

Arduino kleurensensor

Benodigheden:

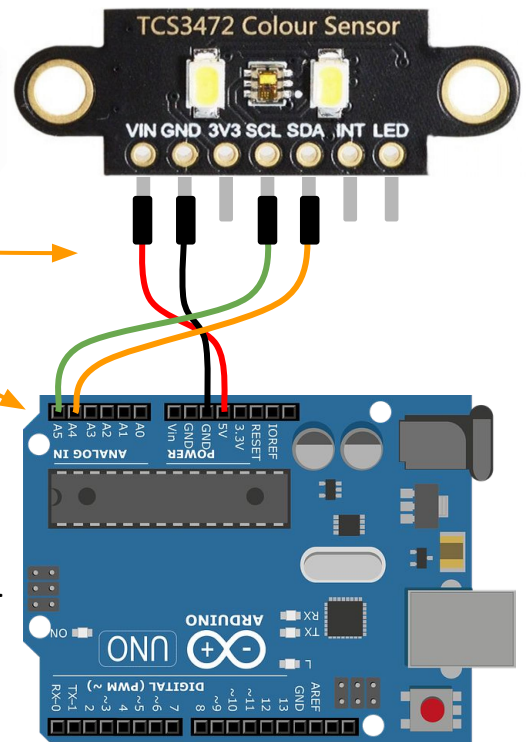
- ☐ 1x Laptop met Arduino omgeving
- ☐ 4x Female-Male dupont draadje
- ☐ 1x TCS3472 kleurensensor
- ☐ 1x Arduino Uno + USB kabel

1. Bouw het circuit zoals hiernaast

Kijk goed naar de afbeelding hiernaast om te zien hoe de vier draden verbonden moeten worden tussen de sensor en de Arduino.

De sensor wordt gevoed door de **5V** van de Arduino die aan de **VIN** van de sensor is verbonden. Door de **GND** van de sensor te verbinden met de **GND** van de Arduino wordt de stroomkring gesloten en kan er stroom gaan lopen.

De **SCL** en **SDA** signalen van de sensor worden gebruikt om de gemeten kleurwaarden naar de Arduino te sturen.




2. Schrijf de code


Open de Arduino omgeving en maak een nieuw project (linksboven “Bestand > Nieuw”).

Kijk goed naar de code op de volgende bladzijde en schrijf dit in de Arduino omgeving. Tussen de code staan ook comments (the tekst achter `//`) die vertellen wat een deel van de code doet. Alles op een regels achter `//` wordt niet uitgevoerd.

3. Test de code

Installeer de [Adafruit_TCS34725](#) library door de stappen bovenaan op de volgende bladzijde te volgen.

Sluit de Arduino met de USB op de computer aan. In de Arduino omgeving ga naar “Hulpmiddelen > Poort” en selecteer de poort waar achter staat “(Arduino Uno)”. Klik nu op de “upload” knop .

Als het uploaden is gelukt open dan de **seriële monitor** (de knop  rechtsboven). Nu zou je de gemeten RGB waardes op het scherm geprint moeten zien worden.

Verdere uitleg over de code

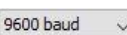
Een kleur meten

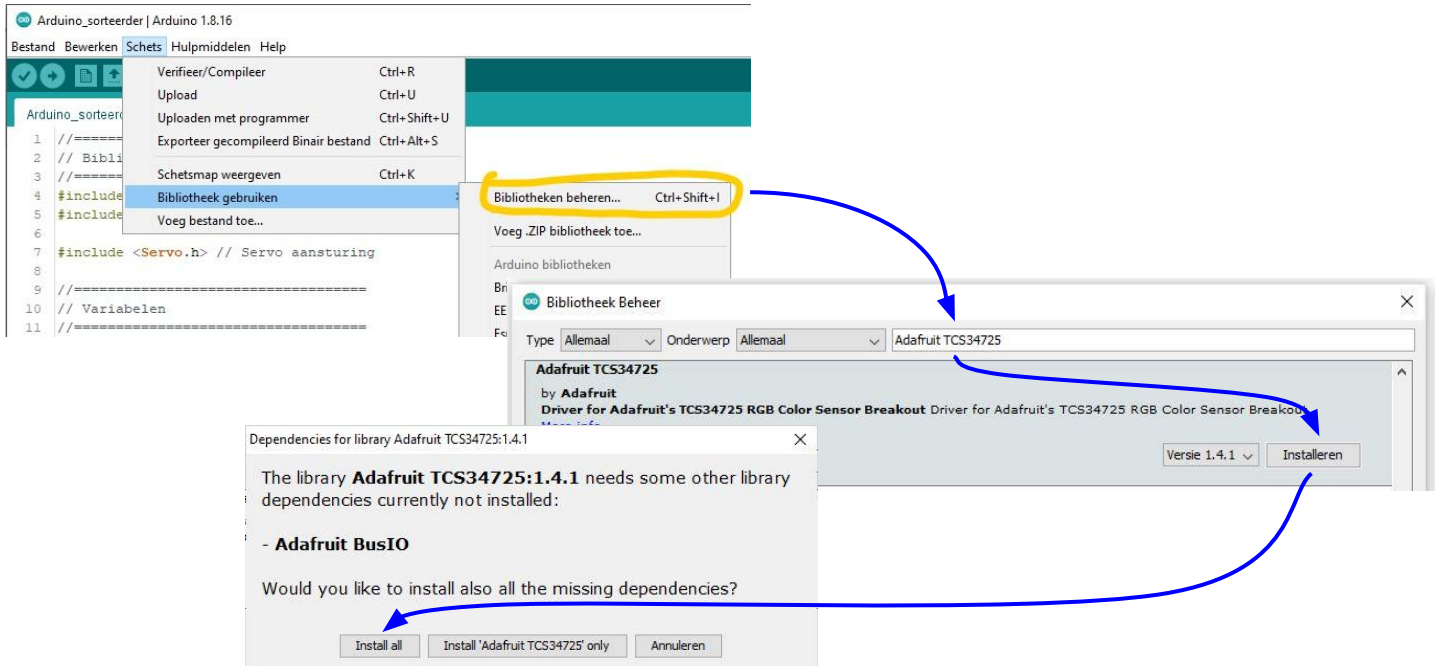
Op de regels **8**, **9** en **10** wordt er een sensor instantie gemaakt. Met de parameter/variabel **TCS34725_INTEGRATIONTIME_614MS** geven we aan elke meting van de kleurensensor 614ms (milliseconden) moet duren. Dit is dan ook de reden dat we op regel **29** een delay in de code hebben staan van 615ms voordat we een nieuwe meting opvragen van de sensor. De meting opvragen doen we met de functie **getRawData** die de gemeten RGB waardes in de **r**, **g** en **b** variabelen zet die we op de regels **14**, **15** en **16** hebben gemaakt.

