NTP Ham Clock

Bouw een NTP server gestuurde Ham Clock
met slechts enkele onderdelen

Cor Struyk, PA0GTB
PA0GTB@veron.nl

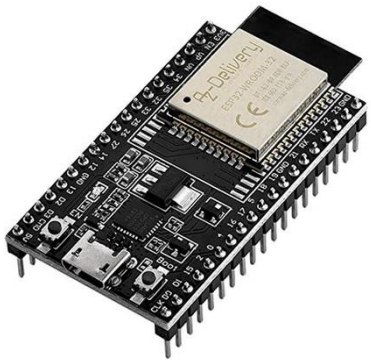
Introductie
Het is moeilijk om in deze Corona tijd stil te zitten en niets om handen te hebben. Naast mijn stoeipartijen met een Raspberry Pi, bekroop me toch weer het gevoel dat ik op zoek moest gaan naar een handzaam Arduino Ham Radio georiënteerd project.

Welnu, ik heb dat gevonden op de website van Bruce, W8BH, die daar een aantal leuke schakelingen had staan. Ik heb uiteindelijke gekozen voor een NTP Server gestuurde Ham Clock welke je op 1 display de lokale tijd en de UTP tijd laat zien.
Er wordt gebruik gemaakt van een ESP32 microcontroller module en een 3.2 Inch 320 x 240 pixels TFT LCD display welke gestuurd wordt middels de TFT_eSPI library, welke ook nog kennen uit het eerder gebouwde Arduino Weerstation project.

Wat is NTP ?
NTP staat voor Network Time Protocol, en het is een Internet protocol wat gebruikt wordt om computers, clocks en dergelijke te synchroniseren op tijd.
Door gebruik te maken van een NTP Server, wordt deze clock elk uur met enkele tienden van een milliseconde gesynchroniseerd met de UTC .
UTC (in het Nederlands ook aangeduid als gecoördineerde wereldtijd) is sinds 1972 een standaardtijd, gebaseerd op een atoomklok en gecoördineerd met de rotatie van de aarde.
UTC is ook de standaardtijd waarmee we als Radio Amateurs werken.

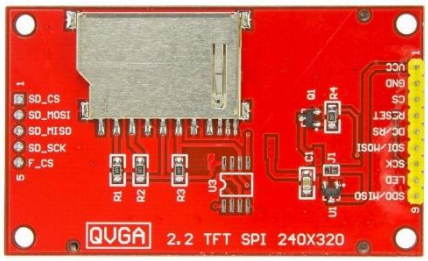
ESP32
Er wordt in dit concept gebruik gemaakt van een standaard verkrijgbare ESP32 microcontroller module met de mogelijkheid om via Wifi met het Internet te verbinden

Er zijn verschillende uitvoeringen van een ESP32 Microcontroller toepasbaar. Belangrijkste is de Wifi mogelijkheid
De ESP32 is een low-cost, low-power system in een chip, met geïntegreerde WiFi en dual-mode Bluetooth module.
Er is verder voorzien in een onboard antenne, switches, RF balun, power amplifier, low-noise receive amplifier, filters, en power-management modules.
De ESP32 is een opvolger van de eerdere ESP8266 microcontroller.



Het Display
Ik heb gebruik gemaakt van een standaard 320 x 240 pixels TFT LCD Display, wat gebruik maakt van een ILI9341 driver en een SPI interface.
Het is een 3.3 V device en is verkrijgbaar is diverse maten van 2.2 inch tot 3.2 inch.
Het display heeft 9 aansluitpunten voor de LCD en nog een 5 tal voor de aanwezige SD kaart houder, welke we overigens niet gebruiken.

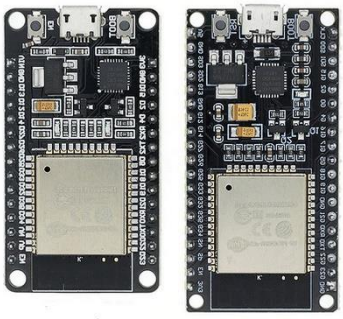
Deze TFT Displays vindt je gemakkelijk op diverse websites maar vooral bij Aliexpress, Ebay en dergelijke.
Je vindt ze ook met een “Touch screen” optie, maar dat is in dit project ook niet aan de orde. Bij meerdere aansluitingen is het belangrijk om alleen de aangegeven aansluitpunten te gebruiken. En let er goed op dat het displaygebruik maakt van een ILI9341 driver, want daar is de sketch ook op gebaseerd.



Test Sketch
Het is verstandig om straks eerst met behulp van een test sketch de werking van het display in combinatie met de ESP32 microcontroller te testen voordat we met de grote sketch beginnen. Hiermee voorkom je een heleboel problemen.

