Deze woordklok toont de tijd als tekst.   
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 voor een Arduino Nano Every met als softwareversie: Character\_Colour\_Clock\_V068.  
De klok bevat een DS3231 tijdmodule waarin een CR2032 of LIR2032 batterij de spanning verzorgt als de klok uit staat. Deze tijdmodule verloopt enkele seconden per jaar maar voorziet niet in de zomer-wintertijdwissel.  
Als de klok na een stroomonderbreking zijn tijd niet meer vasthoudt dient de batterij vervangen te worden.

De optionele DCF77-ontvanger moet voor een goede ontvangst dwars op de zender in Mainflingen, Duitsland, georiënteerd te worden. Dat is grofweg als de zijkanten van de klok naar west en oost wijzen. De DCF77-ontvanger dient minimaal 10 cm van de, signaal storende, LEDs te hangen. Dat is meestal buiten de klok.

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 het programma “BLE terminal HM-10” gebruikt worden.

Voor de Android is “Serial Bluetooth van Kai Morich” een goede app.

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 ingesteld worden.

Door 15 keer op de draaiknop te drukken of in het menu de R van reset in te voeren worden de instellingen naar de oorspronkelijke waarden teruggezet.

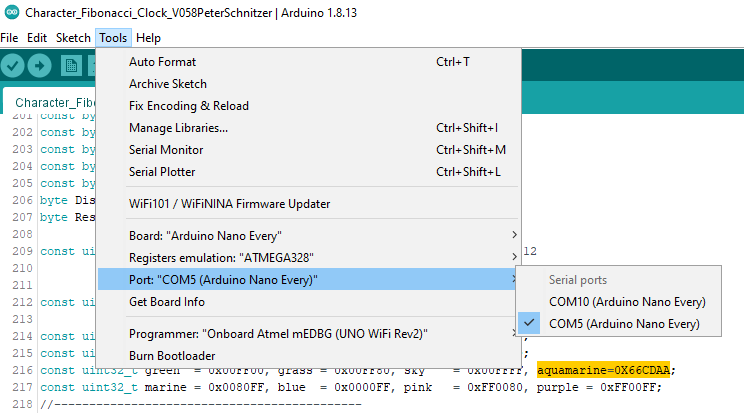
De klok onthoudt de ingestelde waarden van L, M, P en Q in het menu als de stroom van de klok wordt gehaald. Dit is de minimale en maximale lichtsterkte, eigen kleur en displaykeuze.  
Met reset (R) in het menu kan de klok weer worden hersteld naar de oorspronkelijke instellingen

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 worden gedownload.

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

Kies Tools->Serial monitor om de seriële monitor te openen om verbinding met de klok te maken



Met een Bluetooth-verbinding op de telefoon of met een kabel aan de PC kan de klok ook aangestuurd worden.  
Nadat verbinding is gemaakt wordt het volgende getoond.

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

Serial started

RTC DS3231 enabled

Bluetooth enabled

LEDs SK6812 enabled

DCF77 enabled

DCFtiny enabled

Compiled for ARDUINO AVR NANO EVERY

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

Woordklok RGB

Enter time as: hhmmss (132145)

A Debug DCF-signal

D D15122017 is date 15 December 2017

G DCF-signalinfo in display

H Use DCF-receiver

L (L5) Min lightintensity(0-255 bits)

M (M90)Max lightintensity(1%-250%)

N (N2208)Turn On/OFF LEDs between Nhhhh

P (P00234F8A) own colour (n=0-F)

Q 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 display

I For this info

R Reset to default settings

S Self test

W Test LDR reading every second

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

Ed Nieuwenhuys March 2021

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

Brightness Min: 5 bits Max: 40%

LDR read Min: 999 bits Max: 1 bits

LEDs off between: 00 - 00

Display choice: 0

Software: Character\_Colour\_Clock\_V069

10:22:54 23-04-2021  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Sensor: 0 Min:409 Max:409 Out: 64=25% Temp:20C 10:22:54

Het was tien voor half elf 10:22:54 23-04-2021

--> EdT:2 Ed:0 Th:0 EdW:0 ThW:0 Both:0 Valid:1 Min:1 Lock:N

Sensor: 0 Min: 84 Max:409 Out: 29=11% Temp:20C 10:23:00

Het was tien voor half elf 10:23:00 23-04-2021

Als de seriële monitor wordt gebruikt wordt de klok gereset. Je ziet dan welke componenten geïnstalleerd zijn. Met een Bluetooth-verbinding treedt de reset niet op.

