



Arduino Workshop 1

Zuid-Limburg

Leren door doen

Workshop 1

Workshop 1 concentreert op kennismaking en eenvoudige programmeer voorbeelden en oefeningen

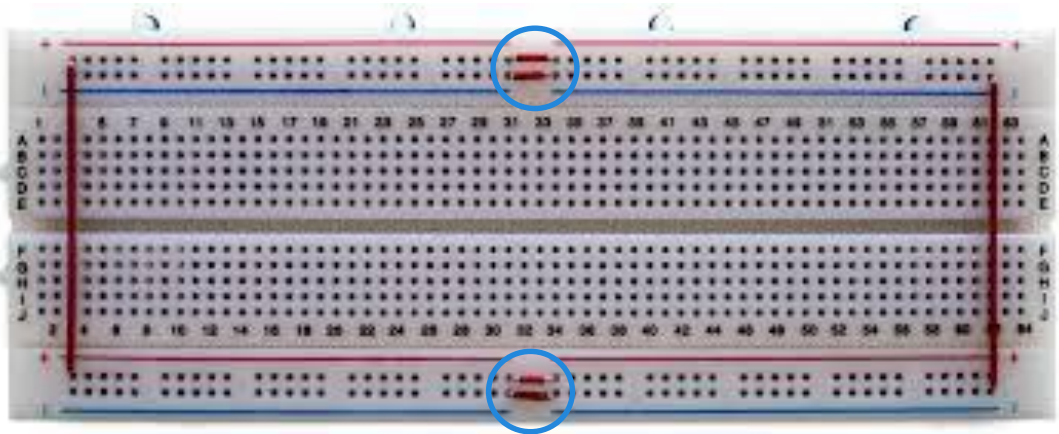
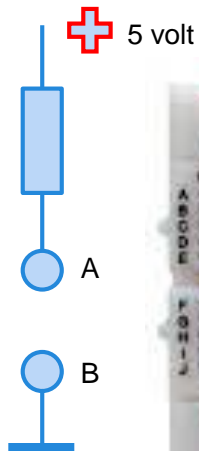
- Kennismaken met de taal
- Oefeningen met LED, drukknop, toon opwekking en LCD aansturing

Leren door doen

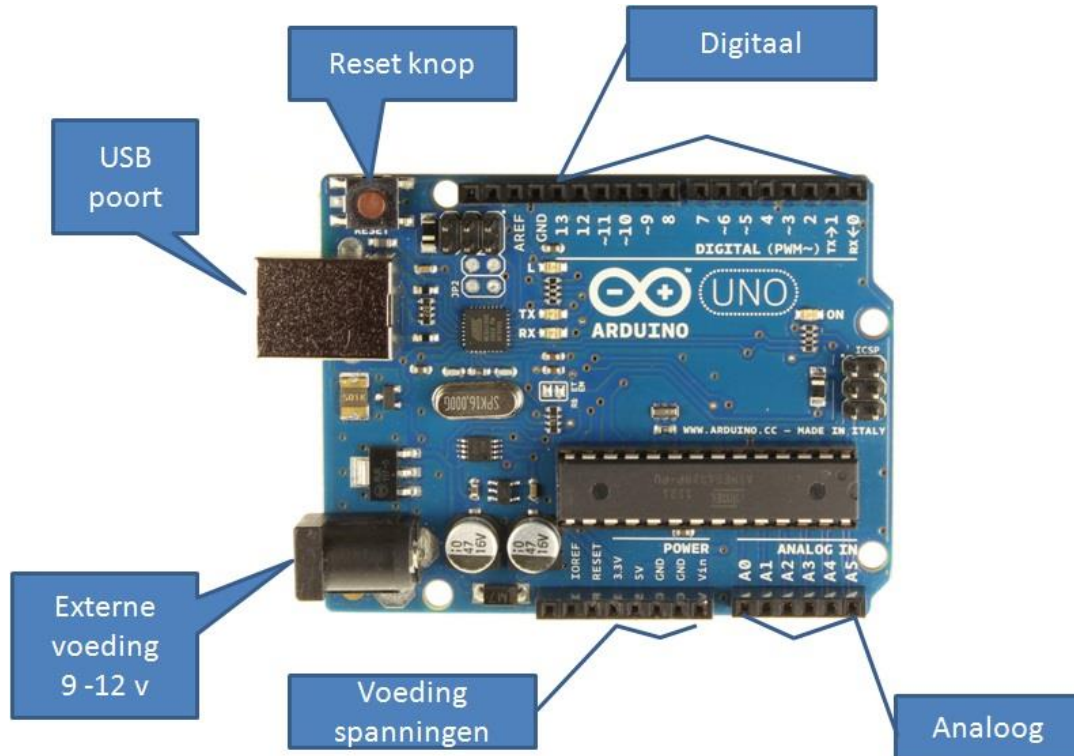
- Werk individueel of in een groepje
- Thuis oefenen
- Informatie uitwisselen

