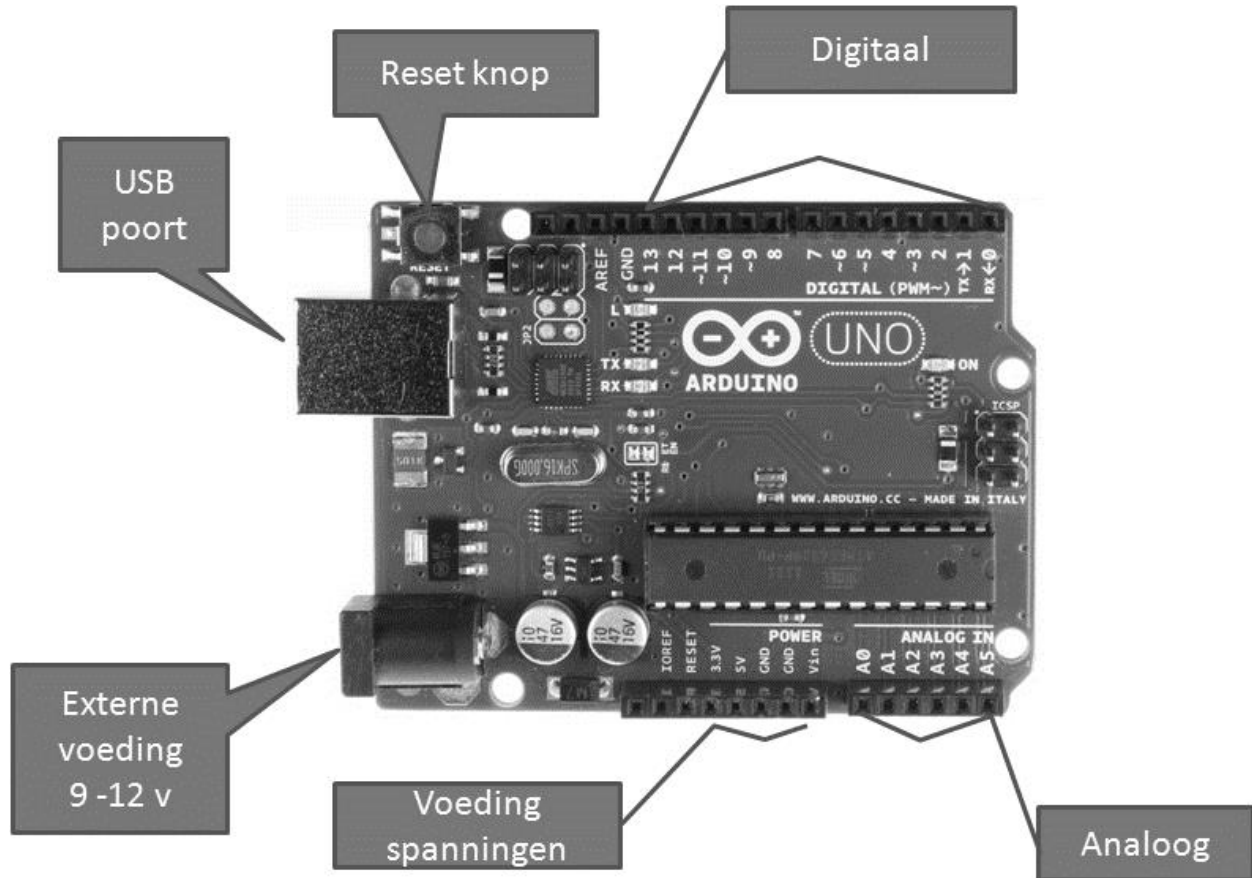


Werkbladen voor Arduino workshop 1

De aansluitingen van de Arduino Uno



Arduino

Arduino is een open-source ontwikkelbord, waarmee het eenvoudig is om informatica aan fysieke objecten te koppelen. Je kunt er robots mee bouwen maar ook je antenne mee besturen. Je kunt hem prima inzetten in transceivers voor besturing. De Arduino is gemakkelijk te programmeren en er is veel software beschikbaar voor allerlei besturings zaken.

