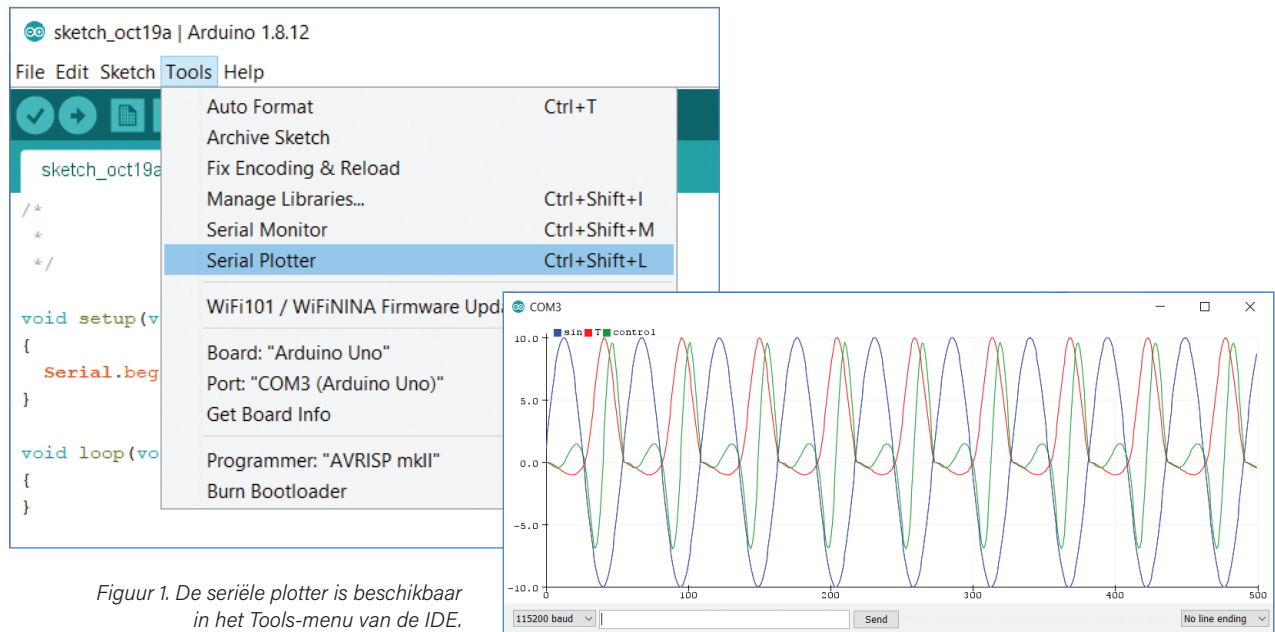


Het gebruik van Arduino's seriële plotter

grafieken plotten met Arduino is gemakkelijk



Figuur 1. De seriële plotter is beschikbaar in het Tools-menu van de IDE.

Clemens Valens (Elektor)

Een van de sterke punten van Arduino is de eenvoudig te gebruiken seriële poort, die het eenvoudig maakt om gegevens te exporteren, opdrachten te versturen of te helpen bij het debuggen. Voor het visualiseren van deze data zijn een seriële monitor en een seriële plotter beschikbaar. De eerste spreekt voor zich, maar hoe gebruikt u de seriële plotter?

De meeste Arduino-gebruikers kennen de seriële monitor die in de IDE is ingebouwd (beschikbaar via het **Tools**-menu, of met de sneltoets **Ctrl+Shift+M**). Dit is een eenvoudige seriële terminal die alle via de seriële poort ontvangen gegevens weergeeft. U kunt bovendien strings (bijvoorbeeld commando's of gegevens) van de computer naar het Arduino-board sturen. De seriële monitor kan handig zijn bij het debuggen door de sketch tekstberichten te laten versturen over wat hij doet of wat hij verondersteld wordt te doen.

