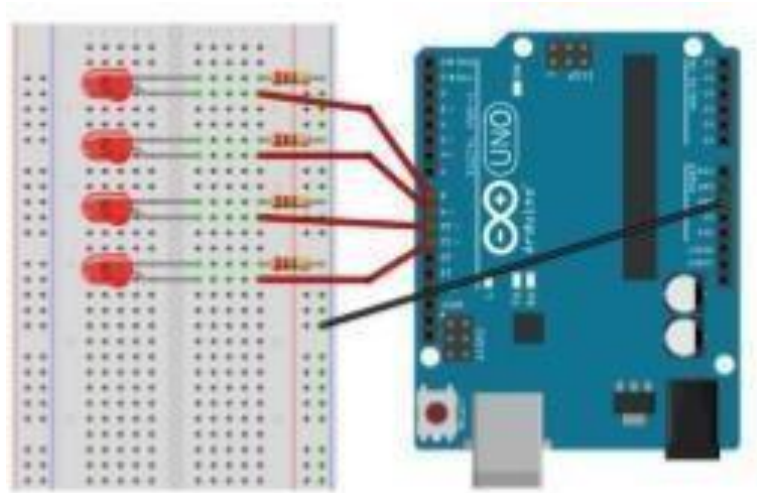
Gambar 23 Sketch percobaan 1 jobsheet 3

Setelah selesai membuat Sketch, lanjutnya tekan tombol upload untuk mengirim Sketch program ke board Arduino untuk dijalankan. Tombol upload adalah menu panah arah ke kanan di bawahnya menu Edit.

4. Tunggu beberapa saat untuk proses mengirimkan sketch program ke board Arduino. Ditandai tulisan “Compiling sketch” pada pojok kiri bawah layar program Arduino. Setelah selesai tulisan menjadi “Done uploading”.
5. Lihat apa yang terjadi pada rangkaian Arduino dan jelaskan apa yang Anda dapat dari pengamatan tersebut. Tuliskan hasil pengamatan anda.