De Arduino programmeertaal

void setup() {.....}

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

De praktijk:

void setup()

```
{
    code
    code
    .....
}
```

void loop() {.....}

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

pinMode(pin, mode);

Met pinMode definieer je een bepaalde pin(nummer) als in of uitgang. Mode kan INPUT of OUTPUT zijn.

digitalWrite(pin, output);

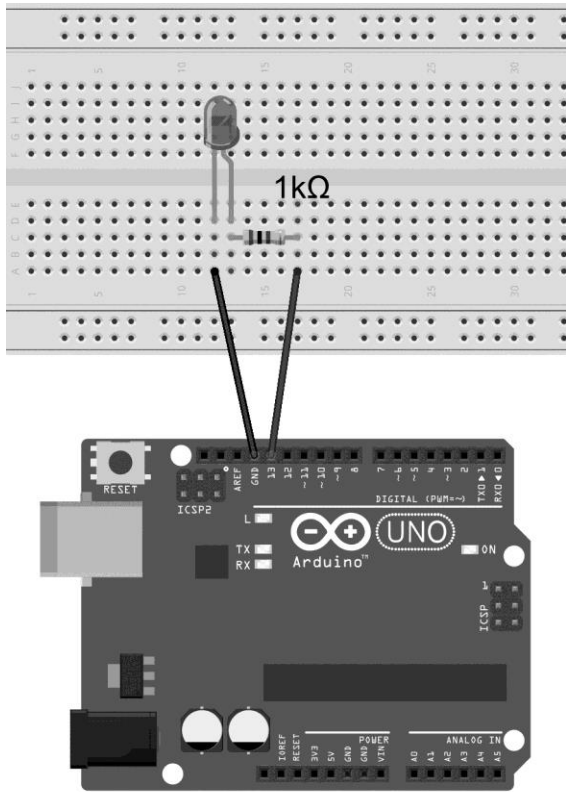
Hiermee bepaal je de digitale waarde van een bepaalde pin(nummer). Output kan HIGH of LOW zijn

delay(time);

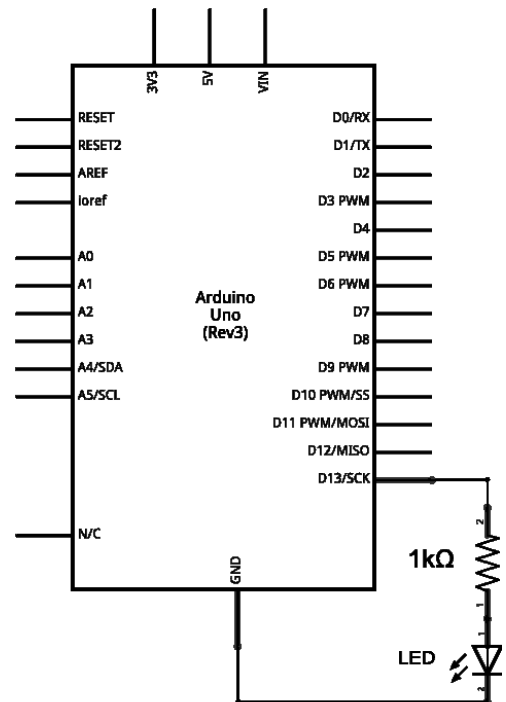
Hiermee laat je het programma pauzeren, de tijd wordt aangegeven in milliseconden

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

Oefening 1: De knipperende LED



Note: weerstand mag ook 470 Ohm zijn



```
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 = 1 seconde
  digitalWrite(13, LOW); // pin 13 wordt laag; de led gaat uit
  delay(1000);           // wacht 1000 milliseconden
}
```

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

Aan de integer variabele x **wordt** de waarde 13 toegekend.

Integers zijn primaire datatypes om getallen te bewaren zonder een decimale punt, een 16-bit waarde met een bereik van 32767 tot -32768.

Aanpassing van voorgaand programma:

// LED knipperprogramma

int ledpin = 13; // de integer variabele ledpin wordt 13

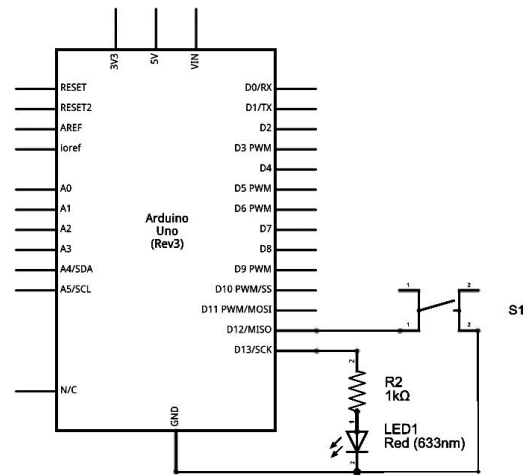
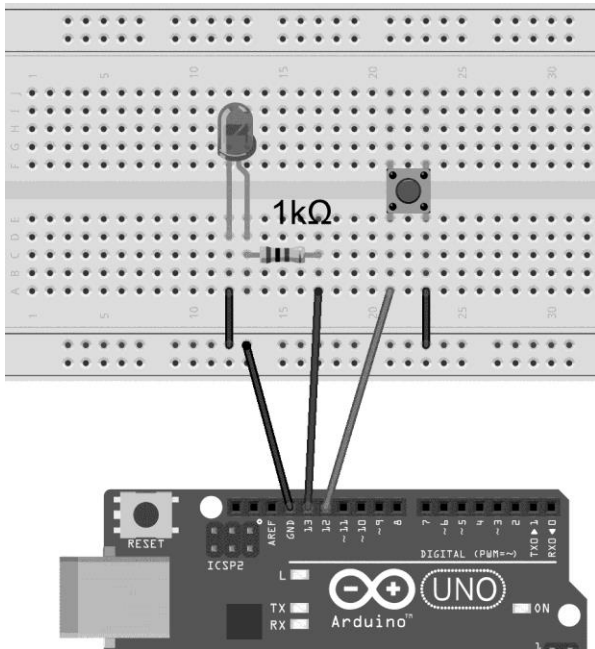
void setup()

```
{  
  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



Note: weerstand mag ook 470 Ohm zijn

digitalWrite(inputpin, HIGH)

Er wordt een pull-up (naar de plus 5 volt) weerstand geactiveerd waardoor deze inputpin 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 (condition) waar is, anders voert hij de code tussen na 'else' uit.

== Betekent 'is gelijk aan' (niet te verwarren met de enkele '=' want die kent een variabele een waarde toe).

// Drukknop oefening

// benoem de pin nummers:

```
int dkpin = 12; // pin nummer voor de drukknop
```

```
int ledpin = 13; // pin nummer voor de LED
```

```
void setup()
```

```
{
```

```
  pinMode(dkpin, INPUT); // initialiseer de drukknop pin als input
```

```
  pinMode(ledpin, OUTPUT); // initialiseer de LED pin als output
```

```
  digitalWrite(dkpin, HIGH); // activeer pull-up weerstand
```

```
}
```

```
void loop()
```

```
{
```

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

```
  {
```

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

```
  }
```

```
  else
```

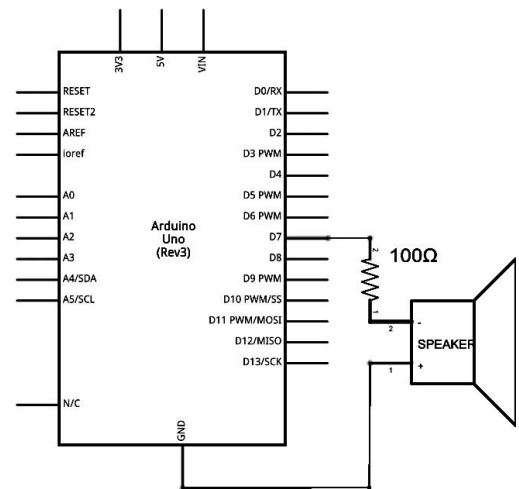
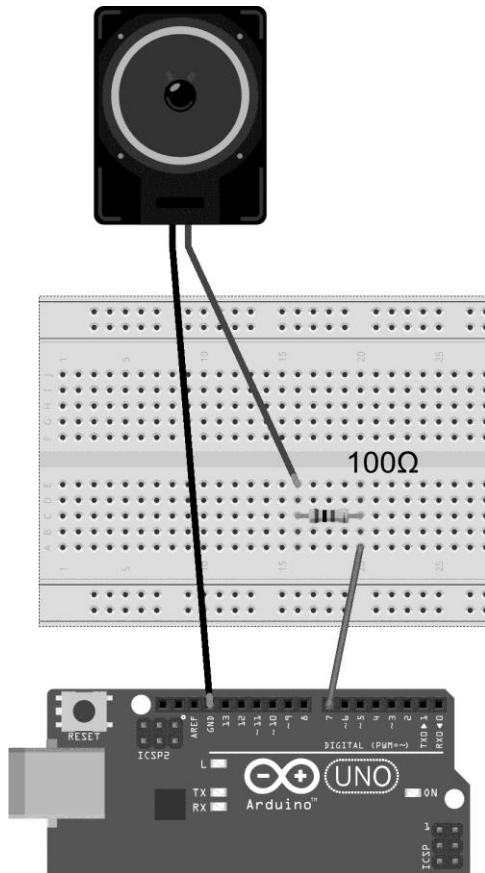
```
  {
```

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

