

Deze woordklok toont de tijd als tekst dat oplicht in een matrix van letters.

De tekst kan als witte tekst maar ook in kleur getoond worden. Voor de verlichting zijn SK6812 RGBW LEDs gebruikt.

De klok is geprogrammeerd met een Arduino IDE programma versie 1.8.13 of nieuwer voor een Arduino Nano Every chip en met als softwareversie: Character\_Colour\_Clock\_V061.  
De klok bevat een DS3231-klokmodule waarin een CR2032 batterij de spanning verzorgt als de klok uit staat. Deze klokmodule verloopt slechts enkele seconden per jaar maar voorziet niet in de zomer-wintertijdwissel. Als de klok zijn tijd niet meer vasthoudt na een stroomonderbreking dient de batterij vervangen te worden. Om de tijd precies gelijk te houden kan een DCF77-tijdsignaalonvanger worden ingebouwd.

Deze klok kan zijn uitgerust met een HM-10 BLE Bluetooth-module. Deze kan verbonden worden met een IPhone of een Android telefoon.

Voor de IPhone kan bijvoorbeeld het programma “BLE terminal” gebruikt worden.

Voor de Android kan “Serial Bluetooth van Kai Morich” gebruikt worden.

Om instellingen en tijd aan te passen kan de klok met een draaidrukknop bediend worden.

Behalve met de draaiknop kunnen instellingen met een Bluetooth-verbinding tussen de klok en een telefoon, tablet of PC uitgerust met Bluetooth-BLE ingesteld worden.

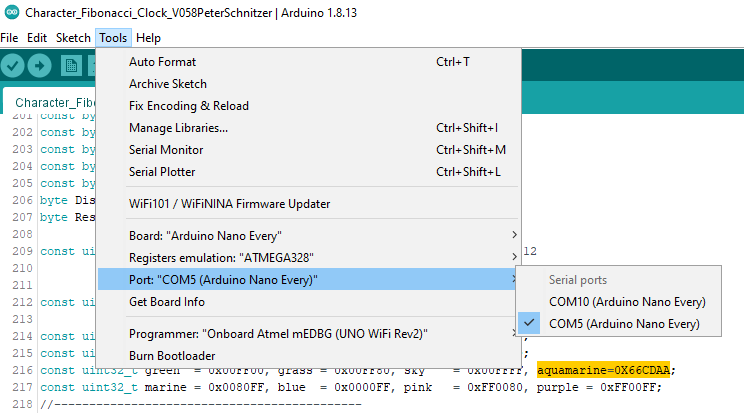
De klok onthoudt de ingestelde waarden van de opties L, M, P en Q in het menu als de stroom van de klok wordt gehaald. Dit is de minimale en maximale lichtsterkte, eigen kleur en displaykeuze.  
Met reset (R) in het menu kan de klok weer worden hersteld naar de oorspronkelijke instellingen

Als derde optie om instellingen aan te passen is een USB-PC-verbinding mogelijk. Met de seriële monitor in de Arduino IDE kan de klok bedient worden. Hiervoor dient een USB-mini kabel aan de PC te worden gekoppeld.

De Arduino IDE kan van de Arduino.cc website gedownload

In het programma Arduino IDE kan bij Tools >Port de Arduino Every gevonden worden. Kies een COM-poort waar Nano Every achter staat

Kies Tools->Serial monitor om de monitor te openen.



Met een Bluetooth-verbinding op de telefoon of PC kan hetzelfde gedaan worden.

Als verbinding met de PC wordt gelegd wordt het volgende getoond.

\*\*\*\*\*\*\*\*\*

Serial started

RTC DS3231 enabled

Bluetooth enabled

Rotary enabled

LEDs SK6812 enabled

DCF77 enabled

DCFtiny enabled

Compiled for ARDUINO AVR NANO EVERY

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Woordklok Sep 2020

Enter time as: hhmm (1321)or hhmmss (132145)

A Debug DCF-signal

D D15122017 is date 15 December 2017

G DCF-signalinfo in display

H Use DCF-receiver

Lnn (L5) Min light intensity ( 1-255 bits)

Mnn (M90)Max light intensity (1%-250%)

Pnnnnnn (P234F8A) own colour (n=0-F)

Qn Display Choice (Q0-6)