Geen informatica cursus

Pull-up & Breadboard



Arduino: outline



Bestand Bewerken Schets Hulpmiddelen Help



sketch_mar01c

```
void setup() {  
  // put your setup code here, to run once:  
  
}  
  
void loop() {  
  // put your main code here, to run repeatedly:  
  
}
```

De programmeertaal

void setup() {.....}

De code binnen de {.....} wordt alleen aan het begin uitgevoerd. Dit is wanneer de Arduino gereset wordt.

void loop() {.....}

De code binnen de {.....} wordt als een oneindige loop uitgevoerd.

pinMode(pin, mode);

Met pinMode definieer je een bepaalde pin als in of uitgang. Mode kan INPUT of OUTPUT zijn. Bijvoorbeeld: pinMode(13, OUTPUT);

digitalWrite(pin, output);

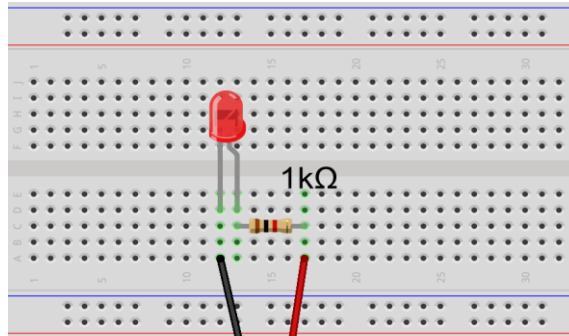
Hiermee bepaal je de digitale waarde van een bepaalde pin. Output kan HIGH of LOW zijn. Bijvoorbeeld: digitalWrite(13, HIGH);

delay(time);

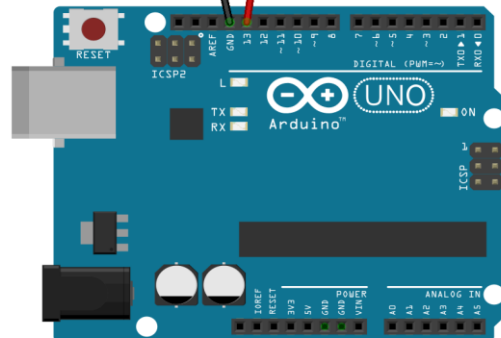
Hiermee laat je het programma pauzeren, de tijd wordt aangegeven in milliseconden. Bijvoorbeeld: delay(1000);

// **commentaar.** Wat je achter '/' op de regel schrijft wordt niet door het programma gebruikt

De knipperende LED



Weerstand 470 ohm





De setup

```
void setup()  
{  
    pinMode(13, OUTPUT);  
}
```


Omzetten naar 'C'

Begin:

doe de led aan

wacht 1 seconde 

doe de led uit

wacht 1 seconde

ga naar Begin

```
void loop()
```

```
{
```

```
  digitalWrite(13, HIGH);
```

```
  delay(1000); // wachten
```

```
  digitalWrite(13, LOW);
```

```
  delay(1000); // wachten
```

```
}
```



sketch_feb19a_knipperled_eerste_oefening

```
void setup()
{
  pinMode(13, OUTPUT); // pin 13 wordt een digitale uitgang
}
void loop()
{
  digitalWrite(13, HIGH); // pin 13 wordt hoog; de led gaat branden
  delay(1000); // wacht 1000 milliseconden
  digitalWrite(13, LOW); // pin 13 wordt laag; de led gaat uit
  delay(1000); // wacht 1000 milliseconden
}
```

Integer variabele

```
int x = 13; // declareer de variabele 'x' als integer
```

Aan de integer 'x' wordt de waarde 13 toegekend.