```
  }
```

```
}
```

Oefening 3: Toon opwekking



Note: weerstand mag ook 470 Ohm zijn

tone(pin, frequentie); Er wordt een toon signaal (50% duty cycle) op uitgang 'pin' gegenereerd

noTone(pin); Stop de toon generatie op de uitgang 'pin'

// Oefenprogramma 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);
}
```

Gebruik maken van subroutines

```
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
}
```


//programma voor het seinen van de letter r (dot, dash, dot)

// in dit programma worden subroutines gebruikt

```
int t=200;          // lengte van dot; sein snelheid

void setup()
{
  pinMode(7, OUTPUT);    // initialiseer de LS pin als output
}

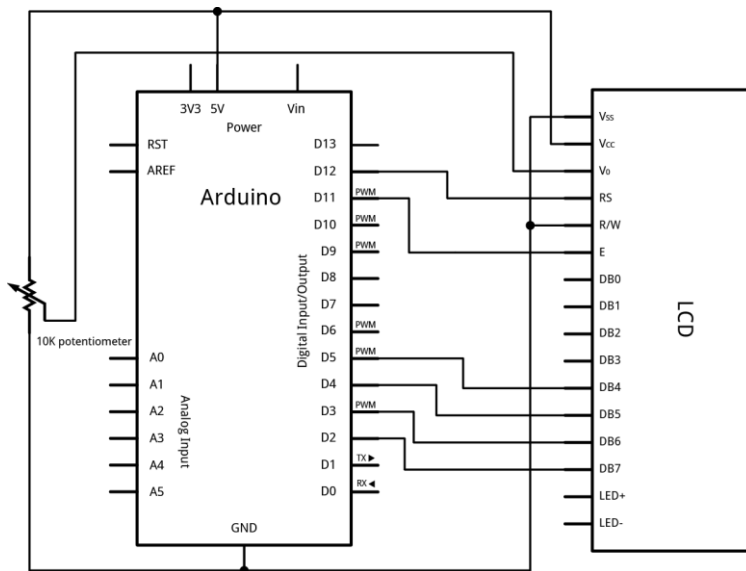
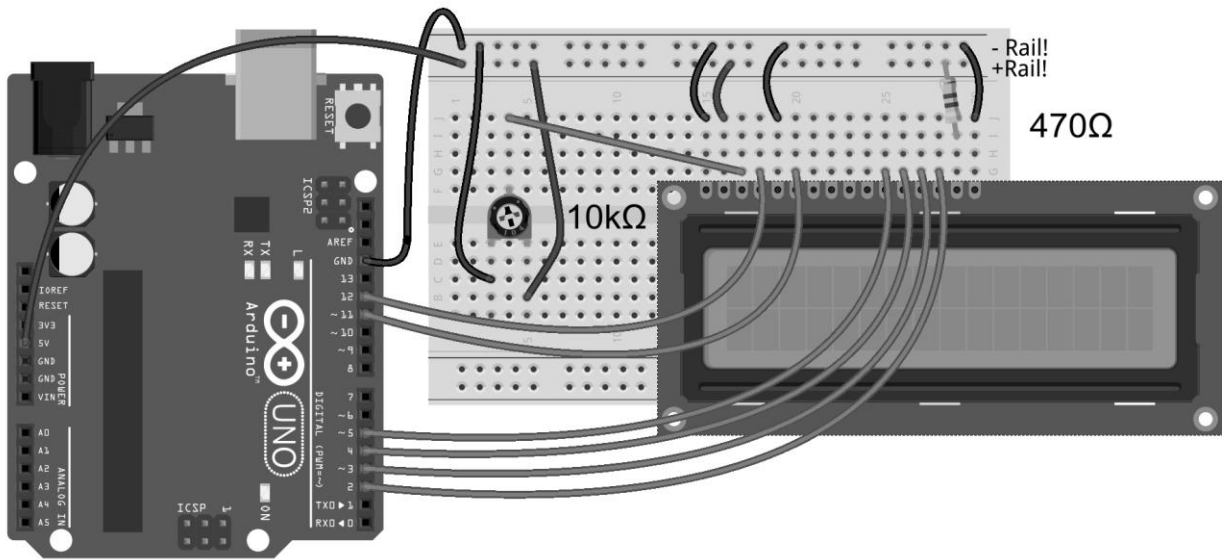
void dot()
{
  tone(7,800);          // toon aan
  delay(t);             // toon t ms (dot) aan
  noTone(7);           // schakel toon uit
  delay(t);             // symbool spatie
}

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
}

void loop()
{
  dot();                // roep subroutine dot aan
  dash();               // roep subroutine dash aan
  dot();                // roep subroutine dot aan
  delay(1000);
}
```