Q0= Gele kleur

Q1= Per uur een kleur

Q2= Alles wit

Q3= Eigen kleur

Q4= Aquamarijn

Q5= Wheel kleur

Q6= Aquamarijn, 5 min rood

Q7= eigen kleur, 5 min rood

I For this info

R Reset to default settings

Xnn (X50) Demo mode. ms delay (0-9999)

Z for Self test

Ed Nieuwenhuys 27 aug 2020

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Brightness Min: 5 bits Max: 40%

LDR read Min: 999 bits Max: 1 bits

Display choice: 7

Software: Character\_Colour\_Clock\_V061

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Als de seriële monitor wordt gebruikt wordt de kloksoftware gereset als je verbinding maakt met de klok. Je ziet dan welke componenten geïnstalleerd zijn.

Met een Bluetooth-verbinding gebeurt de herstart van de klok niet.

Elke 30 seconden wordt een regel geprint

Het is tien over half vier 15:41:49 18-09-2020

LDR:531 (516-531)->100=39% T=21C 15:42:00

Het was tien over half vier 15:42:00 18-09-2020

LDR:532 (516-532)->101=39% T=21C 15:42:30

LDR:532 (516-532)->101=39% T=21C 15:43:00

Het was tien over half vier 15:43:00 18-09-2020

LDR:532 (516-532)->101=39% T=21C 15:43:30 Eerst de lichtsterkte van de lichtsensor; LDR: ( 0 -1024) in bits en daarna omgerekend naar het percentage waarmee de LEDs branden ten opzicht van vol vermogen.  
Daarna de temperatuur in de klok.   
Daarna hoe goed de DCF-ontvanger het laatste uren gewerkt heeft (0-99%)

**Draaiknop**

Druk op de knop om het keuzemenu te activeren.

Na een druk op de knop gaat de tekst UUR knipperen. Door de draaiknop de draaien kunnen de uren veranderd worden

Een tweede druk op de knop laat de minuten knipperen. Nu kunnen de minuten gelijk gezet worden.

Een derde druk doet veel tekst oplichten. Nu kan de intensiteit van de klok aangepast worden door aan de knop te draaien.

De vierde en volgende drukken doorloopt de diverse kleurschema.

4 Gele tekst.

5 Elk uur een andere kleur van de tekst.

6 Alle tekst in wit.

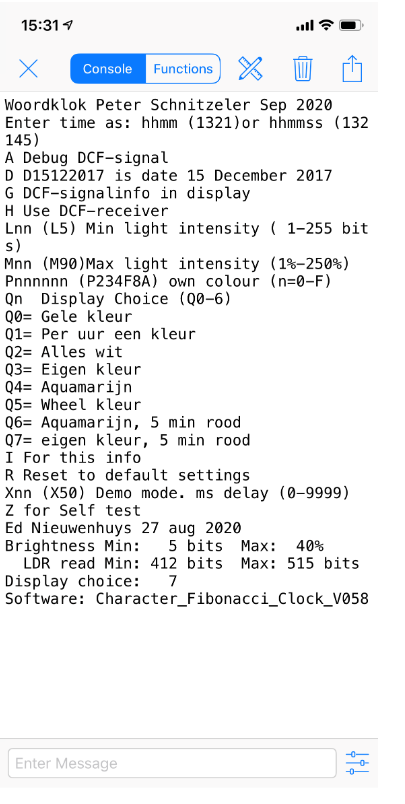
7 Alle tekst kan een eigen samengestelde kleur aannemen.

Deze kan worden gemaakt door in het menu van de klok nadat een verbinding met Bluetooth of de PC is gelegd Prrggbb te tikken. De rr, gg en bb zijn de primaire kleuren. Deze kan de waarde 00 – FF hebben. Dit is een hexadecimale waarde

Dus PFF0000 is knalrood en P0000FF knalblauw. Geel is PFFFF00. Rood en groen geeft geel.

8 Aquamarijn

9 Wheel kleur. Elke seconde verloopt de kleur door de regenboog

10 Aquamarijn, 5 min rood

11 Eigen kleur, 5 min rood

**Bluetooth of seriële verbinding**

Voer i in om het menu in beeld te krijgen

De instellingen kunnen aangepast worden met een letter en vaak een waarde.