TFT_eSPI library
Gebruik ook de juiste TFT_eSPI library van Bodmer, verse 2.2.14
Installeer deze op de gebruikelijke wijze in de Arduino map Libraries. Als je die niet hebt, kun je hem downloaden op https://github.com/Bodmer?TFT_eSPI

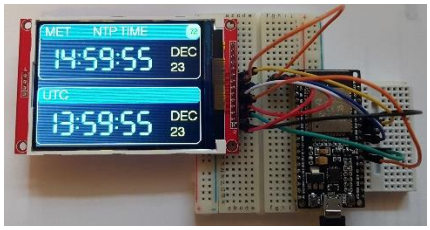
Hardware Informatie ESP32
Als we de ESP32 module wat nader bekijken, valt gelijk op dat hij een stuk groter is dan de oudere ESP8266. Een groot aantal pinnen zijn de I/O pinnen (GPIO pins). Afhankelijk van het type ESP32 wat je gebruikt, kan het aantal I/O pinnen variëren. Je komt de ESP32 module tegen met 30, 32, 36 en 38 pinnen.



Het overgrote deel van de module wordt in beslag genomen door de ESP32-WROOM-32 microprocessor. Het is een 32-bit microprocessor En heeft een aantal knopjes en Led’s. De microprocessor zelf is afgedekt met een koellichaam.

Belangrijk zijn ook de **EN** of **RST** en het **BOOT / Enable** knopje, die gebruikt worden tijdens het programmeren van de ESP32 middels de Arduino IDE. Hierover later meer.

Opbouw van de schakeling
Nou de opbouw is natuurlijk ook gemakkelijk met slechts 2 onderdelen.
Ik heb voor de test gewoon alles op een paar stukjes breadboard samengebracht.



Programmeren met de Arduino IDE
Je gebruikt de Arduino IDE om de sketch aan te passen en de ESP32 te programmeren.
Sluit de ESP32 aan via de USB poort

Stel in de Arduino IDE de ESP32 module in om te kunnen gebruiken onder boardbeheer. Als de ESP32 nog niet aanwezig is, voer dan onderstaande handelingen uit om deze te installeren:

- Klik op Bestand / voorkeuren en voer op het **tabblad Instellingen** deze URL in bij **Meer Board Managers** URLs : https://dl.espressif.com/dl/package_esp32_index.json
- Voeg de module toe aan de IDE via **Hulpmiddelen / Board / Boardbeheer**
- Zoek op “ESP32” en klik bij **esp32 by Community** op installeren (dit kan even duren !!)
- Selecteer nu het juiste board : **hulpmiddelen / Board / ESP32 Arduino (ESP32 Dev Module)**
- **Zet de Upload speed op 115200**

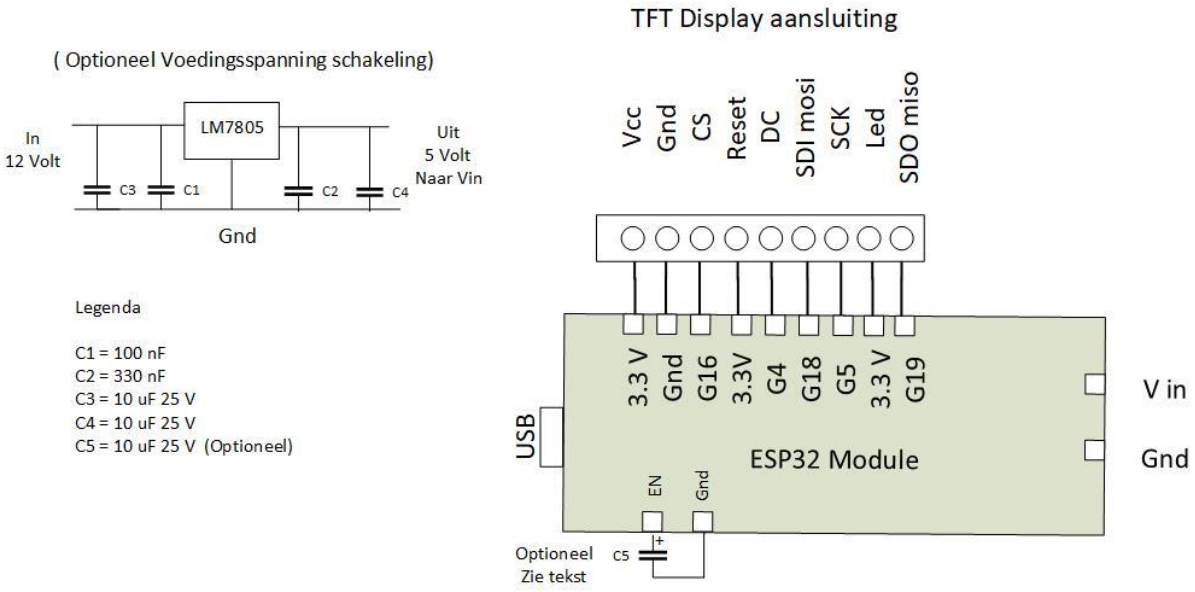
Selecteer tenslotte de juiste Com-poort via **Hulpmiddelen / Poort**

De bijgeleverde sketch (zie VERON Bijlage voor download) dient in principe alleen aangepast te worden voor het gebruik van je eigen Wifi omgeving. Je dient dus deze zaken in verschillende regels van de sketch aan te passen.