Als het werkt verander dan de seinsnelheid door de variable t=100 te maken

Oefening 4: Het LCD



#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)

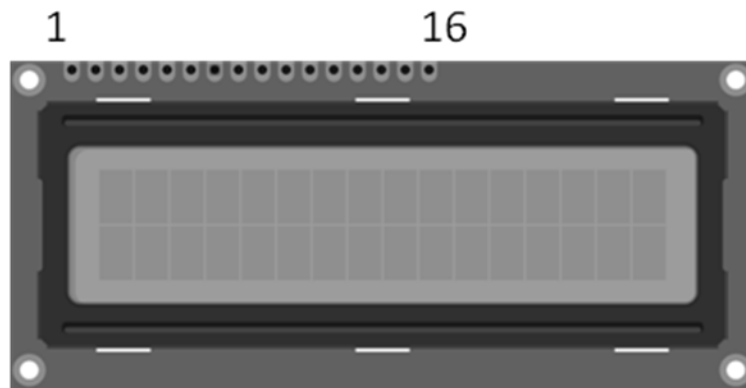
lcd.print("tekst"); Vanaf de huidige cursor positie wordt het woord 'tekst' op het LCD geschreven

for(i=0;i<10;) {.....}

Deze code wordt gebruikt om bewerkingen te herhalen Het eerste stuk wordt aan het begin uitgevoerd (int i = 0). Wanneer het 2e stuk **true** is voert hij de code tussen de haakjes uit (i<10).

i++; // i=i+1.

LCD aansluitingen



LCD	Functie	Aansluiting voor LCD oefening
1	VSS 0 volt	0 volt
2	VDD 5 volt voeding	5 volt
3	VO contrast instelling	Loper 10 k potmeter
4	RS Register Select	Pen 12 Arduino
5	RW Read/Write	0 volt
6	E Enable	Pen 11 Arduino
7	D0 data signaal 0	NC
8	D1 data signaal 1	NC
9	D2 data signaal 2	NC
10	D3 data signaal 2	NC
11	D4 data signaal 4	Pen 5 Arduino
12	D5 data signaal 5	Pen 4 Arduino
13	D6 data signaal 6	Pen 3 Arduino
14	D7 data signaal 6	Pen 2 Arduino
15	A LED verlichting +	Via 470 ohm naar + 5 volt
16	K LED verlichting -	0 volt