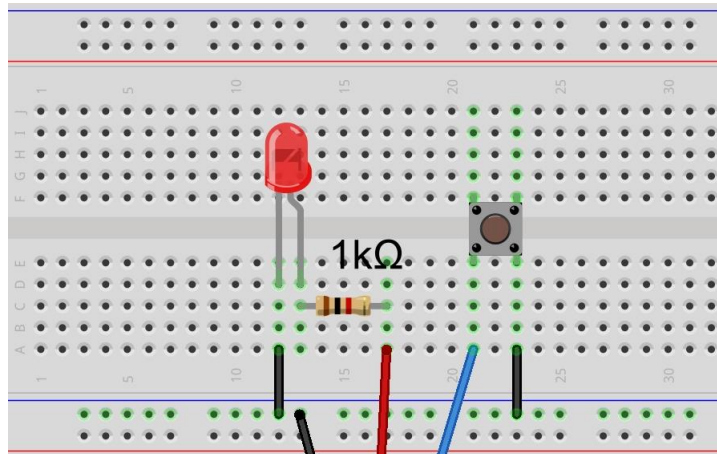
Integer variabelen zijn bedoeld om getallen te bewaren zonder een decimale punt, met een waarde tussen -32768 tot +32767 (16 bit)



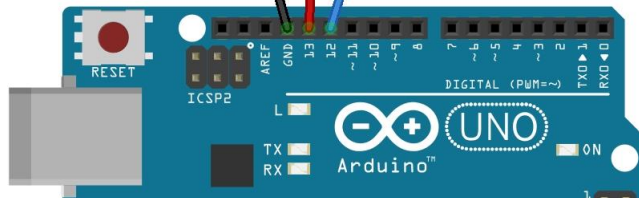
sketch_feb19b_kniperled_oefening_2

```
// LED knipperprogramma
int ledpin = 13; // de integer variabele ledpin wordt 13
void setup()
{
  pinMode(ledpin, OUTPUT); // de pin ledpin wordt een digitale uitgang
}
void loop()
{
  digitalWrite(ledpin, HIGH); // de pin ledpin wordt hoog; de led gaat branden
  delay(1000); // wacht 1 seconde
  digitalWrite(ledpin, LOW); // de pin ledpin wordt laag; de led gaat uit
  delay(500); // wacht 0,5 seconde
}
```

Oefening 2: de drukknop



Weerstand 470 ohm





Enkele nieuwe statements

digitalWrite(inputpin, HIGH)

Er wordt een pull-up weerstand geactiveerd waardoor deze pin standaard HIGH wordt, tenzij er een verbinding met de ground wordt gemaakt.

digitalRead(pin)

Leest de digitale waarde van een digitale INPUT pin. (Geeft een boolean terug, HIGH, LOW of TRUE, FALSE of 1, 0)

if (condition) {.....} else {.....}

Voert de code binnen de eerste haakjes uit wanneer wat tussen de ronde haakjes staat waar (TRUE, HIGH of 1) is, anders wordt de code tussen de haakjes na else uitgevoerd.

== Betekent 'is gelijk aan'! Niet verwarren met de enkele '=', want die kent een waarde toe aan een variabele

Het 'if' statement

```
if(condition) {.....} else {.....}
```

Voert de code binnen de eerste haakjes uit wanneer wat tussen de ronde haakjes staat waar (TRUE, HIGH of 1) is, anders wordt de code tussen de haakjes na else uitgevoerd.

```
if (digitalRead(dkpin) == LOW)           // controleer of drukknop is ingedrukt
```

```
{  
  digitalWrite(ledpin, HIGH);           // zet LED aan  
}
```

```
else
```

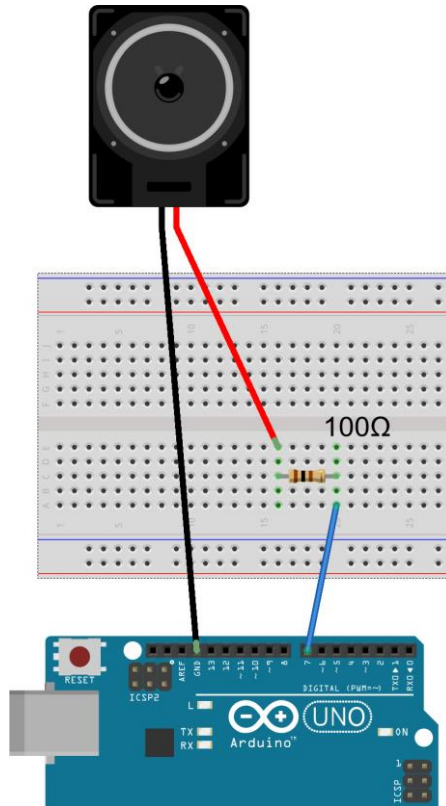
```
{  
  digitalWrite(ledpin, LOW);           // zet LED uit  
}
```



sketch_feb19c_drukknop_oefening

```
// Drukknop oefening
// benoem de pin nummers:
int drpin = 12; // pin nummer voor drukknop
int ledpin = 13; // pin nummer voor LED
void setup()
{
  pinMode(drpin, INPUT); // initialiseer de drukknop pin als input
  pinMode(ledpin, OUTPUT); // initialiseer de LED pin als output
  digitalWrite(drpin, HIGH); // maak drpin hoog
}
void loop()
{
  if (digitalRead(drpin) == LOW) // controleer of drukknop is ingedrukt
  {
    digitalWrite(ledpin, HIGH); // zet LED aan
  }
  else
  {
    digitalWrite(ledpin, LOW); // zet LED uit
  }
}
```


