

Draadloos monitoren en debuggen

voor Arduino, ESP32 en co.

Peter Tschulik (Oostenrijk)

Moderne microcontroller-boards, uitgerust met bijvoorbeeld een ESP32 of een ESP8266, bieden veel prestaties voor weinig geld, plus veel programmeergemak, vooral via de Arduino IDE. Bijzonder interessant is de over-the-air (OTA) functionaliteit. Hiermee kunnen zowel programma-updates als gegevens met behulp van WiFi gemakkelijk worden geladen via een bootloader, zonder de noodzaak om het board fysiek aan te sluiten op de USB-poort van een PC of laptop. Maar om seriële gegevens (zoals debug-informatie) draadloos te verzenden, hebben we een andere oplossing nodig.



De draadloze UART/RS232 lange afstand-set.

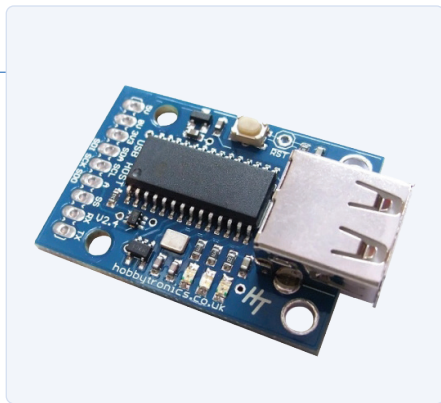
IoT-apparaten die ergens 'in het veld' werken, hebben baat bij de mogelijkheid om software-updates draadloos (over-the-air, OTA) uit te voeren. Een goede tutorial over dit onderwerp met behulp van de Arduino IDE is te vinden op [1]. Deze methode heeft echter één nadeel: de seriële monitor wordt niet ondersteund, zodat uitvoer- en debug-informatie niet kan worden overgedragen. Bij mijn laatste project, een tuinbewateringssysteem gebaseerd op de ESP32, was er een probleem met een slecht werkende sensor. In plaats van een USB-kabel helemaal over

het terras naar de woonkamer en naar mijn computer te leggen, ben ik op zoek gegaan naar een draadloze oplossing. Ondanks een intensieve zoektocht kon ik geen kant-en-klare module vinden die aan mijn eisen voldeed, dus begon ik mijn eigen oplossing te ontwikkelen. Ik meende me systemen te herinneren die draadloze transmissie van seriële interface-gegevens mogelijk maakten; in feite vond ik al snel wat ik zocht bij een bedrijf in het Verre Oosten. De 'Wireless UART-RS232 long distance set' [2] biedt overdrachtsnelheden tot 115.200 baud met configuratieopties en een

bereik tot 1 km. Met een prijs van £ 50 voor een zend/ontvang-paartje vond ik dit een goede prijs/prestatie-verhouding voor dit project.

Een USB host-adapter

De keuze van een USB/serieel-converter bleek veel moeilijker. Hoewel het mogelijk was om het seriële datasignaal van de externe bewateringssysteem rechtstreeks aan te sluiten op het seriële transmissiesysteem, wilde ik een meer universele oplossing met een USB-aansluiting, zodat ook andere typen development-boards



Figuur 1. Iets goeds hoeft niet groot te zijn: het USB host-board (bron: www.hobbytronics.co.uk).



Figuur 2. De draadloze UART/RS232-modules moeten worden geconfigureerd. Op de foto zijn de jumper en de groene en rode LED's te zien.

gemakkelijk gebruik zouden kunnen maken van de verbinding. Hiervoor is een zogenaamde USB-hostadapter nodig, die op de externe locatie in de USB-poort wordt gestoken en met het ontwikkelboard praat zoals een PC dat zou doen. Gelukkig vond ik na wat intensief speuren op internet het 'USB Host Shield' voor Arduino [3][4]. Het artikel was op voorraad bij een plaatselijke leverancier, dus ik had het binnen een paar dagen. Maar ik ontdekte al snel een probleem. Er zijn veel verschillende ontwikkelplatforms in de maker-wereld en er zijn verschillende merken USB-interfacechips. De ESP32-boards bijvoorbeeld gebruiken de CP210x- of FTDI-chips, Arduino-boards gebruiken de Atmel-controller of FTDI-chips. Helaas

kunnen de bibliotheken voor het Arduino USB Host Shield slechts één type chip tegelijk ondersteunen. Dat betekent dat de host shield-software configureerbaar moet zijn en dat u moet uitzoeken welk type chip geschikt is voor het ontwikkelplatform dat u gebruikt. Niet bepaald een plug&play-oplossing! Uiteindelijk vond ik een USB host-interfacekaart uit het Verenigd Koninkrijk die veel beter geschikt was voor verschillende merken interface-chips (**figuur 1**) [5].

