§4 een arduino programmeren, de basis

Als eerste gaan we de software klaar maken voor de arduino. Volg hiervoor de volgende stappen.

1) Open de arduino IDE

2) Ga naar tools

3) Selecteer arduino uno bij Board

4) Selecteer de juiste poort onder Tools->Port (als er meerdere zijn probeer ze dan uit)

5) Ga naar tools en klik op Serial Monitor

6) Plak de volgende code in het arduino bestand

void setup() {

Serial.begin(9600);

}

void loop() {

Serial.print("het werkt");

Serial.print("\n");

delay(200);

}

7) Klik op het pijltje in de donker blauwe balk (naast het check teken en onder de witte balk), dit uploadt de code naar de arduino.

8) Kijk in de serial console of er “het werkt” komt te staan.

9) Als het niet werkt controleer of de arduino wel goed ingestoken is en of de serial monitor wel of 9600Baud is ingesteld.

Dan nu wat uitleg over de basis van het programmeren voor arduino.

De structuur van de code bestaat uit twee grote delen, de setup en de loop. De setup wordt maar een keer uitgevoerd en daarna wordt de loop uitgevoerd en deze blijft constant uitgevoerd worden. Je kunt hierboven in de code zien hoe dit in elkaar zit.

Dan nu een paar basis “commando’s” voor de arduino.

Met Serial.print() kun je waardes en tekst weergeven in de serial console, als je tekst wil weergeven, vergeet dan niet om “ ” toe te voegen.

Met pinMode() kun je een van de pinnen op de arduino in een bepaalde stand zetten, de opties zijn INPUT en OUTPUT. Wil je een signaal ontvangen, gebruik dan INPUT, wil je iets versturen, gebruik dan OUTPUT. Gebruik pinMode() als volgt, zet als eerste het nummer van de pin tussen de haakjes, deze kun je aflezen op de arduino. Blijf voorlopig bij de nummers 2 tm 13. Als tweede zet je stand erin. Een voorbeeld is pinMode(13, OUTPUT);

Om een signaal te lezen gebruik je digitalRead(). Je voert het nummer van de pin (die op de INPUT modus staat) in tussen de haakjes. Een voorbeeld is digitalRead(13);

Om een signaal te sturen gebruik je digitalWrite(). Je voert eerst het nummer van de pin in en dan de waarde. Mogelijke waardes zijn HIGH en LOW. HIGH kun je zien als 1 en LOW 0, dus aan en uit.

Je kunt ook variabelen gebruiken, dit zijn waardes met een naam. Als je een waarde vaker in jouw code nodig hebt komen deze erg van pas, ze helpen je het overzicht te bewaren en snel dingen aan te passen. Als je een getal als variabele wilt gebruik je int. Een voorbeeld is int ledPin = 13; Dit zet ledPin gelijk aan de waarde 13. Als je in de code nu ledPin gebruikt wordt dit vervangen door 13. Je kunt ook tekst als waarde geven. Een voorbeeld van een programma dat een string uitprint is:

void setup() {

//zet 'snelheid' signaal

Serial.begin(9600);

}

void loop() {

//stel een tekst in

String string = "Tekst";

Serial.println(string);

delay(200);

}

Als je tussen stukken code wilt wachten, dan kan dit via delay(), tussen de haken kun je het aantal milliseconden dat je wachten invoeren. Een voorbeeld is delay(200). Om 0,2s te wachten.

Dan komt nu een praktisch voorbeeld.

Als onderdelen heb je nodig: de arduino, een breadboard, een paar male-to-male kabels (met een uitstekende pin aan beide kanten), een resistor van 330ohm en een kleine LED.

Voordat je begint met het maken van het elektrisch circuit moet je de arduino uitzetten.

Laat een kabel van de 5V pin van de arduino naar de +rij van het breadboard gaan. Laat een kabel van de GND (grond/-) naar de -rij van het breadboard gaan. Zet nu een 330ohm resistor op het breadboard van de -rij naar de kortere pin van de LED gaan. Laat vervolgens een kabel van de +rij naar de lange pin van de LED gaan. Verbind de arduino met de computer. Als het goed hebt gedaan gaat nu het lampje aan. Als je een flits ziet, dan heb je de korte en lange pin omgedraaid of de verkeerde weerstand gebruikt, je hebt dan een nieuwe LED nodig.

Als laatste onderdeel van deze paragraaf gaan we wat code schrijven die de LED aan en uit laat gaan.

Dit is de code:

int ledPin = 13; //stel de pin in

void setup() {

pinMode(ledPin, OUTPUT); //zet de modus van de pin op OUTPUT om het signaal te kunnen sturen

}

void loop() {

digitalWrite(ledPin, HIGH); //zet de pin op aan, hierdoor kan er 5V door

delay(1000); //wacht 1s

digitalWrite(ledPin, LOW); //zet de pin uit, hierdoor gaat er 0V door

delay(1000); //wacht 1s

}

Kijk naar de comments (de tekst met // ervoor) voor de uitleg per regel. Probeer nu zelf het lampje aan en uit te laten gaan met een andere snelheid. Tip: gebruik als +pin hier pin 13 en niet de 5V pin.