// Programma LCD oefening

```
#include <LiquidCrystal.h>

int i=0;

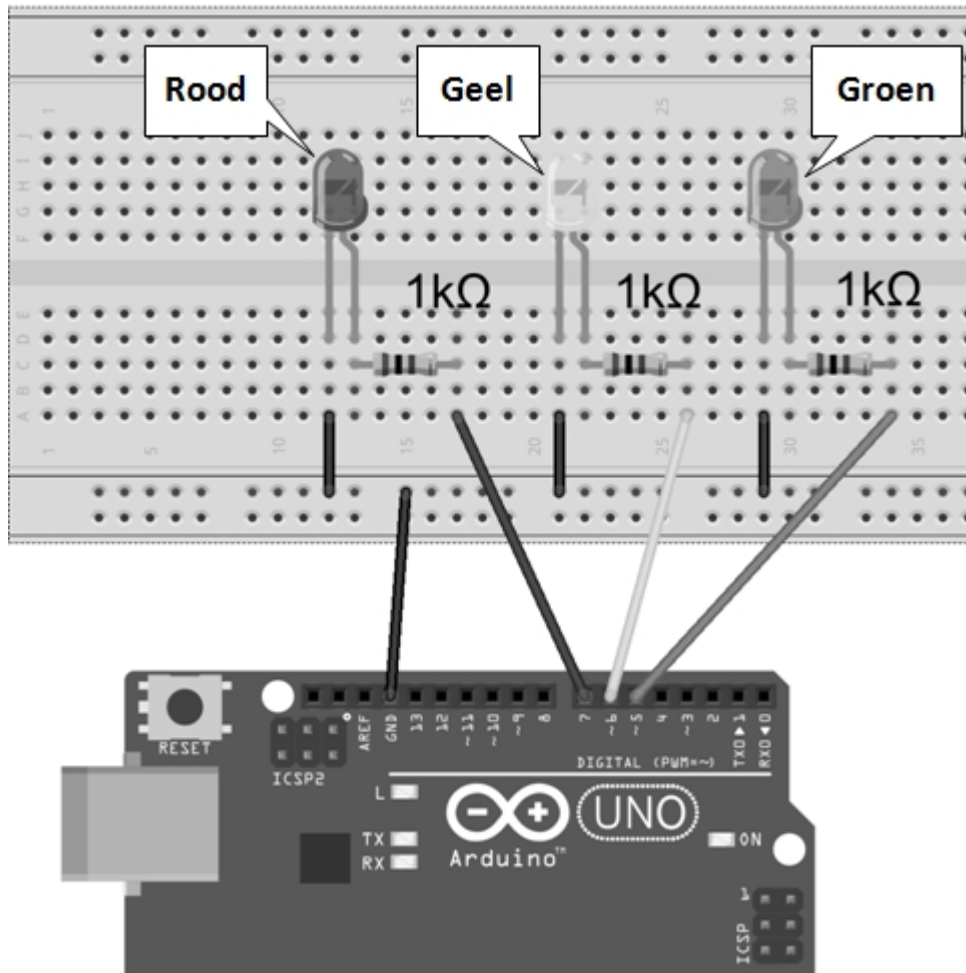
// LCD RS pin aan digital pin 12
// LCD Enable pin aan digital pin 11
// LCD R/W pin aan ground
// LCD D4 pin aan digital pin 5
// LCD D5 pin aan digital pin 4
// LCD D6 pin aan digital pin 3
// LCD D7 pin aan digital pin 2

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 de waarde van i op het LCD
        delay(1000);        // wacht 1 sec
        i++;                 // hoog i met 1 op
    }
}
```

Huiswerk



Note: weerstanden mogen ook 470 Ohm zijn

Huiswerk oefening 1: Maak een verkeerslicht

Dus eerst de rode LED aanzetten, laat deze LED 2 seconden branden, doof dan de rode LED, dan de gele LED aanzetten, enz...

Sluit de leds aan zoals in het plaatje is aangegeven, weerstandwaarden van 470 Ohm i.p.v 1 kilo-ohm zijn natuurlijk ook goed. Denk in je programma er aan dat de LEDs aan andere pennen aangesloten zijn dan bij de eerste knipper LED oefening.

Als extra opdracht kan ook nog een 'Duits' verkeerslicht gemaakt worden.

Huiswerk oefening 2: Maak een morse aanroep generator

Deel 1: Laat de Arduino de volgende tekens seinen: CQ. Het seinen wordt gestart met de drukknop. Gebruik het schema van de drukknop oefening maar vervang de LED door een luidsprekertje en vervang de weerstand door een met de waarde 470 Ohm.

Deel 2: Sein nu 'CQ de PI3ZLB' driemaal achter elkaar met aan het einde het een 'k'. Als er een toontje uit de luidspreker komt moet er ook een LED aangesloten op pin 13 gaan branden.

Huiswerk oefening 3: Spelen met tekst en getallen op het LCD

Deel 1: Maak een programma dat op het LCD op de **tweede** regel de tekst **workshop** schrijft en op de **eerste** regel **in het midden** een getal. Dit getal moet beginnen met 0 en automatisch met 1 ophogen. Als het getal 9 is wordt weer afgeteld tot 0 en daarna weer verhoogd tot 9 enz. Gebruik het schema als bij de LCD oefening.

Deel 2: Dezelfde oefening, maar nu tellen tot 15 en dan weer omlaag tot 0 enz. Kijk goed wat je nu ziet op het LCD en probeer dat in orde te krijgen.....

Extra oefening voor gevorderden 1: looplicht

Maak een looplicht met tenminste 4 leds. Telkens 1 led aan en heen en weer, dus 1,2,3,4,3,2,1,2,3,4 enz.

Extra oefening voor gevorderden 2: contact dender

Vervolg op de drukknop oefening. De drukknop oefening moet nu als volgt werken: als je eenmaal op het knopje drukt gaat de led aan, druk je nu weer op het knopje dan gaat de led uit; enz. Opmerking: bij het openen en sluiten van het contact van het drukknopje treedt er contact dender op. Daar zal in het programma rekening mee moeten worden gehouden. Bedenk hier zelf een oplossing voor of zoek op het internet hoe anderen dit oplossen.