Wat vooral interessant is aan dit board is dat het met een breed scala aan apparaten, zoals flash-geheugensticks, USB-toetsenborden, USB-joysticks, USB-muizen en PS3/PS4-dual-shock-controllers overweg kan, en seriële drivers heeft voor FTDI, CP210X, PL2303, CH340/1 en CDC, MIDI-apparaten en USB-modems. Een lijst van alle gratis firmware-applicaties staat op hun website. Als u precies weet voor welke toepassing u het board wilt gebruiken, kunt u dat bij de bestelling opgeven en dan wordt het board geleverd met geïnstalleerde firmware. Voor deze toepassing had ik de *USB Host - Serial Driver for FTDI, CP210X, PL2303, CH340/1 and CDC*-firmware [6] nodig; deze ondersteunt alle relevante USB-chips die door de verschillende development-boards worden gebruikt.

Voor het programmeren en configureren van de draadloze UART/RS232-modules of het USB host-board

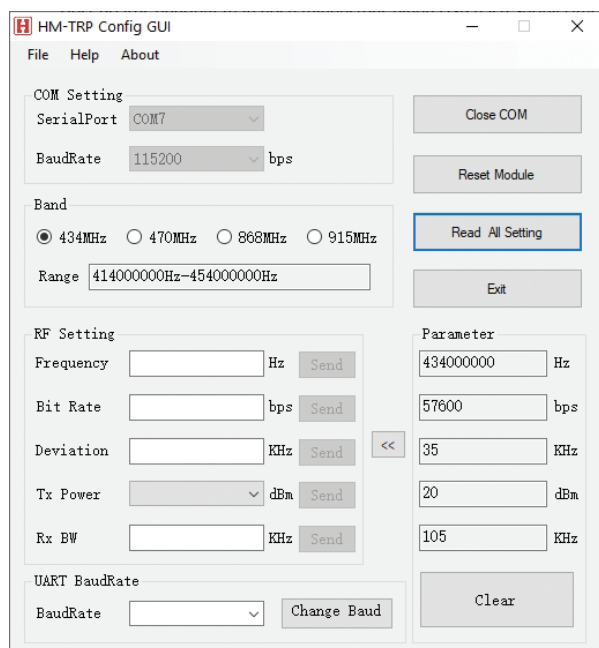
is tenminste één USB/serieel-converter nodig, die ofwel van dezelfde fabrikant van de draadloze UART RS232-modules [8] kan worden verkregen of bijvoorbeeld van Logilink [9]. Voor mijn project had ik twee RS232 niveau-omzetters nodig. Ik heb gekozen voor de goede oude MAX232-oplossing [7].

Het opzetten van de verbinding

Toen ik de draadloze UART/RS232-set ontving, was me niet duidelijk hoe ik de modules moest configureren. Ik nam contact op met de verkoper en ontving de volgende dag een e-mail met de software, een handleiding en een link naar een configuratievideo [10]. Eerst moest ik de modules uit hun plastic doosjes halen en een pinheader solderen op connectorpositie J2 (een rij pads net voor de RS232-connector). Daarna heb ik conform de instructies jumpers op de pinnen gezet, zodat de module in de configuratie-modus komt. Vervolgens heb ik de module op de PC aangesloten via de USB/serieel-adaptor. De module wordt gevoed uit een 5V-voeding (bijvoorbeeld een USB plug-in voeding) via de mini-USB-connector (**figuur 2**).

Als beide LED's oplichten, kunt u het programma *HY-TRP Setting GUI.exe* opstarten en de juiste COM-poort en de standaard-baudsnelheid op 9.600 instellen. Klik vervolgens op de knop *Open COM* en lees alle waarden uit met *Read All Settings*. Dat levert het resultaat van **figuur 3**.

Verdere details zijn te vinden in de handleiding [10]. Enkele problemen met de betrouwbaarheid van de verbinding werden opgelost door de baudrate voor de transmissie te verlagen van 115.200 naar 57.600. Het USB host-board heeft een buffer die verschillende overdrachts-snelheden tolereert. Beide modules moeten identiek worden geconfigureerd. Vergeet ten slotte niet om de jumpers te verwijderen zodra de configuratie voltooid is.

