

● Meten en sturen met Arduino ●

Henk Siewert

Microcontrollerbordjes als de Arduino en vergelijkbare platformpjes zijn ideaal om allerlei sensoren uit te lezen en aan te sturen. Voor de wat mooiere en veelzijdigere kant van de software maak ik graag gebruik van Processing en Python. Processing biedt veel meer mogelijkheden dan de seriële monitor van de Arduino of monitors als Putty en CuteCom. Ook zal ik aandacht besteden aan het VT100-protocol.

Even bijpraten

Dit stuk wordt getypt in LibreOffice Writer en niet, zoals ik gewend was, in Microsoft Word. Tot nu toe werkte ik veel met Windows 11. Hoewel, werken kon je het eigenlijk niet meer noemen. De hoeveelheid ellende die ik steeds weer ondervond na weer een update begon dergelijke vormen aan te nemen dat ik definitief heb besloten om Windows als primair systeem het raam uit te gooien.

Het is toch wel erg als je met een machine van nog geen jaar oud met 16 Gig RAM, 500 Gig SSD en een processor met 12 kernen moet gaan zitten wachten tot Windows weer eens een update installeert en je ook maar af moet wachten of alles daarna weer werkt zoals je gewend was. Ik vind dat, als je dingen speciaal op een bepaalde manier hebt ingesteld, omdat het voor jou goed werkt, dit door het systeem moet worden gerespecteerd en niet na een update ongedaan moet zijn gemaakt.

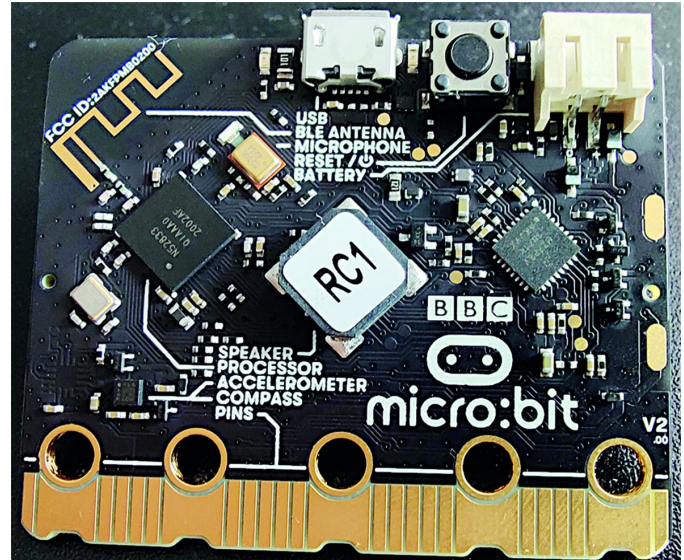
Dus werken? Nou meestal niet. Er was altijd wel weer iets veranderd. Of het deed het niet.

Na de laatste update weigerde het systeem nog iets te downloaden van het internet. Dat was de druppel die bij mij de emmer deed overlopen. Dus nu ben ik overgegaan op Linux Mint, op mijn primaire systeem, en Xubuntu 23 op mijn tweede systeem, waarop ik al het hardware-ontwikkelwerk doe. Ik heb via een VNC-verbinding de Xubuntu-computer in het tweede virtuele venster van de Mint-computer staan. Dus dat is lekker makkelijk werken. En ik kan natuurlijk via SSH en X11 direct een programma van de ene op de andere computer opstarten en op het scherm weergeven. Wil je daar meer over weten? Stuur dan een email. Dan maak ik daar ook een stukje over.

Het dumpen van Windows 11 betekent niet dat alle Windows projecten, zoals de ZX81 emulator EightyOne en dergelijke overboord gaan. Zeker niet. Ik heb VMWare Player geïnstalleerd en daarin twee virtuele machines aangemaakt, Windows XP Pro en Windows 11. Oh, wat een heerlijkheid om weer met XP te kunnen werken! Vooral Visual Studio 6 draait weer als een tierelier. En wat fijn om zonder gedoe MS-DOS programma's op XP te kunnen draaien. Geldt trouwens ook voor CP/M, via de Z80mu-emulator. Kom ik later nog wel op terug. Nu eerst verder met dit verhaal.

Arduino, Raspberry enzovoorts

Zoals eerder gezegd zijn de Arduino, Raspberry, Processing en Python heel geschikt om te sturen en te meten. Op het ogenblik beschik ik over een aantal Arduino Uno 3's, Een Uno 4 en een Uno 4 Wifi. Van de Raspberry heb ik de Pi 3, Pi 4 en de Zero W. Die gaan allemaal ingezet worden voor verschillende projecten die ontwikkeld worden. Waarschijnlijk komen daar nog wel een aantal soorten bij. Wat dacht je van de BBC-micro:bit?