Oefening 3: Toon generatie



Weerstand 470 ohm



Toon generatie statements

tone(pin, frequentie);

Er wordt een toon signal (50% duty cycle) op uitgang 'pin' gegenereerd. Bijvoorbeeld `tone(7,800);`

noTone(pin);

Stop de toon generatie op de uitgang 'pin'.
Bijvoorbeeld: `noTone(7);`



```
sketch_feb19d_toon_generatie_oefening_1
```

```
// Defenprogramma toon generatie
void setup()
{
  pinMode(7, OUTPUT); // initialiseer de LS pin als output
}
void loop()
{
  tone(7,800); // toon van 800 Hz op pin 7
  delay(500); //laat toon 500 ms aan
  noTone(7); //schakel toon uit
  delay(1000);
}
```

De subroutine

```
void dot()
{
  tone(7,800);           // toon aan
  delay(t);             // toon t ms (dot) aan
  noTone(7);           // schakel toon uit
  delay(t);            // symbool spatie
}

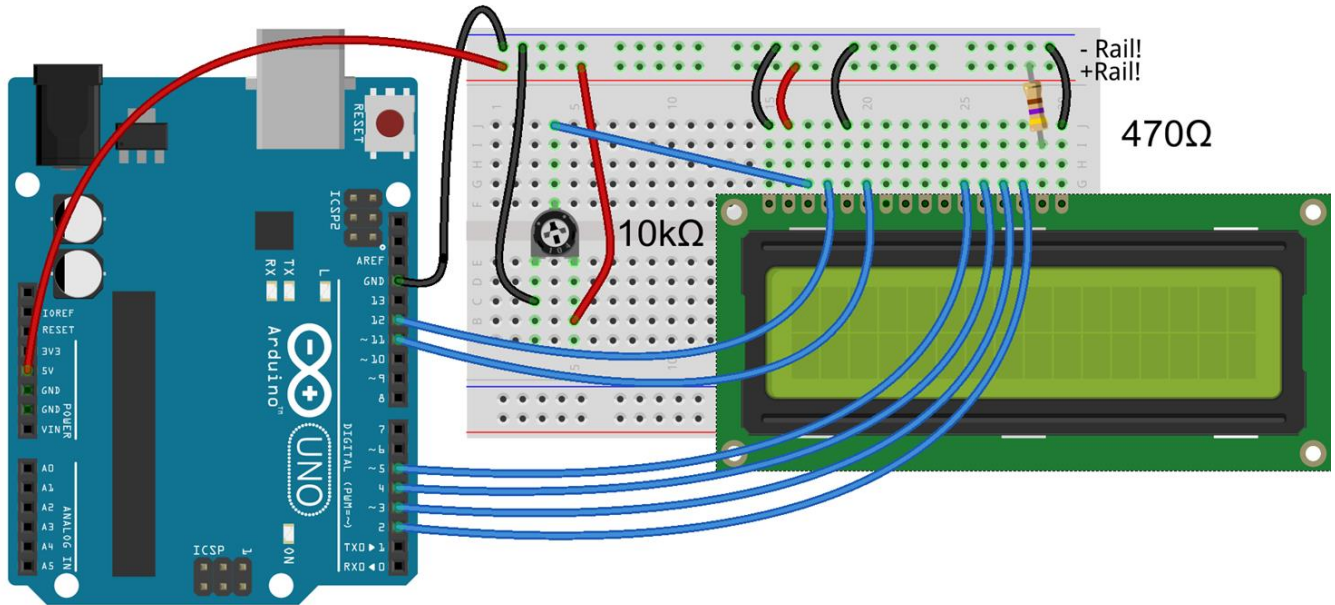
void dash()
{
  tone(7,800);           // toon aan
  delay(3 *t);          // toon 3 keer t ms (dash) aan
  noTone(7);           // schakel toon uit
  delay(t);            // symbool spatie
}
```