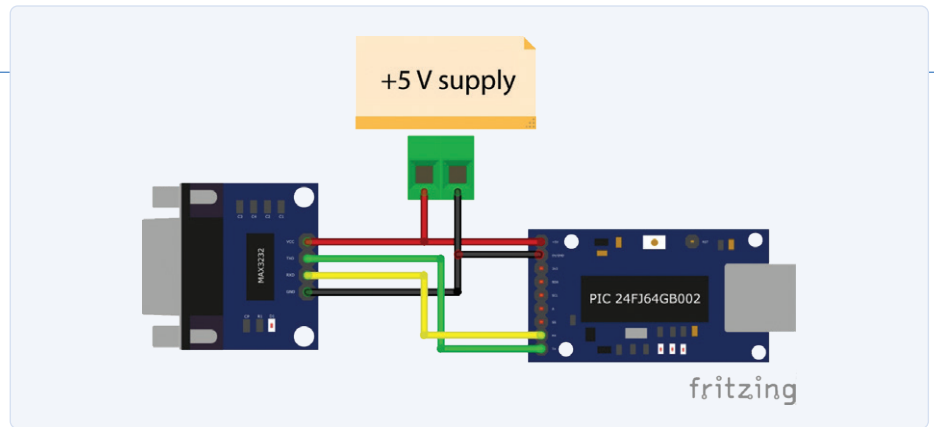


Figuur 3. De aanbevolen configuratie voor de UART/RS232-module.

Configuratie van het USB host-board

Zie voor het programmeren van het USB host-board met de juiste firmware de beschrijving in [5]. Bij Hobbytronics kunnen veel programma's worden gedownload die op het board kunnen worden geflashed. De juiste firmware kan op het board worden geladen met behulp van de instructies in [6], en daar vindt u ook een beschrijving van de parameters van de opdracht-interface. Voor de configuratie moet het USB host-board worden aangesloten op de MAX3232-converterprint, zoals afgebeeld in **figuur 4**. De RS232/TTL-converter wordt verbonden met de RS232/USB-adapter, die op zijn beurt via een USB-kabel wordt verbonden met de PC. De schakeling heeft nu alleen nog stroom nodig van een stabiele 5V-voeding.

Nu kunt u een terminalprogramma starten (hier bijvoorbeeld Hterm), de juiste USB-poort selecteren en de baudrate instellen op de standaardwaarde van 9.600 baud. Door op de knop **Connect** linksboven te klikken,