Lokaal Wifi access
Stel op deze regels de informatie van jouw Wifi station in :

35 #define WIFI_SSID "< de SSID naam >";
36 #define WIFI_PWD "< het wachtwoord >";

Het Schema



Er wordt in principe maar gebruik gemaakt van 2 componenten
De ESP32 module en het TFT Display
Direct aansluiten via de USB connector is dus mogelijk.

Ik wilde zelf graag een nog beschikbare 12 Volt voedingsadapter gebruiken en heb een extra schakelingetje toegevoegd met een LM7805 spanningsregelaar en die na programmering van de sketch aangesloten op de Vin van de ESP32 module. De ESP32 heeft ook een spanningsregelaar on-board. die de spanning op Vin of vanaf USB naar 3.3V verlaagd. Dit is ook de spanning voor het display. Je kunt natuurlijk ook gewoon 5V aansluiten op de pin Vin van de ESP32.

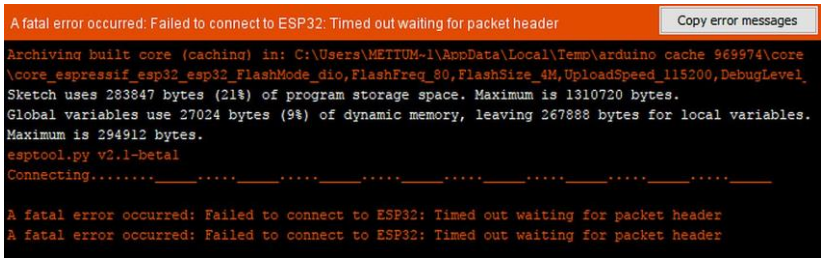
Display test
Om het display te testen, om te weten of alles goed werkt, kan je eerst de Test sketch gebruiken uit de VERON download bijlage.
Zorg eerst dat je in de Arduino IDE de ESP32 onder boardbeheer hebt toegevoegd !
Sluit daarna de ESP32 module via de USB connector aan op een USB poort van je Computer.

Controleer welke COM-poort er wordt toegevoegd en stel deze in op de Arduino IDE.

Het programmeren
Het is aan te bevelen om eerst via de Verifieer mode van de Arduino IDE te controleren of er in de sketch geen fouten zitten. Daarna kun je met de Upload beginnen.

Als de upload klaar is, moet je een paar keer op het **Enable** of **RST** knopje van de ESP32 drukken. Daarna zal de sketch gestart worden. Je kunt ook de voedingsspanning even uit- en inschakelen. Beetje vreemd, maar zo werkt het bij de ESP32.
De groene Led (rechtsboven) geeft aan dat alles gesynchroniseerd is. Bij geen synchronisatie wordt deze Led Oranje of Rood (24 uur geen Sync). Het getal is je positieve Wifi veldsterkte notatie.

Probleemoplossing bij uploaden sketch naar ESP32
Sommige (met name Chinese) uitvoeringen van de ESP32 modules geven zijdens de upload een foutmelding, dat de communicatie niet goed verlopen is.



<u>Er zijn een 2 tal manieren om dit probleem op te lossen :</u>

Zorg dat je voordat de echte Upload begint, het **BOOT / Flash** knopje op de ESP32 hebt ingedrukt, en houd dit ook vast tijdens de upload van de sketch totdat gemeld wordt dat de upload 100% is. Of, Breng een Elektrolytische condensator van 10 uF aan tussen de punten **EN** en **Gnd** van de ESP32 dan ben je er ook vanaf.

Behuizing
Afhankelijk van het formaat van het gekozen display, kun je een behuizing kiezen. Ik heb al wel de spullen klaarliggen, maar het werkplaatsje waar ik normaliter mijn werkzaamheden kan verrichten, is vanwege de Corona perikelen nu ook niet toegankelijk. Maar dat komt nog wel.

Resume
Met behulp van een ESP32 module en een display, kun je dus een handige dubbele klok maken voor in de shack. Wil je meer weten over dit type projecten, kijk dan nog maar eens op de website van Bruce, W8BH, die een passie heeft in het ontwerpen van klokken. www.w8bh.net
Ik heb van Bruce ook toestemming gekregen om dit Dual-Clock project voor jullie te mogen beschrijven en om te zetten naar NL toepassing.

Bijlage voor download
Het ZIP-bestand is weer te vinden op www.veron.nl/electronlinks
Daarin vind je:

- De test_sketch voor het TFT Display
- De laatste versie sketch voor deze Dual Ham Clock
- Library informatie

Veel plezier met het bouwen. 73 de Cor, PA0GTB