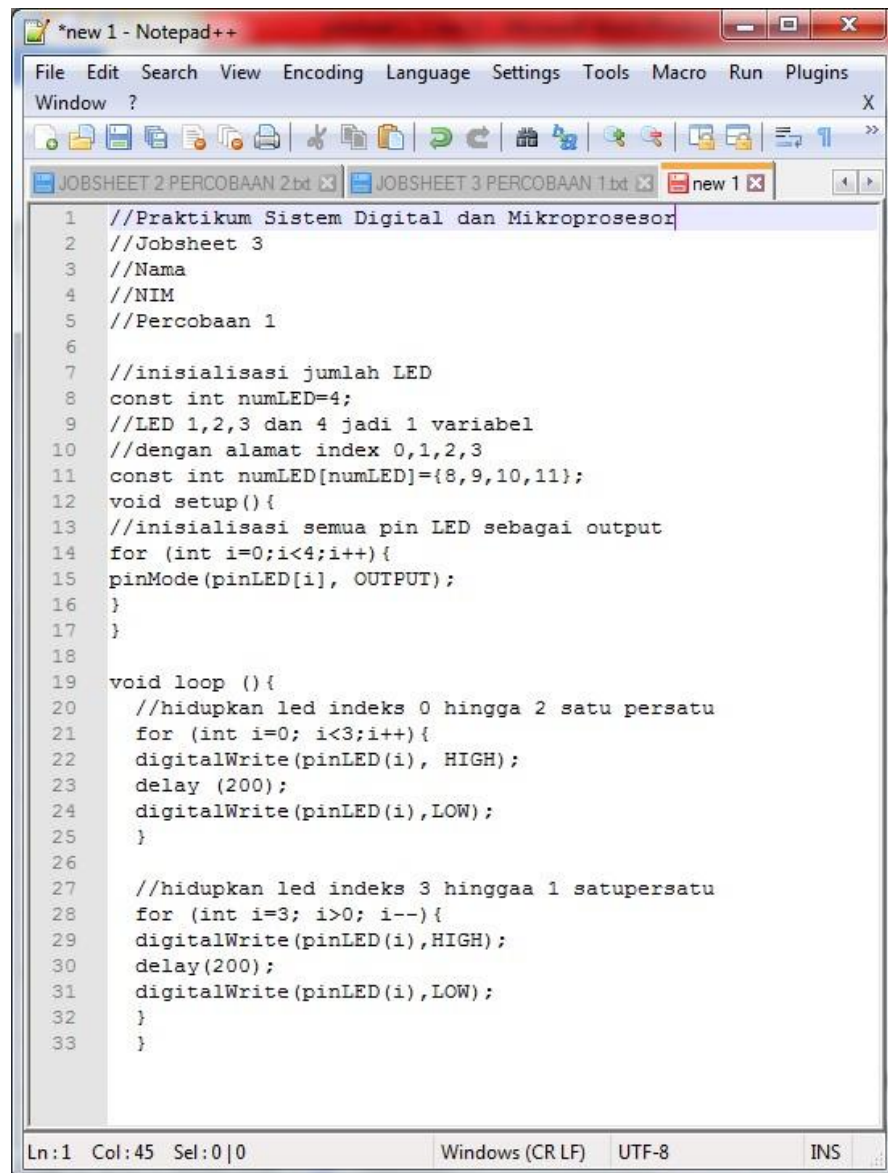
PERCOBAAN 2

Memodifikasi Time Delay menggunakan IF 1. Buatlah rangkaian seperti gambar 3.



Gambar 24 Percobaan 2

1. Siapkan 4 buah resistor dan 4 buah LED. Siapkan kabel jumper untuk menyuplai GND pada project board.
2. Masing-masing kaki negatif LED dihubungkan ke GND dengan resistor. Sedangkan keempat LED tersebut dihubungkan berturut-turut dengan pin 8, 9, 10, dan 11 pada board Arduino.
3. Buka program Arduino, dan ketiklah sketch program berikut!



```
1 //Praktikum Sistem Digital dan Mikroprosesor
2 //Jobsheet 3
3 //Nama
4 //NIM
5 //Percobaan 1
6
7 //inisialisasi jumlah LED
8 const int numLED=4;
9 //LED 1,2,3 dan 4 jadi 1 variabel
10 //dengan alamat index 0,1,2,3
11 const int pinLED[numLED]={8,9,10,11};
12 void setup(){
13 //inisialisasi semua pin LED sebagai output
14 for (int i=0;i<4;i++){
15   pinMode(pinLED[i], OUTPUT);
16 }
17 }
18
19 void loop (){
20   //hidupkan led indeks 0 hingga 2 satu persatu
21   for (int i=0; i<3;i++){
22     digitalWrite(pinLED(i), HIGH);
23     delay (200);
24     digitalWrite(pinLED(i),LOW);
25   }
26
27   //hidupkan led indeks 3 hinggaa 1 satupersatu
28   for (int i=3; i>0; i--){
29     digitalWrite(pinLED(i),HIGH);
30     delay(200);
31     digitalWrite(pinLED(i),LOW);
32   }
33 }
```

Gambar 25 Sketch percobaan 2

Setelah selesai membuat Sketch maka selanjutnya tekan tombol upload untuk mengirim Sketch program ke board Arduino untuk dijalankan. Tombol upload adalah menu panah arah ke kanan di bawahnya menu Edit. Kalau tidak ada kesalahan pasti Sketch bisa dijalankan di Arduino. Jika ada kesalahan (error), maka carilah apa penyebabnya dan temukan pemecahannya.

4. Tunggu beberapa saat untuk proses mengirimkan sketch program ke board Arduino. Ditandai tulisan “Compiling sketch” pada pojok kiri bawah layar program Arduino. Setelah selesai tulisan menjadi “Done uploading”.

5. Lihat apa yang terjadi pada rangkaian Arduino dan jelaskan apa yang Anda dapat dari pengamatan tersebut. Tuliskan hasil pengamatan anda kemudian bandingkan dengan hasil pengamatan percobaan 1.

