

NeoPixel klok

Wat is dit en hoe werkt het?



Versie 1.0, April 2020

Deze handleiding werd ontwikkeld door **Hans Caluwaerts** voor Maakbib (STEM-partnerschap VLAIO)



AGENTSCHAP
INNOVEREN &
ONDERNEMEN



Vlaanderen
is ondernemen



en valt onder de Creative Commons licentie



www.maakbib.be

www.decreatievestem.be

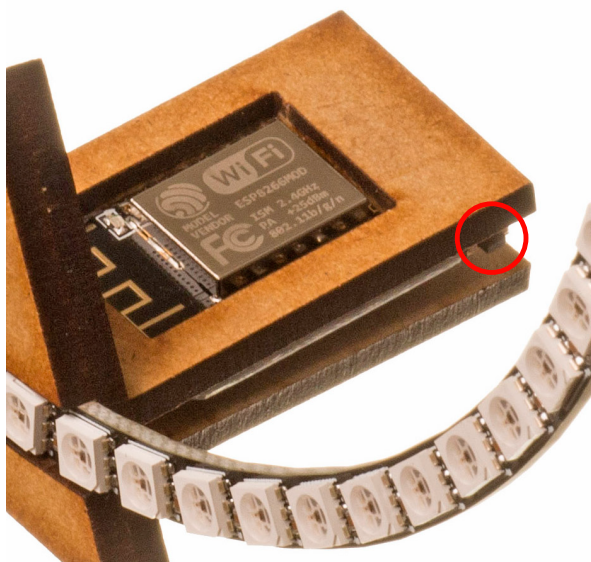
www.vlaio.be/nl

www.stem-academie.be

WiFi LED klok - Gebruik

Er staan geen knoppen op de Neopixel klok, er is geen webinterface en dus er is ook geen bediening mogelijk.

Bijna juist want er is een kleine - een beetje verdoken - miniatuur RESET knop die je waarschijnlijk nooit zal gebruiken. Druk je op de RESET knop dan start de Neopixel klok helemaal opnieuw van bij het begin en gaat de huidig getoonde tijd verloren.



Als je de Neopixel klok wil gebruiken, verbind je de Neopixel klok via een USB kabel (USB-A mannelijk naar micro USB) met een USB voeding (telefoonlader, USB aansluiting op je computer, ...).



Zorg ervoor dat de Neopixel klok zich binnen het bereik van een WiFi netwerk bevindt.

Bijna klaar, er is nog wat programmeerwerk!

Neopixel klok - Programmeren

De klok moet ergens de juiste tijd “vinden”, anders kan de klok niet werken.

Via het internet kan de Neopixel klok de tijd opvragen maar dan is er natuurlijk een internetverbinding nodig.

WiFi kan daarvoor zorgen maar je zal zelf wat moeten meehelpen.

In het programma (of sketch zoals dat bij Arduino genoemd wordt) staat de naam van het WiFi netwerk waarmee de Neopixel klok moet verbinden om de tijd op te vragen. Hoogstwaarschijnlijk zal je ook nog een wachtwoord nodig hebben om verbinding te kunnen maken met het WiFi netwerk.

Beide gegevens zal je zelf moeten ingeven in de sketch, want dat zal voor iedereen anders zijn.

Nadat je de WiFi gegevens hebt ingegeven zal je het aangepaste programma in de microcontroller moeten laden, pas dan kan het nieuwe programma uitgevoerd worden.

De “gemakkelijkste” weg om de sketch aan te passen is door gebruik te maken van de Arduino IDE, misschien staat die al op je computer, je kan die gratis downloaden.

<https://www.arduino.cc/en/Main/Software>

De microcontroller in de Neopixel klok is een Wemos D1, die is gebaseerd op de ESP8266 chip.

De Wemos D1 is geen echte Arduino en wordt na de standaardinstallatie van de Arduino IDE niet herkend.

Je kan via de board manager zelf ESP8266 gebaseerde boards toevoegen, hoe dat precies gebeurt staat beschreven in het document “ESP8266 installatie in de Arduino IDE”.

Nadat je ESP8266 gebaseerde boards toegevoegd hebt aan de Arduino IDE kan je de sketch openen en de WiFi gegevens aanpassen, de sketch bewaren, compileren en uploaden naar de WiFi LED klok.

Je eigen Wifi gegevens moet je ingeven in de lijnen 49 en 50, de aanhalingstekens moeten behouden blijven.

```
49 const char ssid [] = "naam van je wifi netwerk";  
50 const char wachtwoord[] = "wachtwoord van je wifi netwerk";
```

Download en importeer de TimeLib bibliotheek. <https://github.com/PaulStoffregen/Time>

Importeer de Adafruit NeoPixel bibliotheek via Arduino bibliotheekbeheer.

Selecteer “Wemos D1 R2 & mini” bij boards en selecteer de poort waarmee je Neopixel klok verbonden is. Klik op de uploaden knop en wacht tot de sketch geüpload is.

Als alles goed verloopt zie je de tijd verschijnen op de Neopixel klok.