Tik dit in en druk op return of verzenden.

In het menu kan Q0 tot en met Q7 ingevoerd worden voor de diverse kleurpaletten.

Q0= Gele kleur

Q1= Per uur een kleur

Q2= Alles wit

Q3= Eigen kleur

Q4= Aquamarijn

Q5= Wheel kleur

Q6= Aquamarijn, 5 min rood

Q7= eigen kleur, 5 min rood

**Menu-opties**

Ttime as: hhmmss (132145)

Voer de tijd in als hhmmss.

Bijvoorbeeld 051205 om 5 uur, 12 minuten en 5 seconden in de ochtend in te voeren

A Debug DCF-signal

Met deze optie wordt de DCF77-ontvangst getoond.   
Druk weer A om te stoppen

@... 8336 37563 22% [1] 00:15:25 00-00-2000/0 F2

@... 8131 37018 21% [1] 00:35:26 00-00-2000/0 F2

@... 3886 36427 10% [0] 00:35:27 00-00-2000/0 F2

@M.. 4022 36498 11% [0] 00:35:28 00-00-2000/0 F2

@M.. 3749 36496 10% [0] 00:35:29 00-00-2000/0 F2

@MH. 3970 36427 10% [0] 16:35:56 28-08-2020/5 F2

@MH. 3818 36496 10% [0] 16:35:57 28-08-2020/5 F2

@MHY 3947 36496 10% [0] 16:35:58 28-08-2020/5 F2

@MHY 0 36457 0% [2] 16:35:59 28-08-2020/5 F2

DCF77 Lib OK --> 16:35 28-08-2020

@MHY 3672 34703 10% [0] 16:35:00 28-08-2020/5 F2

TIME & DATE OK --> 16:35 28-08-2020

@... 9733 37354 26% [1] 00:00:01 00-00-2000/0 F2

@... 4119 35355 11% [0] 00:00:02 00-00-2000/0 F2

@MHY Minuut, uur en jaar zijn goed gedecodeerd.

Als de letter klein is geprint dan is die tijdseenheid niet goed gedecodeerd.

Bijvoorbeeld @MhY. De uren zijn mogelijk onjuist gedecodeerd

In de seconderegel: “@MHY 3947 36496 10% [0]” zijn 3947 positieve pulsen ontvangen van de 36496 getelde pulsen. 10% is positief wat overeenkomt met 0.1 seconde. Dit is een 0 bit

[0] 0 bit

[1] 1 bit

[2] is het 0 seconde signaal

[9] is een ontvangst waarvan de pulselengte buiten zijn grenzen valt.

Na 20 seconden komen de eerst bits van de tijdseenheid binnen. Na 58 seconden kan de tijd en datum berekend worden.

De tijd en datum zie je per seconde zich ontwikkelen.

Als alle controles goed zijn komt er een OK van de Arduino DCF77 routine (DCF77 Lib OK --> ) of van het eigen geschreven algoritme (TIME & DATE OK -->).

Na elke minuut volgt een regel met statistiek.

@MHY    0 26029  0% [2] 20:35:59 29-08-2020/6 F4  
--> EdT:39 Ed:1026 Th:1416 Both:870 Valid:1572 Min:1738 OK:0 Lock:Y  
Er zijn 1738 minuten verlopen en daarin zijn 1572 geldige tijd-datums gedecodeerd.

1416 zijn er door het Arduino DCF77-algoritme goed ontvangen, 1026 door de zelf geschreven routine. 870 daarvan zijn gelijktijdig goed gedecodeerd en van 39 wist de zelf geschreven routine nog een juiste tijd te ontcijferen.  
De software schrijft alleen een geldige tijd naar de RTC als beide algoritmes tegelijk de zelfde tijd en datum ontvangen.

D D15122017 is date 15 December 2017

Voer de datum in als Dddmmyyyy.

G DCF-signal info in display

Dit heeft geen functie in deze klok.

De ontvangst wordt altijd met de oranje LED op de Arduino Nano weergegeven

H Use DCF-receiver

Zet de DCF-ontvanger uit.   
Door H weer in te voeren gaat de ontvanger weer aan.