D. Tugas

1. Analisis perbedaan dari percobaan 1 dan 2?
2. Buatlah program yang lain dengan jumlah LED lebih banyak yaitu 6 dengan kondisi nyala sebagai berikut:
 - a. i=1, i=2, i=3, i=4, i=5 dan i=6
 - b. nyala berganti-gantian satu persatu

E. Jawaban

1. Pada kedua rangkaian terdapat perbedaan rumus yang membuat perbedaan yang terdapat pada rangkaian yang disebabkan oleh perbedaan fungsi yang digunakan.

2.

```
Text [dropdown] [download icon] [upload icon] 1 (Arduino Uno R3) [dropdown]

1 //Rafly Electrica Nurillah
2 //1905336
3 //TE 02
4
5 //Praktikum Sistem Digital dan Mikroprosesor
6 //Jobsheet 3
7
8 const int pinLED1=8;
9 const int pinLED2=9;
10 const int pinLED3=10;
11 const int pinLED4=11;
12 const int pinLED5=12;
13 const int pinLED6=13;
14
15 void setup(){
16     pinMode (pinLED1, OUTPUT);
17     pinMode (pinLED2, OUTPUT);
18     pinMode (pinLED3, OUTPUT);
19     pinMode (pinLED4, OUTPUT);
20     pinMode (pinLED5, OUTPUT);
21     pinMode (pinLED6, OUTPUT);
22
23 }
24
25 void loop(){
26     for(int i=0;i=7;i++){
27
28         if(i==1){
29             digitalWrite (pinLED1, HIGH);
30             digitalWrite (pinLED2, LOW);
```

```

31     digitalWrite (pinLED3, LOW);
32     digitalWrite (pinLED4, LOW);
33     digitalWrite (pinLED5, LOW);
34     digitalWrite (pinLED6, LOW);
35 }
36 else if(i==2){
37     digitalWrite (pinLED1, LOW);
38     digitalWrite (pinLED2, HIGH);
39     digitalWrite (pinLED3, LOW);
40     digitalWrite (pinLED4, LOW);
41     digitalWrite (pinLED5, LOW);
42     digitalWrite (pinLED6, LOW);
43 }
44 else if(i==3){
45     digitalWrite (pinLED1, LOW);
46     digitalWrite (pinLED2, LOW);
47     digitalWrite (pinLED3, HIGH);
48     digitalWrite (pinLED4, LOW);
49     digitalWrite (pinLED5, LOW);
50     digitalWrite (pinLED6, LOW);
51 }
52 else if(i==4){
53     digitalWrite (pinLED1, LOW);
54     digitalWrite (pinLED2, LOW);
55     digitalWrite (pinLED3, LOW);
56     digitalWrite (pinLED4, HIGH);
57     digitalWrite (pinLED5, LOW);
58     digitalWrite (pinLED6, LOW);
59 }
60 else if(i==5){

```

```

61     digitalWrite (pinLED1, LOW);
62     digitalWrite (pinLED2, LOW);
63     digitalWrite (pinLED3, LOW);
64     digitalWrite (pinLED4, LOW);
65     digitalWrite (pinLED5, HIGH);
66     digitalWrite (pinLED6, LOW);
67 }
68 else if(i==6){
69     digitalWrite (pinLED1, LOW);
70     digitalWrite (pinLED2, LOW);
71     digitalWrite (pinLED3, LOW);
72     digitalWrite (pinLED4, LOW);
73     digitalWrite (pinLED5, LOW);
74     digitalWrite (pinLED6, HIGH);
75 }
76 else{
77     digitalWrite (pinLED1, LOW);
78     digitalWrite (pinLED2, LOW);
79     digitalWrite (pinLED3, LOW);
80     digitalWrite (pinLED4, LOW);
81     digitalWrite (pinLED5, LOW);
82     digitalWrite (pinLED6, LOW);
83 }
84     delay(1000);
85 }
86 }
87 }

```