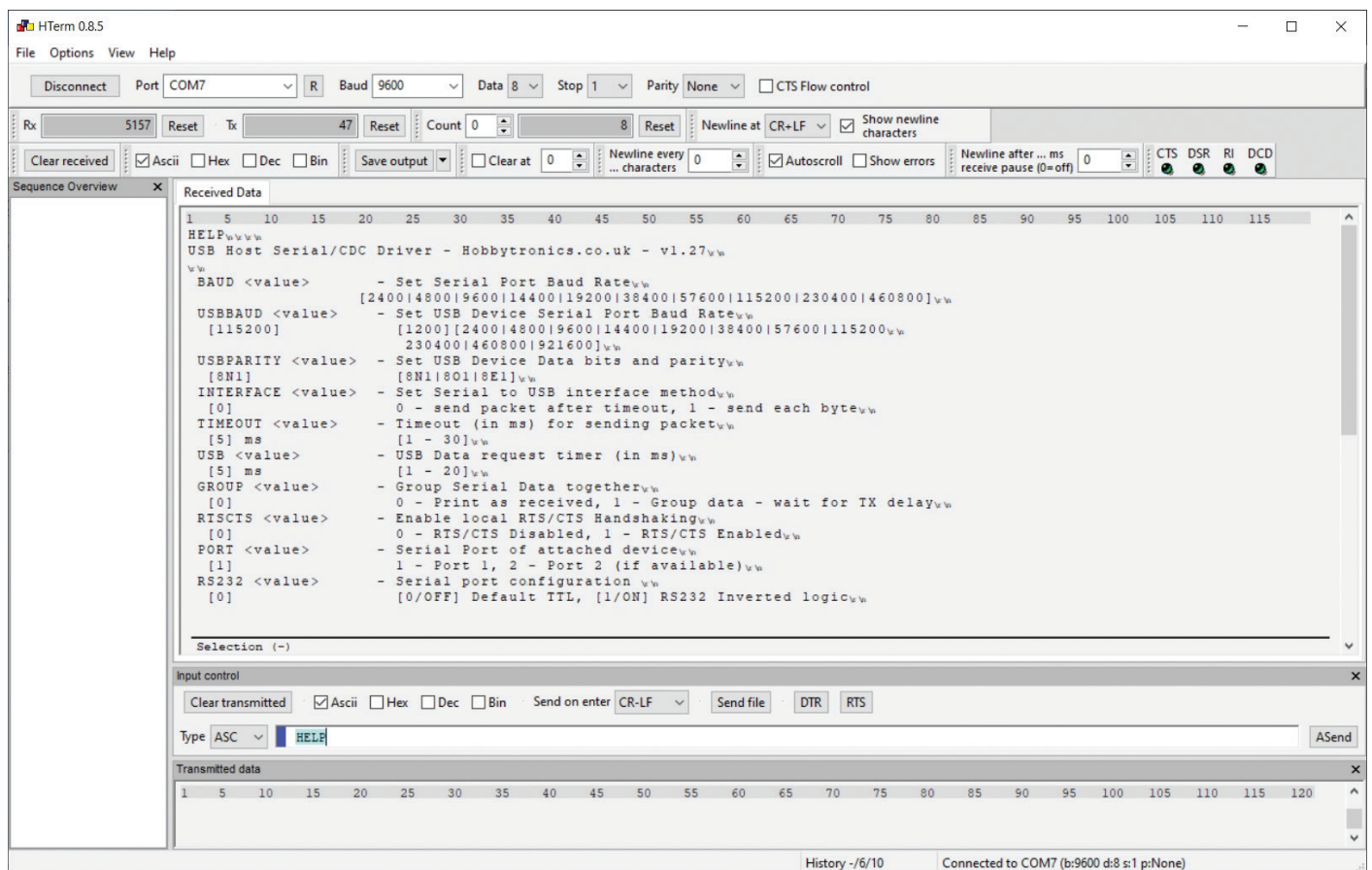
Figuur 4. Aansluiting van het USB host-board voor configuratie.

maakt u verbinding met het USB host-board. Rechtsboven moet **Newline at** en onder het hoofdvenster moet **Send on Enter** worden ingesteld op **CR + LF**. Als u nu **HELP** invoert in het **Input Control**-veld onder het hoofdvenster, worden de actuele parameters opgesomd zoals in **figuur 5**.

De parameters voor onze toepassing worden nu gewijzigd met behulp van het veld **Input Control**. Als eerste kunnen we het commando **BAUD 57600** invoeren om de baudrate te wijzigen, die overeen moet komen met de

baudrate die voor de draadloze systemen is ingesteld. Als u op **Disconnect** klikt en dan op **Connect** (om de verbinding te verbreken en weer te maken), zou u de communicatieverbinding met het USB host-board moeten kunnen controleren met **HELP**.

Het commando **USBBAUD 115200** stelt de USB-baudrate in op 115.200. Dit betekent dat de seriële interface van het aangesloten Arduino/ESP32/ESP8266-board in de Arduino IDE altijd moet beginnen met het commando **Serial.begin (115200)**.



Figuur 5. Configuratie van het USB host-board met behulp van Hterm.

Tabel 1. Parameter-instellingen in HiTerm.

BAUD	57600
USBBAUD	115200
USBPARITY	8N1
INTERFACE	0
TIMEOUT	5ms
USB	5ms
GROUP	0
RTSCTS	0
PORT	1
RS232	0

Voor *USBPARITY*, *INTERFACE*, *TIMEOUT*, *USB*, *RTSCTS*, *PORT* en *RS232* worden de standaardinstellingen gebruikt. De standaardwaarde van de optie voor het groeperen van seriële gegevens is *OFF*. Het commando *GROUP 0* stelt ook deze eigenschappen in op hun standaardwaarde. **Tabel 1** geeft een overzicht van de juiste instellingen. Hiermee is de configuratie voltooid.