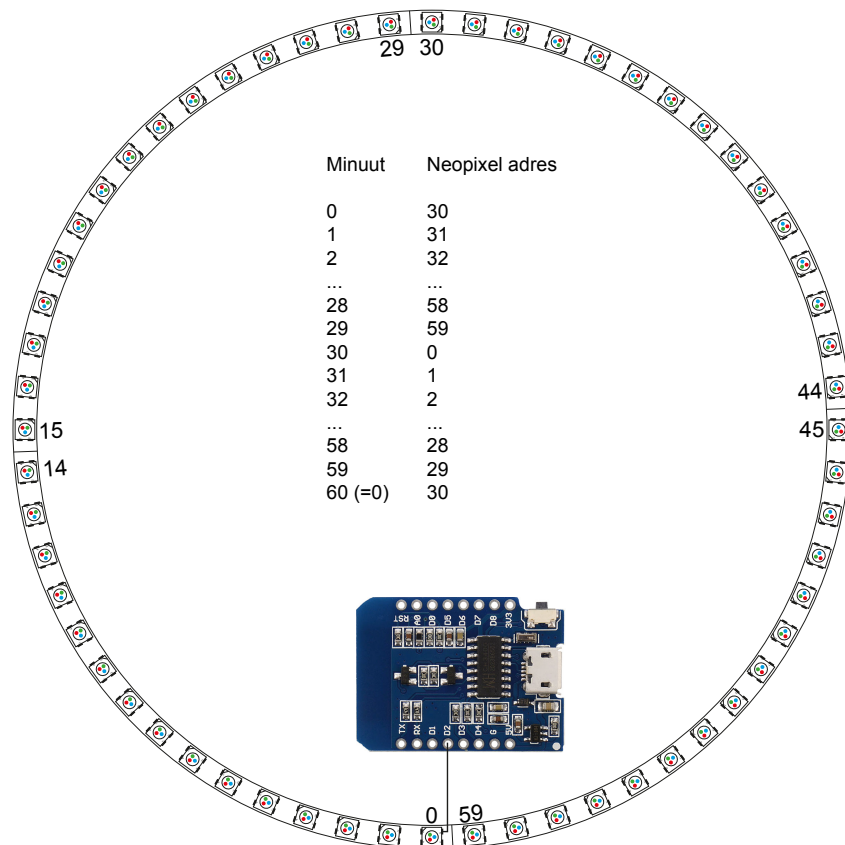
Neopixel klok – Hoe werkt het

Zolang de Neopixel klok probeert om verbinding te maken met het ingestelde WiFi netwerk zal er 1 Neopixel onderaan de cirkel rood knipperen.

Eens de Wifiverbinding tot stand is gekomen zal de Neopixel klok iedere 10 minuten de tijd vragen aan een tijd server.

De ontvangen tijd wordt aangepast aan de lokale tijdzone en er wordt rekening gehouden met zomer-/winteruur.

De Wemos D1 doet de nodige berekeningen om de overeenkomstige Neopixels te laten oplichten in de correcte kleur. Blauw duidt de minuten aan, groen het uur en rood de seconden. Het doen oplichten van de Neopixels is iets ingewikkelder omdat de eerste Neopixel onderaan de schaal zit; het was gemakkelijker geweest als de 1^e Neopixel zou samenvallen met de 1^e minuut. Onderstaande figuur toont het adres van de Neopixels, dat valt niet samen met de minuten van de klok.



Iedere seconde wordt de tijd op de Neopixel klok aangepast.

Een LDR meet het omgevingslicht, de intensiteit van de Neopixels wordt automatisch aangepast aan het omgevingslicht.

Neopixel klok – Zelf parametren

Kan ik de instellingen van de Neopixel klok zelf aanpassen?

Absoluut, maar wees voorzichtig, maak eerst een back-up van een werkende sketch. Als je geen of weinig programmeerervaring hebt kan je het beter niet doen.

In de sketch staat bijna bij elke lijn commentaar; commentaar verklaart wat er op die lijn gebeurt. Lees eerst de commentaar voor je begint te experimenteren. Hou steeds rekening met eventuele waarschuwingen vermeld bij het commentaar.

Kan ik iets kapot maken door de sketch aan te passen?

Absoluut, als je alle Neopixels op maximum intensiteit laat branden dan verbruikt de Neopixel klok te veel stroom, het risico is reëel dat je de USB kabel en de USB voeding overbelast. Om naar de originele toestand terug te keren zal je de Neopixels moeten loskoppelen van de Wemos D1, dit wil eigenlijk zeggen dat je de Neopixel klok moet demonteren.

Lijn 109 stelt de maximale helderheid in na RESET. 50 is de maximum veilige waarde.

```
109 | int helderheid = 50;
```

in de TAB LED staat

```
55 | helderheid = map(LDR_waarde, 0, 1023, 1, 50);
```

Hier wordt de maximale helderheid ingesteld volgens het omgevingslicht, 50 is de maximum veilige waarde.

Waarom is die sketch zo ingewikkeld?

De sketch gebruikt enkel een bibliotheek om UDP pakketten op te sturen en te ontvangen; het decoderen van het ontvangen UDP pakket met de tijdsinformatie werkt zonder bibliotheek en is vrij omvangrijk. Als je wat meer programmeerervaring opdoet zal je zien dat deze sketch eigenlijk nog “eenvoudig” is.