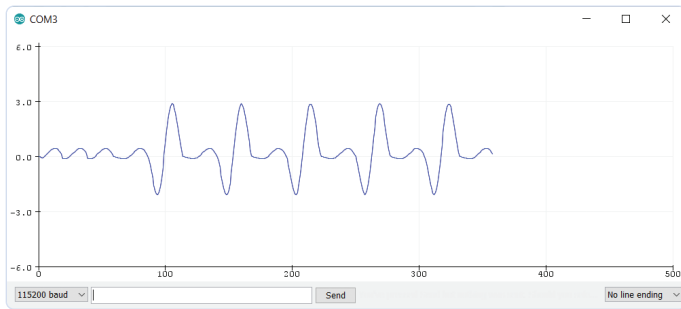
De seriële plotter

De seriële monitor kan natuurlijk ook numerieke waarden weergeven. Het volgen van het verloop van een parameter die als een lijst met waarden wordt weergegeven is niet altijd gemakkelijk, en daarom

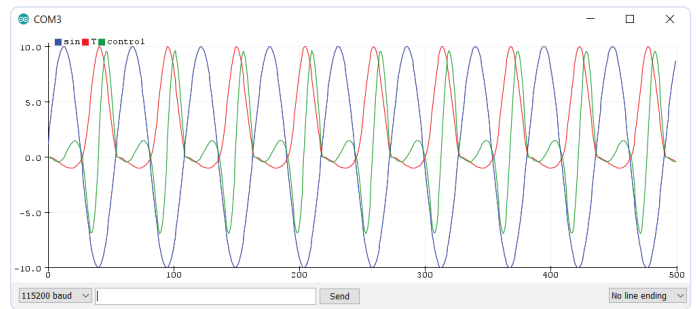
heeft de Arduino IDE ook een seriële plotter (**Serial Plotter**, in het **Tools**-menu of met de sneltoets **Ctrl+Shift+L**, zie **figuur 1**).

De seriële plotter maakt grafieken van de numerieke gegevens die hij ontvangt en geeft deze weer. Hij is niet extreem krachtig en heeft niet veel opties, maar is eenvoudig in het gebruik. Helaas is er geen documentatie voor. U moet de Java-broncode [1] raadplegen om te weten te komen waartoe hij in staat is. Op die manier ontdekt u de volgende mogelijkheden:

- onbeperkt aantal grafieken;
- automatische schaling van de verticale of Y-as;
- vijfhonderd punten op de horizontale of X-as;
- toont de meest recente 500 datapunten.



Figuur 2. De horizontale as heeft ruimte voor maximaal 500 punten. Als u meer zendt, begint hij te scrollen.



Figuur 3. Het aantal grafieken is onbeperkt (in theorie). Gebruik labels om bij te houden welke grafiek welke gegevens weergeeft.

Een grafiek plotten

Kies het openen van de seriële plotter dezelfde baudrate als de snelheid die in `Serial.begin` in de sketch is aangegeven. Het vakje naast het snelheidskeuzevak is bedoeld voor het terugsturen van gegevens naar de sketch, precies zoals in de seriële monitor. Zo kunt u de schets tot op zekere hoogte sturen.

U hoeft geen speciale commando's te leren om de seriële plotter te gebruiken, hij maakt gebruik van dezelfde seriële poort-commando's als de seriële monitor. De seriële plotter is gewoon een andere manier om de gegevens te visualiseren die via de seriële poort worden verzonden. Het enige belangrijke is de manier waarop u de gegevens formatteert. Om een grafiek van een enkele parameter te plotten (**figuur 2**), een sensorwaarde bijvoorbeeld, is het voldoende om de waarden te scheiden door een nieuwe regel-karakter. Met andere woorden, om de waarden te verzenden gebruikt u

```
Serial.println(sensor_value);
```

Hoe zit het met de schaal?