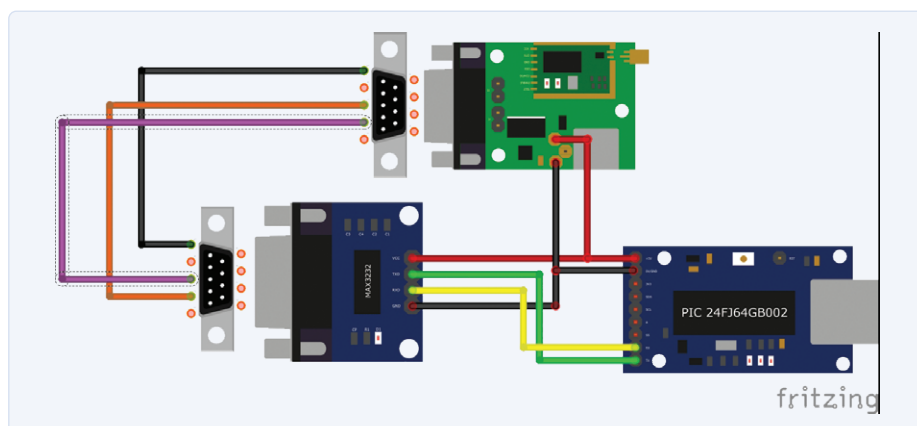
Bedraden van de modules

Voordat de zender (bij het IoT-board in het veld) en ontvanger (bij de PC) bedraad kunnen worden, moet de op de print gemonteerde voedingsaansluiting (naast de USB-aansluiting) van de draadloze UART/RS232 long distance-modules worden verwijderd, zodat de aansluitpinnen bereikbaar zijn.

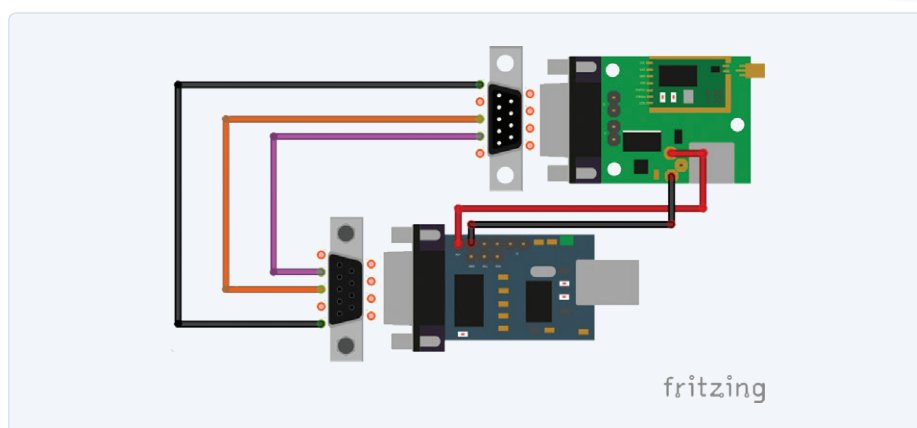
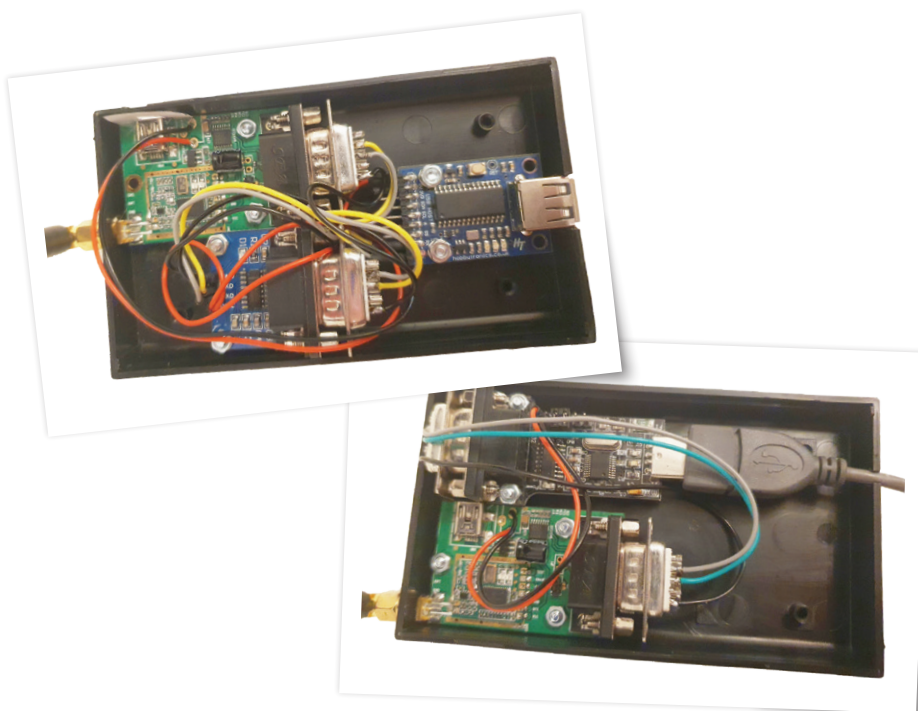
Aan de zenderzijde wordt het apparaat gevoed via de mini-USB-aansluiting van het USB host-board. **Figuur 6** toont de bedrading aan de zenderzijde. Om de draadloze UART/RS232-module op de RS232/TTL-adapter aan te sluiten, moet u een korte kabel gebruiken met male SubD9 connectoren. De draden naar pin 2 en pin 3 zijn gekruist in de kabel, zoals te zien is in de figuur.