RGB kleuren

In de computer wordt elke kleur met de drie kleuren Rood, Groen, Blauw (RGB) gemaakt. Daarom krijgen we van de sensor een rode, groene en blauwe waarde terug. Hiermee weten we hoeveel rood, groen en blauw er in de gemeten kleur zit. Op de volgende pagina zie je hoe je met een combinatie van rood, groen en blauw alle kleuren kan maken.

Tekst printen

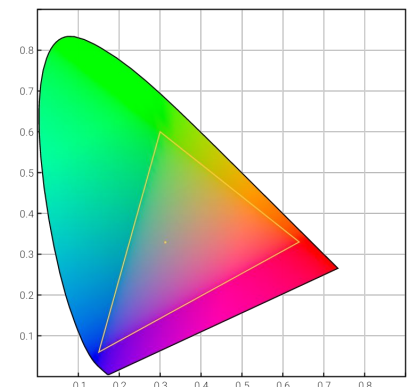
Op regel **37 t/m 40** gebruiken we de functies **Serial.print()** en **Serial.println()**. Met deze functies kan je tekst naar de **seriële monitor** schrijven. Om dit te kunnen doen moet je wel eerste **Serial.begin(9600)** hebben aangeroepen zoals hier op regel **21**. Hier is 9600 de snelheid. Deze snelheid moet overeenkomen met de snelheid in de **seriële monitor** zoals je die rechtsonder ziet ().



```

1. // We gebruiken de library (bibliotheek) om de kleurensensor te gebruiken
2. #include "Adafruit_TCS34725.h"
3.
4. // We maken een variabele aan waar we een instantie
5. // van de kleurensensor in opslaan.
6. //
7. // TCS34725_INTEGRATIONTIME_614MS betekend dat elke meting 614ms duurt
8. Adafruit_TCS34725 kleurensensor = Adafruit_TCS34725(
9.     TCS34725_INTEGRATIONTIME_614MS,
10.    TCS34725_GAIN_1X );
11.
12. // In deze variabelen gaan we de gemeten rood (r), groen (g) en blauw (b)
13. // waardes opslaan.
14. uint16_t r;
15. uint16_t g;
16. uint16_t b;
17. uint16_t c;
18.
19. void setup() {
20.     // Met Serial kunnen we tekst printen
21.     Serial.begin(9600);
22.
23.     // Start de kleurensensor
24.     kleurensensor.begin();
25. }
26.
27. void loop() {
28.     // Wacht totdat de meting klaar is
29.     delay(615);
30.
31.     // Zet de gemeten waarden in de r, g, b variabelen
32.     kleurensensor.getRawData(&r, &g, &b, &c);
33.
34.     // Print de gemeten waarden
35.     // print() : print tekst
36.     // println(): print tekst en ga daarna naar een nieuwe lijn.
37.     Serial.print("R: "); Serial.print(r, DEC); Serial.print(" ");
38.     Serial.print("G: "); Serial.print(g, DEC); Serial.print(" ");
39.     Serial.print("B: "); Serial.print(b, DEC); Serial.print(" ");
40.     Serial.println(" ");
41.
42. }

```



src:
<https://developer.android.com/reference/android/graphics/ColorSpace.Named?hl=es-419>