De seriële plotter gebruikt deze waarden voor de verticale as, de Y-as; hij vergroot de horizontale of X-as zelf. Met andere woorden, de eerste verzonden waarde wordt beschouwd als de 'y'-waarde voor $x = 0$, de tweede waarde is de 'y'-waarde voor $x = 1$, en zo vervolgens. Wanneer x de waarde 500 bereikt, begint de grafiek horizontaal te scrollen; nieuwe punten worden aan het eind toegevoegd, de oude punten vallen aan het begin over de rand.

Als de y-waarde buiten het bereik valt, wordt de verticale as automatisch opgeschaald om alles binnen de verticale grenzen te houden. Hetzelfde geldt ook omgekeerd – dus als alle waarden plotseling in een kleiner bereik passen (omdat een hoge waarde aan het begin wegvalt omdat de grafiek aan het scrollen is – wordt de schaal navenant aangepast.

Meerdere grafieken plotten

Om grafieken voor twee of meer parameters te tekenen, stuurt u de waarden voor de verschillende grafieken als een lijst met waarden die worden gescheiden door een komma (`,`), een spatie (` `) of een tab (`\t`) en sluit u de lijst af met bijvoorbeeld een nieuwe regel-teken:

```
Serial.print(temperature);
Serial.print(',');
Serial.print(humidity);
```

```
Serial.print(',');
Serial.print(pressure);
Serial.println();
```

Bij gebruik van de komma als scheidingsteken staat het formaat bekend als 'CSV' (*Comma Separated Values*). De meeste spreadsheets kunnen dergelijke bestanden importeren (bij de betere spreadsheets kunt u zelf het scheidingsteken opgeven), dus als het er goed uitziet in de seriële plotter kunt u de seriële monitor gebruiken om (veel van) deze regels te exporteren en naar een spreadsheet te kopiëren om mooie grafieken te genereren of er andere leuke dingen mee te doen.

De lijst met waarden wordt geïnterpreteerd als y-waarden voor een en dezelfde x-waarde, dus alle grafieken worden op hetzelfde moment en met dezelfde snelheid bijgewerkt. De plotter kiest voor elke grafiek een andere kleur, de gebruiker heeft hier geen controle over. De enige manier om dit enigszins te beïnvloeden is door de volgorde van de gegevens te veranderen.

Labels

De seriële plotter heeft één 'luke' optie: u kunt een label specificeren voor elke grafiek. Er zijn twee manieren om dit te doen:

- voordat u waarden verzendt, stuurt u eerst een lijst met labels die gescheiden zijn door een komma, een spatie of een tab en die eindigt met een nieuwe regel-teken. De labels zelf mogen deze tekens uiteraard niet bevatten. De gegevens moeten in dezelfde volgorde als de labels worden verzonden om ze te laten overeenstemmen. Voorbeeld:

```
Serial.println("temperature,humidity,pressure")
```

- stuur labels en waarden als paren en scheid de twee met een dubbele punt (`,`), bijvoorbeeld:

```
Serial.println("temperature:21,humidity:76,pressure:1007")
```

Zorg ervoor dat deze paren altijd in dezelfde volgorde worden verzonden om te voorkomen dat de plotter de gegevens voor verschillende grafieken door elkaar haalt. Een lijst van label-waarden moet opnieuw worden gescheiden door een komma, een spatie of een tab en eindigen met een nieuwe regel-teken.

Merk op dat zodra u een tekststring verstuurt, deze wordt beschouwd als een label. Het is dus vrij eenvoudig om de labels te verknoeien. Een andere manier om dit te doen is het vergeten van scheidingstekens, waardoor bijvoorbeeld gegevens in een label terecht komen.