De compacte module uit [8] wordt gebruikt als RS232/USB-converter voor de ontvanger en wordt bedraad conform **figuur 7**. Ik heb een USB-verlengkabel gebruikt tussen de PC en de ontvanger om enige speelruimte te hebben bij de plaatsing van de ontvanger binnenshuis.

Tenslotte installeerde ik de zender en de ontvanger in twee bij elkaar passende



Figuur 6. Bedrading van de draadloze zender (bij het IoT-board). De groene print is de draadloze module. De IoT-print die gegevens levert, wordt aangesloten op de USB-poort rechtsonder.



Figuur 7. Bedrading van het ontvangerstation. De PC wordt rechtsonder aangesloten via een USB-verlengkabel.

behuizingen. De tests van mijn OTA-systeem toonden consistente transmissie door twee verdiepingen met plafonds van gewapend beton. Al met al biedt het systeem een handige en betrouwbare 'virtuele USB kabel' voor de langere afstand die nuttig is bij het ontwikkelen en debuggen van remote IoT-toepassingen. ◀

200549-03

Vragen of opmerkingen?

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

Een bijdrage van

Idee, ontwerp en tekst: Peter Tschulik
Redactie: Rolf Gerstendorf
Vertaling: Jelle Aarnoudse
Layout: Giel Dols

Praktijktips

- › Configureer de seriële communicatie-interface van het gebruikte board op 115.200 baud met `Serial.begin(115200);` in de Arduino IDE.
- › Zodra de firmware naar het remote board is overgebracht, voorziet u de zenderunit van 5 V via de mini-USB-aansluiting van de draadloze UART/RS232-module. Sluit het vervolgens aan op het remote board met behulp van een korte USB-kabel.
- › Sluit de ontvanger aan op een vrije USB-poort van de PC. De ontvanger wordt gevoed vanuit de PC.
- › Start de Arduino-IDE, selecteer een willekeurig board, selecteer de USB-interface van de receiver en start de seriële monitor.
- › Alle gegevens die door het board worden verzonden, moeten nu te zien zijn.



GERELATEERDE PRODUCTEN

- › **Bundel: The Complete ESP32 Projects Guide + ESP32-DevKitC-32D (SKU 19897)**
www.elektor.nl/19897



- › **ESP32 & ESP8266 Compilation (PDF) (SKU 18516)**
www.elektor.nl/18516

WEBLINKS

- [1] ESP32 Basic Over The Air (OTA) Programming In Arduino IDE: <https://lastminuteengineers.com/esp32-ota-updates-arduino-ide/>
- [2] Draadloze UART/RS232 lange-afstandmodule: <http://www.aliexpress.com/i/32242647632.html>
- [3] SparkFun USB Host Shield: <http://www.sparkfun.com/products/9947>
- [4] USB Host Shield voor Arduino: <https://bit.ly/3fJQrTO>
- [5] USB Host Controller Board V2.4: <https://www.hobbytronics.co.uk/usb-host-board-v24>
- [6] USB Host - seriële driver voor FTDI, CP210X, PL2303, CH340/1 en CDC: <https://www.hobbytronics.co.uk/usb-host-serial>
- [7] RS232/TTL-adaptor: <https://bit.ly/3peJH3r>
- [8] Bifunctionele USB COM Port DB9 TTL232 RS232 TTL 232 CH340T USB2.0 kabel-adaptor: <https://bit.ly/3fHUGQ3>
- [9] LogiLink USB-2.0/serieel-adaptor: <https://bit.ly/3ma21Kg>
- [10] Configuratie-instructies voor de draadloze UART/RS232 lange afstand-module HY035_HM-TRP-RS232.zip: <https://1drv.ms/u/s!AjrAGEbCBLsugxiqmd1FRRCJjic>