Elke 30 seconden wordt een regel geprint

DCF77 Lib OK --> 14:46 28-08-2020

TIME & DATE OK --> 14:46 28-08-2020/5

LDR:549 bits = 29% light intensity. Temp=21C DCF:7% 14:46:00

twee uur en 46 minuten 14:46:00 28-08-2020  
……

LDR:308 bits = 21% light intensity. Temp=21C DCF:20% 15:02:30

DCF77 Lib OK --> 15:03 28-08-2020

LDR:310 bits = 21% light intensity. Temp=21C DCF:21% 15:03:00

drie uur en 3 minuten 15:03:00 28-08-2020

LDR:317 bits = 21% light intensity. Temp=21C DCF:21% 15:03:30

DCF77 Lib OK --> 15:04 28-08-2020

TIME & DATE OK --> 15:04 28-08-2020/5

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 en hoe goed de DCF-ontvanger het laatste uren gewerkt heeft (0-99%)

**Draai of drukknop**

De klok kan in plaats van zijn uitgerust met drie membraanknoppen. De middelste knop van de drie is hetzelfde als drukken op de rotary draaiknop. Drukken op buitenste knoppen is gelijk aan links of rechtsom draaien aan de draaiknop.

Druk op de knop van de drukknop om het keuzemenu te activeren.

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

Een tweede druk op de knop laat HET IS WAS even 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.

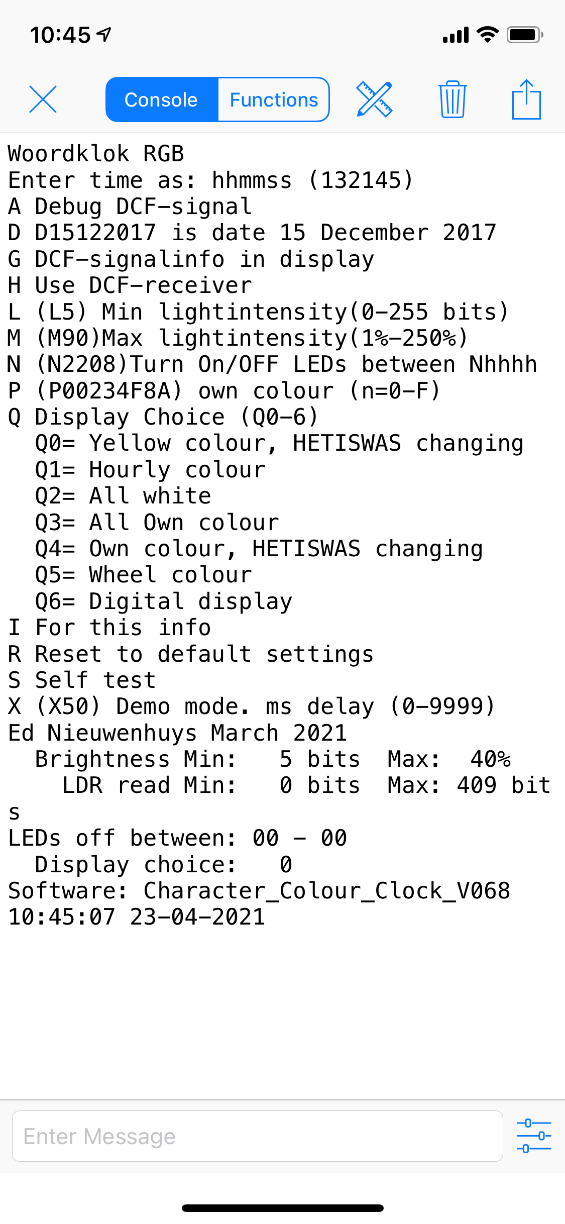
5 Elk uur een andere kleur van de tekst.

6 Alle tekst in wit.

7 Alle tekst kan een eigen samengestelde kleur aannemen.

Deze eigen kleur kan worden gemaakt door in het menu van de klok, nadat een verbinding met Bluetooth of de PC is gelegd is, Prrggbb te tikken. De rr, gg en bb zijn de primaire kleuren rood, groen en blauw. Deze primaire kleur 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. Op het internet zijn de RGB-waarden voor een kleur te vinden. (color picker)

8 Eigen kleur, HET IS WAS veranderen van groen via geel naar rood.  
 HET verandert in een minuut en IS en WAS in een uur van groen naar rood

9 De regenboogkleuren worden in een klein uur doorlopen10 Digitale display indien achter alle letters een LED is.

14 keer drukken reset alle waarden in de software naar

default settings

**Bluetooth of seriële verbinding**

Maak verbinding met de klok. Serieel of met bluetooth  
Stuur de i van informatie over om het menu in beeld te krijgen

De instellingen kunnen aangepast worden met een letter vaak gevolgd door een waarde.

Tik dit in en druk op return of verzenden.

In het menu kan bijvoorbeeld Q0 tot en met Q6 ingevoerd worden voor de diverse kleurpaletten.

Q0= Yellow colour

Q1= Hourly colour

Q2= All white

Q3= All Own colour

Q4= Own colour, HETISWAS changing

Q5= Wheel colour

Q6= Digital display

De instellingen worden permanent in het geheugen opgeslagen.