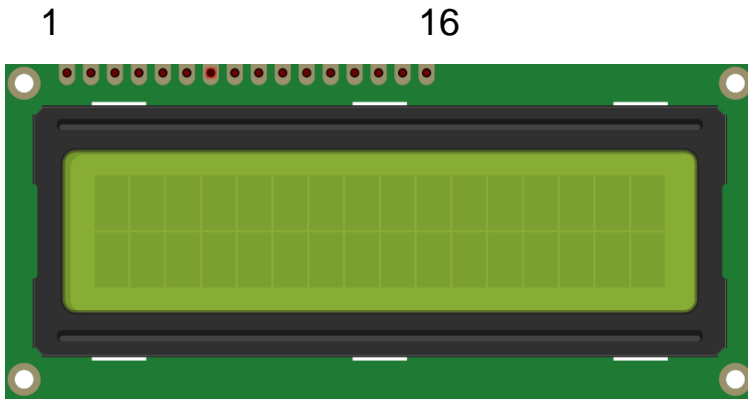
tone_2_ZLsketch_may27a \$

```
1 //programma voor het seinen van de letter r {dot, dash, dot}
2 // in dit programma worden subroutines gebruikt
3
4 int t = 200;           // lengte van dot; sein snelheid
5
6 void setup()
7 {
8   pinMode(7, OUTPUT); // initialiseer de LS pin als output
9 }
10
11 void dot()
12 {
13   tone(7, 800);       // toon aan
14   delay(t);          // toon t ms {dot} aan
15   noTone(7);         // schakel toon uit
16   delay(t);          // symbool spatie
17 }
18
19 void dash()
20 {
21   tone(7, 800);       // toon aan
22   delay(3 * t);       // toon 3 keer t ms {dash} aan
23   noTone(7);         // schakel toon uit
24   delay(t);          // symbool spatie
25 }
26
27 void loop()
28 {
29   dot();              // roep subroutine dot aan
30   dash();             // roep subroutine dash aan
31   dot();              // roep subroutine dot aan
32   delay(1000);
33 }
```

Oefening 4: Het LCD



LCD aansluitingen



LCD	Functie
1	VSS 0 volt
2	VDD 5 volt voeding
3	VO contrast instelling
4	RS Register Select
5	RW Read/Write
6	E Enable
7	D0 data signaal 0
8	D1 data signaal 1
9	D2 data signaal 2
10	D3 data signaal 2
11	D4 data signaal 4
12	D5 data signaal 5
13	D6 data signaal 6
14	D7 data signaal 6
15	A LED verlichting +
16	K LED verlichting -

LCD en 'for' loop

#include <LiquidCrystal.h>

Het programma wordt geïnstrueerd om de functie 'LiquidCrystal.h' uit de bibliotheek te gebruiken.

Enkele instructies van deze bibliotheek:

LiquidCrystal lcd(12, 11, 5, 4, 3, 2); LCD RS pin aan digital pin 12, LCD Enable pin aan digital pin 11, LCD D4 pin aan digital pin 5, LCD D5 pin aan digital pin 4, LCD D6 pin aan digital pin 3, LCD D7 pin aan digital pin 2

lcd.begin(16, 2); LCD heeft 16 posities per rij en 2 rijen

lcd.setCursor(x, y); De cursor wordt op positie x van regel y gezet (bijvoorbeeld 0,0)

lcd.print("tekst"); Vanaf de huidige cursor positie wordt het woord 'tekst' op het LCD geschreven

for (i = 0 ; i < 10 ;) {i ++.....}; wat andere vorm **for (int i = 0 ; i < 10 ; i++) { };**

Deze code wordt gebruikt om bewerkingen te herhalen

Het eerste stuk wordt eenmalig aan het begin uitgevoerd (i = 0).

Als i kleiner is dan 10 dan wordt de code tussen de haakjes uitgevoerd, als i niet kleiner is dan 10 dan wordt het 'for' statement beëindigd

i++ is een verkorte notatie voor i = i+1.



sketch_feb19f_LCD_oefening

```
// Programma LCD oefening
#include <LiquidCrystal.h>
int i=0;
// LCD RS pin aan digital pin 12 // LCD Enable pin aan digital pin 11
// LCD R/W pin aan ground // LCD D4 pin aan digital pin 5
// LCD D5 pin aan digital pin 4 // LCD D6 pin aan digital pin 3 // LCD D7 pin aan digital pin 2
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup()
{
  lcd.begin(16, 2); //LCD type 16 x 2
  lcd.setCursor(0, 0); //zet cursor op positie 0 (vooraan) en rij 0 (eerste regel)
  lcd.print("hello, world!"); //schrijf 'hello, world' op LCD
}
void loop()
{
  for( i=0; i<10;)
  {
    lcd.setCursor(0, 1); //zet cursor op positie 0 (vooraan) en rij 1 (tweede regel)
    lcd.print(i); //schrijf I op het LCD
    delay(1000); //wacht 1 sec
    i++; //hoog i met 1 op
  }
}
```