Lnn (L5) Min light intensity ( 1-255 bits)

Voer de laagste lichtsterkte in bits in door L met een getal van maximaal 255. Bijvoorbeeld: L12

L5 in een redelijke waarde zodat de klok midden in de nacht niet te fel brandt

Mnn (M90)Max light intensity (1%-250%)

Met M en een waarde tussen 1 en 250 kan de lichtsterkte worden ingesteld.

M50 is een goede startwaarde.

Pnnnnnn (P234F8A) own colour (n=0-F)

Je kan een eigen kleur voor de verlichting instellen. Deze wordt als hexadecimale waarde ingevoerd.

De kleuren in de LED zijn RGB, rood, groen, blauw die een waarde tussen 00 en FF (=255) kunnen hebben. De volgorde is Prrggbb

PFF0000 = rood

P0000FF = blauw

P00FF00 = groen

P66CDAA = aquamarine

De klok dimt lichtsterkte door deze RGB-waarde te verkleinen. Bij de kleur aquamarine zal het rood dus als eerste gaan verdwijnen als het donker wordt.   
De minimale lichtsterkte ingesteld met de optie Lnn kan kleurverlies voorkomen.

Qn Display Choice (Q0-6)

Q0= Yellow colour, HETISWAS changing

Q1= Hourly colour

Q2= All white

Q3= All Own colour

Q4= Own colour, HETISWAS changing

Q5= Wheel colour

Q6= Digital displayKies het gewenste kleurschema met Q0 – Q7

I For this info

Print het menu door I in te voeren

R Reset to default settings

Voer R in als je de klok ontregeld hebt.

Xnn (X50) Demo mode. ms delay (0-9999)

Door X en een waarde in te voeren gaat de klok snel in de tijd vooruit. De waarde is het aantal milliseconden die een minuut duurt op de klok.

Druk X weer in om te stoppen

Z for Self test

Alle LEDs laten een spektakel zien.

De Z weer om te stoppen. Dit kan even duren omdat de zelftest ongeveer een minuut duurt

**DCF77-ontvanger**

De klok bevat een DCF-77 atoomkloktijdontvanger.   
In de software zijn twee methodes gemaakt die het tijdsignaal decoderen.

De eerste is de standaard DCF77 library-routine van het Arduino-platform

De tweede is zelf geschreven. Door beide methodes te combineren wordt de tijd sneller en vaker goed ontvangen.

De ontvangst is te volgen door in het menu de optie A te kiezen

Elke seconde wordt dan een regel geprint

@mHY 5062 36495 13% (0) 14:31:58 28-08-2020/5 F2 Ed:33 Th:28 Min:40 OK:0 L:Y

@MHY is minuut uur jaar. Als de letter klein is dan is de decodering hier van niet optimaal,  
Daarna volgt het aantal hoge pulsen gevolgd door het totaal aantal metingen. Daarna volgt het percentage positieve pulsen. Deze zijn optimaal 10% en 20% voor respectievelijk een (0) en een (1) bit. Dan volgt de tijd en datum/weekdag tot zover gedecodeerd. Je ziet per seconde zich de tijd en datum ontwikkelen. F is het aantal fouten per minuut. Daarna hoe vaak de zelf geschreven decoder en die van de Arduino de tijd goed hebben ontvangen gevolgd door het aantal minuten dat de klok aanstaat.

OK: en L: zijn interne controles

Op de Arduino knippert een oranje LED die het positieve signaal weergeeft. Als je goed kijkt zie je het verschil tussen 0.1 en 0.2 seconde (10% en 20%); de 0 en de 1

Als het lampje onregelmatig knippert is de ontvangst niet perfect.

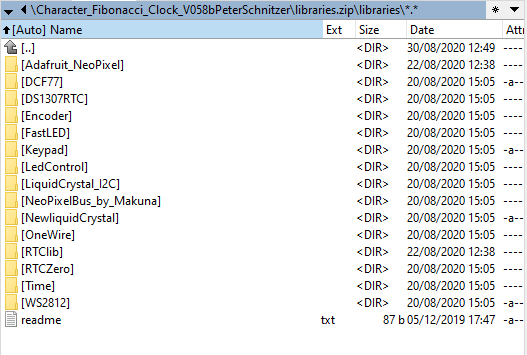
Ed Nieuwenhuys

[ednieuw@xs4all.nl](mailto:ednieuw@xs4all.nl)