Met Reset de oorspronkelijke instellingen teruggezet.

**Menu-opties**

Enter time as: hhmm (1321)or 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 Every rechts naast de USB-aansluiting 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 (66) als eerste gaan verdwijnen als het donker wordt.   
De minimale lichtsterkte, ingesteld met de optie Lnn, kan dit kleurverlies voorkomen.

Qn Display Choice (Q0-6)

Q0= Gele kleur HET IS WAS verkleurt van groen naar rood

Q1= Per uur een kleur

Q2= Alles wit

Q3= Eigen kleur

Q4= Eigen kleur HET IS WAS verkleurt van groen naar rood

Q5= Wheel kleur. Regenboogkleuren per uur

Q6= Digitale display als speciale uitvoering van de klok

Kies het gewenste kleurschema met Q0 – Q6

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.

W Test LDR reading every second

Voer W in om elke seconde de uitlezing van de LDR te kunnen volgen.

Xnn (X50) Demo mode. nn=milliseconden 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**

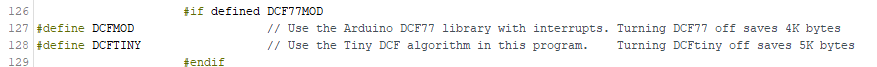
De klok kan een DCF-77 atoomkloktijdontvanger hebben.

In de software zijn twee methodes beschikbaar die het tijdsignaal decoderen.

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

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

De ontvangstmethode(n) wordt gedefinieerd in het programma.



De ontvangst is te volgen door in het menu de optie A te kiezen.   
(als de #define DCFTINY optie niet is uitgezet in het programma.)

Elke seconde wordt dan een regel geprint

@mH. 1606 13332 12% [0] 12:55:57 23-04-2021/5 F52 OK:0

@mHY 1158 13332 8% [0] 12:55:58 23-04-2021/5 F52 OK:0

@mHY 0 13333 0% [2] 12:55:59 23-04-2021/5 F52 OK:0

@MHY is minuut uur jaar. Als de letter klein is dan is de decodering hier van niet optimaal,  
Daarna volgt het aantal ontvangen hoge pulsen gevolgd door het totaal aantal metingen. Daarna volgt het percentage positieve pulsen. Een 0 bit is een korte hoge pulse van rond de 10% (100 msec) en een 1 bit is een hoge pulse van ronde de 20% (200 msec lang).   
Dan volgt de tijd en datum/weekdag tot zover gedecodeerd. Je ziet per seconde zich de tijd en datum ontwikkelen.   
F is het aantal fout ontvangen secondemetingen.

@MHY 1370 13316 10% [0] 12:56:58 23-04-2021/5 F53 OK:0

@MHY 0 13348 0% [2] 12:56:59 23-04-2021/5 F53 OK:1Als de tijd goed ontvangen is dan zijn de MHY in hoofdletter en wordt OK:1

De volgende regel wordt geprint

--> EdT:24 Ed:54 Th:25 EdW:0 ThW:0 Both:15 Valid:65 Min:158 Lock:YEd is de statistiek van de zelf geschreven methode en Th die van Arduino

EdT is de teller van de goed ontvangen tijden

Ed: en Th: zijn de goed ontvangen tijden én datums

EdW:0 ThW:0 : zijn de goed door gegeven tijden en datums. Dit zou 0 moeten zijn maar heel af en toe geven de methoden toch een tijd door waarmee de tijd verzet zou worden.

Als beide methoden aangezet zijn zal de tijd niet aangepast worden.

Both:15 Goed ontvangen tijden en datum op het zelfde moment

Valid:65 Totaal goed ontvangen tijden en datum

Min:158 totaal aantal minute dat de klok aanstaat

Daarna volgt hoe vaak de zelf geschreven decoder (Ed) en die van de Arduino (Th) de tijd goed hebben ontvangen. Als laatste het aantal minuten dat de klok aanstaat.

Lock:Y Als Lock is Y dan is in principe de tijd niet te veranderen. Eens per uur gaat de Lock naar N.

In deze versie wordt lock niet gebruikt en wordt de tijd aangepast als deze meer dan 2 seconden verschilt. Als beide methoden een goede tijd ontvangen dan gaat Lock weer naar Y

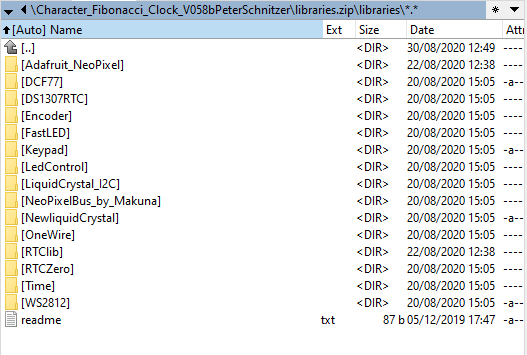
Op de Arduino knippert rechts van de USB aansluiting een oranje LED die het positieve signaal weergeeft. Als je goed kijkt zie je het verschil tussen 0.1 en 0.2 seconde (10% en 20%); de 0 en de 1

Als het lampje onregelmatig knippert is de ontvangst niet perfect.