Eigenaardigheden en beperkingen

- De horizontale as gaat eeuwig door, er is geen manier om de grafiek programmatisch te herstarten. De enige manier om dit te doen is om de seriële plotter te sluiten en vervolgens weer te openen. Soms is het zelfs nodig om dit twee keer te doen om oude labels en/of gegevens kwijt te raken.
- Het is niet mogelijk om een punt te plotten op een (x,y) positie naar keuze; nieuwe punten worden altijd aan het einde toegevoegd.
- De verticale assen schalen automatisch; dat lijkt leuk maar kan erg vervelend zijn, vooral als er uitschieters of glitches zijn die de schaal oprekken en het moeilijk of zelfs onmogelijk maken om de veel kleinere belangrijke gegevens te zien. Dat kan ook gebeuren als de ene gegevensreeks in een bereik ligt van bijvoorbeeld [-1,+1] en de andere in het bereik [-100,+100]. In dat geval zal de schaal van -100 tot +100 lopen. Sommige gebruikers proberen de plotter te slim af te zijn door iets als `min:-100,max:100` toe te voegen aan de lijst met waarden in een poging om de autoscaling te stoppen, maar dit werkt alleen als elke andere waarde binnen deze grenzen blijft (figuur 4).
- Om redenen die alleen bekend zijn bij de ontwikkelaar(s) van de seriële plotter loopt de minimale schaal meestal van -6 tot +6, maar -5 tot +5 kan ook voorkomen.

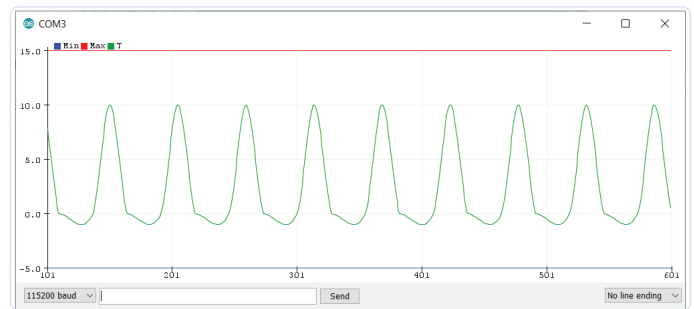
Helpfunctie

Hieronder staat een kleine helpfunctie die u in een sketch kunt invoegen om de seriële plotter iets gebruiksvriendelijker te maken en tegelijkertijd de sketch leesbaar te houden:

```
void plot(String label, float value, bool last)
{
    Serial.print(label); // May be an empty string.
    if (label!="") Serial.print(":");
    Serial.print(value);
    if (last==false) Serial.print(",");
    else Serial.println();
}
```

Vragen of opmerkingen?

Hebt u technische vragen of opmerkingen naar aanleiding van dit artikel? Stuur een e-mail naar de auteur of naar de redactie van Elektor, via redactie@elektor.com.



Figuur 4. Het versturen van constante minimum- en maximumwaarden vermijdt het automatisch schalen van de verticale as, maar alleen zolang alle andere waarden binnen deze grenzen blijven.

Als de string *label* is gespecificeerd, zal deze functie een label:waarde-paar afdrukken. Als het label een lege string is, dan zal het alleen de waarde afdrukken. Dit maakt het mogelijk om pure CSV-geformatteerde gegevens uit te voeren in de seriële monitor, goed voor gebruik met spreadsheets.

Merk op dat u de seriële monitor en de seriële plotter niet tegelijkertijd kunt openen.

Na het openen van de seriële poort wordt die als volgt aangeroepen:

```
plot("Temperature",temperature_value,false); // Graph1,
more graphs follow
plot("Humidity",humidity_value,false); // Graph2, more
graphs follow
plot("Pressure",pressure_value,true); // Graph3, last
graph
```

Vergeet niet de seriële plotter opnieuw te starten (afsluiten en opnieuw openen) telkens wanneer u de sketch wijzigt, om ervoor te zorgen dat alle verouderde gegevens en labels worden verwijderd. Herhaal deze procedure in geval van twijfel. ◀

200540-04

Een bijdrage van

Idee, tekst en illustraties:
Clemens Valens
Redactie: C.J. Abate

Vertaling: **Jelle Aarnoudse**
Layout: **Giel Dols**

WEBLINK

- [1] **Broncode van de seriële plotter van de Arduino IDE:**
<https://github.com/arduino/Arduino/blob/master/src/processing/app/SerialPlotter.java>