BBC Micro:Bit in de nieuwe - versie 2 - uitvoering. Met een flink aantal extra's.

En dan zijn er natuurlijk nog de ESP32 en aanverwanten. En zo zijn er nog wel een aantal op te noemen. Wordt vast erg leuk om daar mee aan de slag te gaan. Alleen de Raspberry Pi Pico ga ik niet gebruiken. Voor serieus meetwerk is dat ding volkomen ongeschikt. Maar daarover heb je al in een eerder stuk van mij kunnen lezen.

Monitors

Om gegevens uit te lezen die van bijvoorbeeld sensoren terugkomen naar de microcontroller kun je gebruik maken van de in de verschillende IDE ingebouwde seriële monitors. Zoals bijvoorbeeld de monitor in de Arduino IDE.

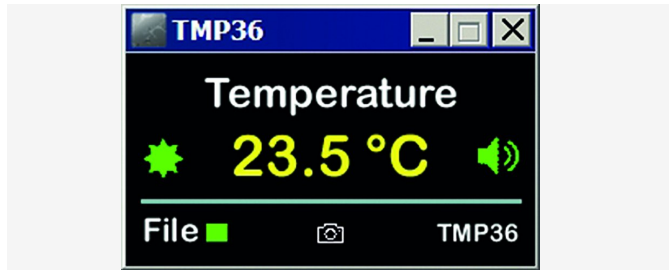
```

sketch_feb11a.ino
1 void setup() {
2   // put your setup code here, to run once:
3
4 }
5
6 void loop() {
7   // put your main code here, to run repeatedly:
8
9 }
10
Serial Monitor x
Message (Enter to send message to 'Arduino Uno' on '/dev/ttyACM0') New Line 9600 baud
24.6
24.7
24.6
24.7
24.6
Ln 1, Col 1 Arduino Uno on /dev/ttyACM0

```

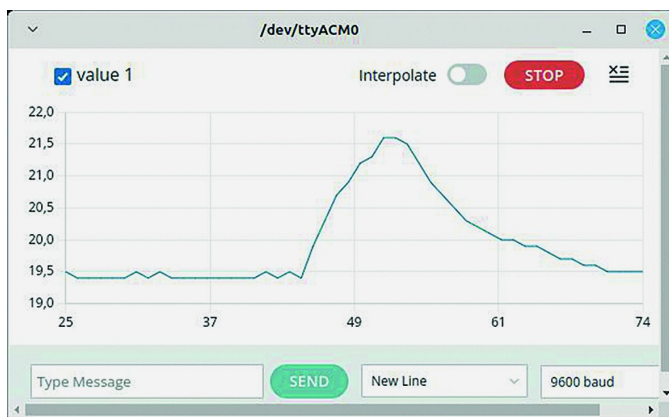
Arduino IDE seriële monitor met de waarden van een TMP36 temperatuur sensor. Niet echt een sexy uiterlijk.

Maar het kan, vooral grafisch gezien, een stuk mooier. Bijvoorbeeld via een programma gemaakt in Processing. In een vorig stuk in SoftwareBus 2024 -1, heb ik al een voorbeeldje gegeven van wat er mogelijk is.



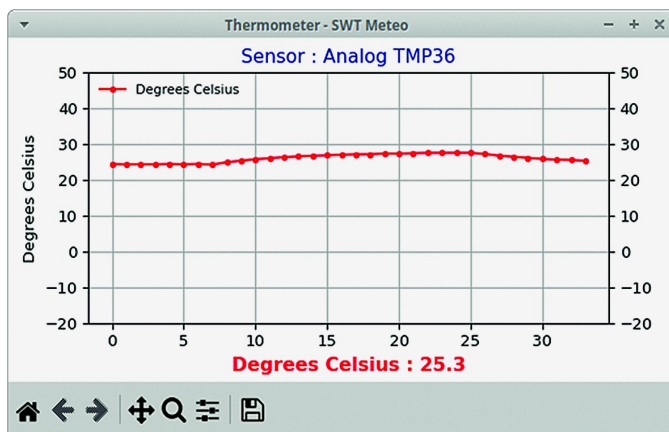
Een Processing programma voor het uitlezen van een TMP36 sensor. Ziet er, volgens mij, toch een stuk beter uit. Op de afbeelding draait het programma onder Windows.

Soms is het handig, vooral als je het verloop van metingen in de tijd wilt weergeven, gebruik te maken van een realtime grafiek. De Arduino IDE biedt die mogelijkheid.



Meetwaarde TMP36 weergave in de grafiek van de Arduino IDE.

Maar je kunt ook in Python werken en zelf de grafiek vormgeven. Dan kun je zelf bepalen wat, en hoe je de gegevens op het scherm wilt hebben.



Realtime grafiek gemaakt in Python.