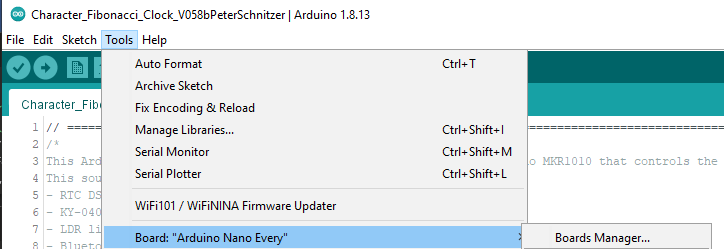
De DCF77-ontvanger dient voor een goede ontvangst dwars op de zender in Mainflingen, Duitsland, georiënteerd te worden. Dat is grofweg als de zijkanten van de DCF-ontvanger naar west en oost wijzen.

**Installatie en uploaden van de software**

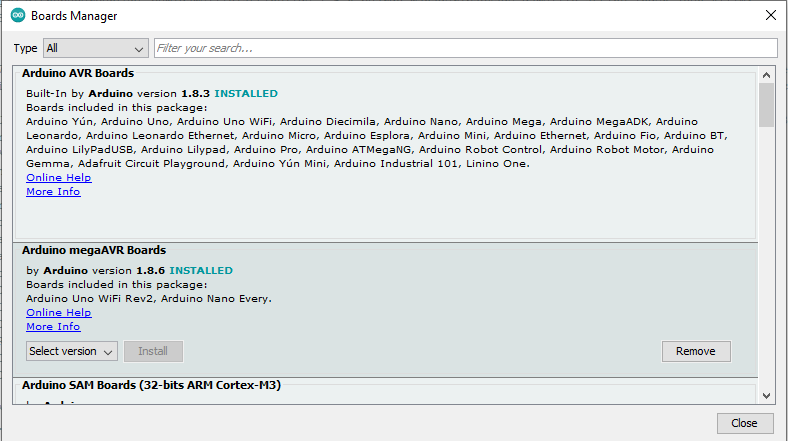
In de Arduino-folder staat een folder libraries. Kopieer uit de zip de folders in deze libraries folder



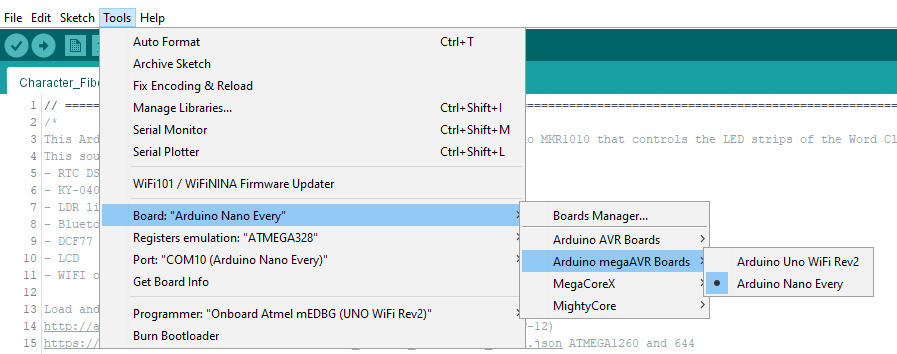
Open in Tools -? Board -> Boards manager



Installeer Arduino megaAVR Boards

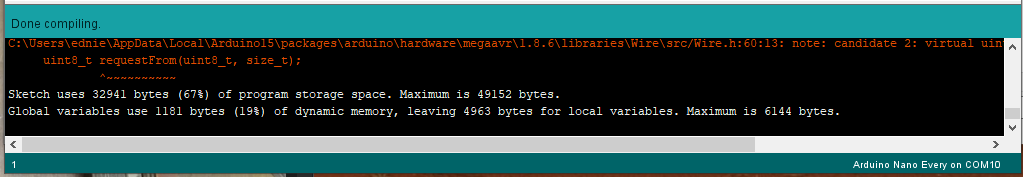


Daarna kan je de Arduino Nano Every kiezen



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

Zie je dit?



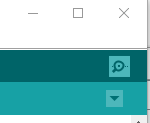
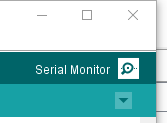
Nee???

Zo ja

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

Druk -> om te uploaden

Druk serial monitor.



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





Ed Nieuwenhuys

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

<https://github.com/ednieuw>

+31 (0)6 236 10 994