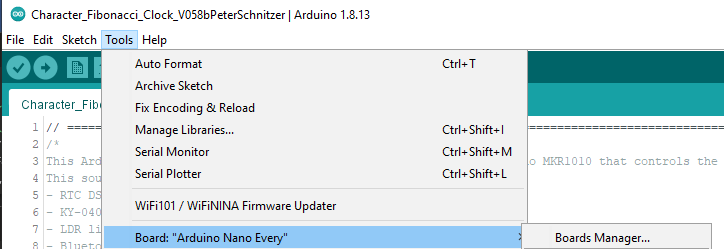
06 236 10 994

Installatie en uploaden van de software

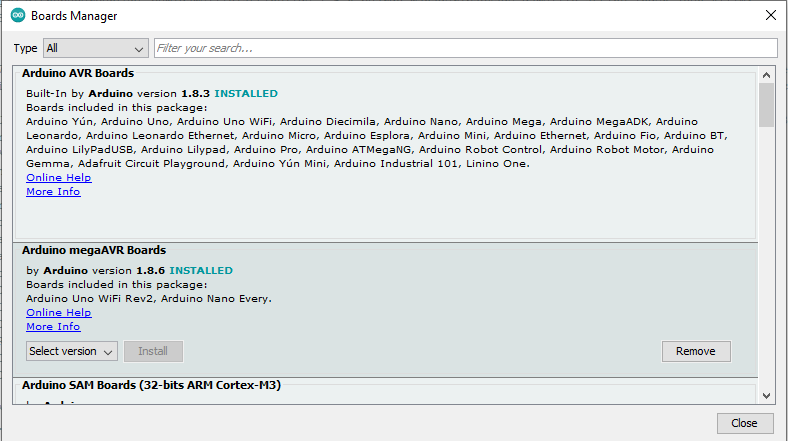
In de Arduino-folder staat een folder libraries. Kopieer uit de zip de folders in deze libraries folder



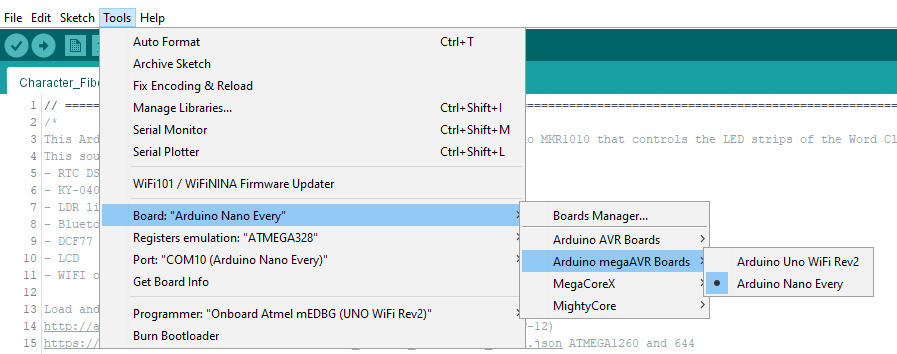
Open in Tools -? Board -> Boards manager



Installeer Arduino megaAVR Boards

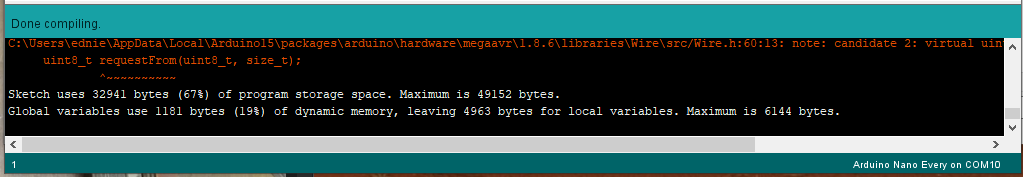


Daarna kan je de Arduino Nano Every kiezen



Druk op “V” links boven in de IDE om te compileren

Zie je dit?



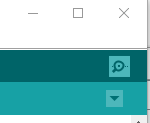
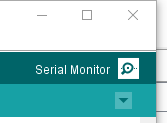
Nee??? Ed bellen

Zo ja

Kies de COM-poort van de Every (bij mij COM10)

Druk -> om te uploaden

Druk serial monitor.



Dit is je programma en de libraries in het Word document opgenomen.