Lijkt je dat wat? OK, dan gaan we aan de slag. O ja, meedenkers, programmeerders en andere creatievelingen die mee willen doen zijn van harte welkom. Stuur maar een email. Kom, stook je soldeerbout op, slijp je potlood, til de kat van je toetsenbord en stof je monitor af. Dan gaan we op avontuur.

Installeren

Voor alle duidelijkheid, alle beschrijvingen zijn gebaseerd op het werken op Linux. In mijn geval op de Xubuntu 23-distributie.

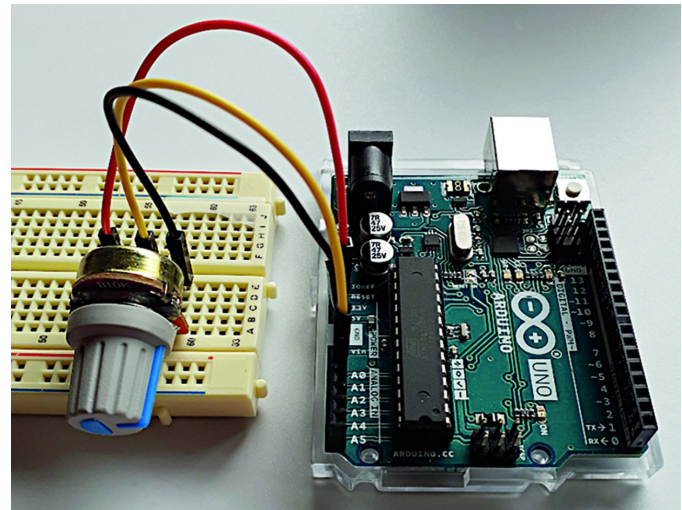
Maar ik denk dat deze beschrijvingen op alle Debian- en Ubuntu-distributies zullen werken. Ik heb begrepen dat de gloednieuwe HCC Linux interessegroep Linux Mint heeft omarmd. Dus daar zal alles ook op worden getest.

Om te kunnen programmeren heb je gereedschap nodig. In de meeste gevallen zijn er kant en klare IDE's beschikbaar. Voor de Arduino is de inmiddels welbekende groene IDE te downloaden van <https://www.arduino.cc/>. We zijn aangekomen bij versie 2.3.1. Beschreef ik in SoftwareBus 2024-1 nog hoe je de IDE met `install.sh` kon installeren, inmiddels is er ook een zogenaamde *Appimage* beschikbaar. Een Appimage is een gecompileerd en uitvoerbaar programma, compleet met alle bijbehorende bibliotheken. Download de Appimage van de Arduino-website en plaats die in een makkelijk terug te vinden directory (map). Ga naar die map en klik met de rechter muisknop op het bestand. Kies dan 'Eigenschappen' en klik op de tab 'Rechten'. Vink tenslotte het vakje aan bij 'Sta toe dat dit bestand wordt uitgevoerd...' En klik op de 'Sluiten'-knop. Je kunt nu het Arduino IDE starten. Toen ik dit stuk aan het schrijven was verscheen er plotseling een Nieuwe versie van de Arduino IDE, versie 2.3.1. Ik heb toen de Zip-versie gedownload. Kwestie van uitpakken en het uitvoerbare bestand starten. Wel zo makkelijk. Aan jou de keuze.

Je bent natuurlijk niet verplicht om de Arduino IDE te gebruiken. In SoftwareBus 2024-1 beschreef André Reinink hoe je met Visual Studio Code en PlatformIO aan de slag kunt gaan. Is natuurlijk ook een mogelijkheid. Ik ben daar niet mee bezig. Voor mij is het inderdaad een 'Chaotische ontwikkelomgeving'.

Project 1

Voor ons eerste project ga ik terug naar de potmeterschakeling die ik in SoftwareBus 2024-1 heb beschreven. Dus trek die even tussen de oude kranten en reclamefolders vandaan. Of download hem van website van CompUsers. En kijk even wat ik daar heb verteld. Voor alle duidelijkheid hier nog even een foto met de opstelling waar ik voor dit artikel nu mee werk.



Arduino Uno en broodplank met potentiometer

Het programma voor de Arduino, ook wel de sketch genoemd - wat ik een onzinnige naam vind en dus niet zal gebruiken - staat al in SoftwareBus 2024-1. En is bovendien te downloaden van mijn website. Jawel! Ik heb een speciaal hoekje ingericht voor deze serie stukken. Daar kunnen alle programma's worden 'gesponst' (gedownload) en is ook eventueel nieuws en aanvullende informatie te vinden. Kijk op <http://winlin.atwebpages.com/> en klik op het CompUsers-logo.

Dus zet alles klaar, start het potmeterprogramma op de Arduino en kijk of alles werkt. Dan gaan we in het volgende nummer verder met ons eerste Processing-programma.