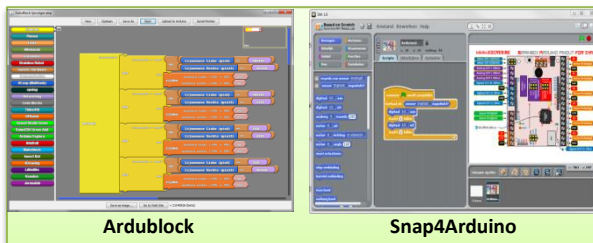


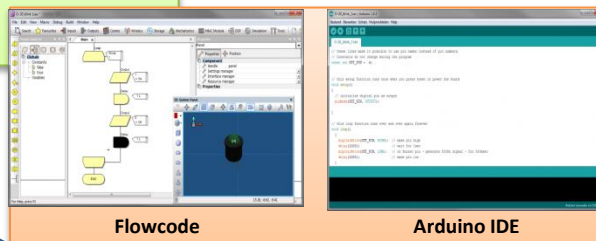
STEM & ARDUINO - EERSTE GRAAD



Ardublock

Snap4Arduino

Eerste graad

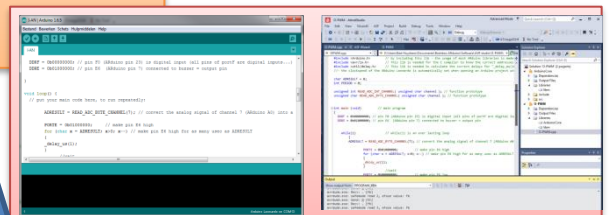


Flowcode

Arduino IDE

Tweede graad

LEERLIJN



Arduino IDE 'C'

AVR STUDIO 'C'

Derde graad

BRAINBOX AVR / ARDUINO & ARDUBLOCK

www.e2cre8.be - barthuyskens@telenet.be

INHOUDSOPGAVE

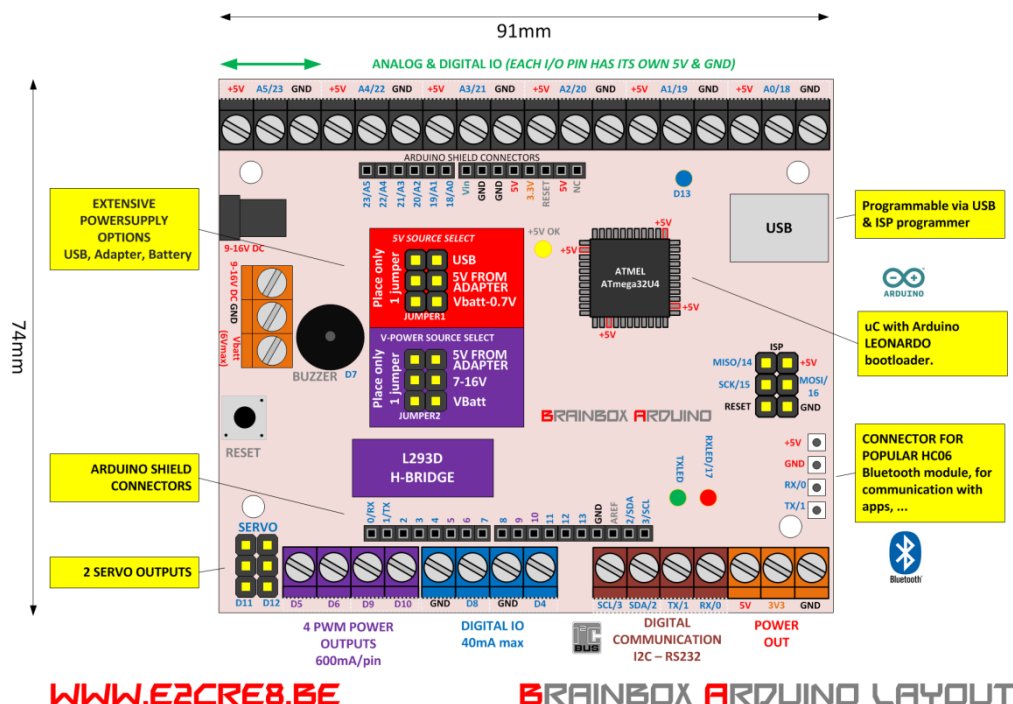
BRAINBOX AVR / ARDUINO.....	3
PINOUT	4
BRAINCHART	4
ARDUBLOCK	4
CONTROLE.....	5
TIJD.....	6
PINNEN	7
TESTS	8
WISKUNDE.....	9
VARIABELEN	10
TEKEN UW EIGEN AANSLUITSCHEMA	11
SCHEMA BBA.....	12
VOEDINGSOPTIES	13
JUMPER 1: 5V SELECTIE	13
JUMPER 2: VOEDING VOOR POWER UITGANGEN	14
VASTE ACTUATOREN	15
3 VASTE LEDS OP DE BRAINBOX ARDUINO.....	15
RINGTONES - GELUID.....	16
OUTPUTS.....	17
O-20 LED AANSTUREN.....	17
O-20 RGB LED AANSTUREN	19
O-500 DC MOTOR	21
O-WARMTE GENEREREN	24
O-SERVO – HOBBY SERVO MOTOREN	26
DIGITALE INPUTS – WAAR OF NIET WAAR.....	28
I-DIG SCHAKELAAR OF DRUKKNOP.....	28
ANALOGUE INPUTS – MEETWAARDE 0-1023	29
I-AN POTENTIOMETER.....	29
I-AN TEMPERATUURMETING MET MCP9700	30
I-AN LICHTMETING MET FOTOTRANSISTOR	31
I-AN AFSTANDMETING MET INFRAROOD LICHT	32
I2C COMMUNICATIE.....	34
I2C LCD	34
LIBRARY INSTALLEREN	34
I2C ADRESSEN.....	35

BRAINBOX AVR / ARDUINO

De Brainbox AVR (2016) is 100% compatibel met Arduino maar is meer dan een gewone Arduino Leonardo. Door de stevige schroefconnectors, de 600mA uitgangen, de servo uitgangen en de vele voedingsopties is deze Brainbox zonder Breadboard of Shields meteen bruikbaar voor de meeste projecten die je kan bedenken. Perfect voor onderwijs en hobbydoeleinden door de stevige opbouw en de grote hoeveelheid gratis lesmateriaal.

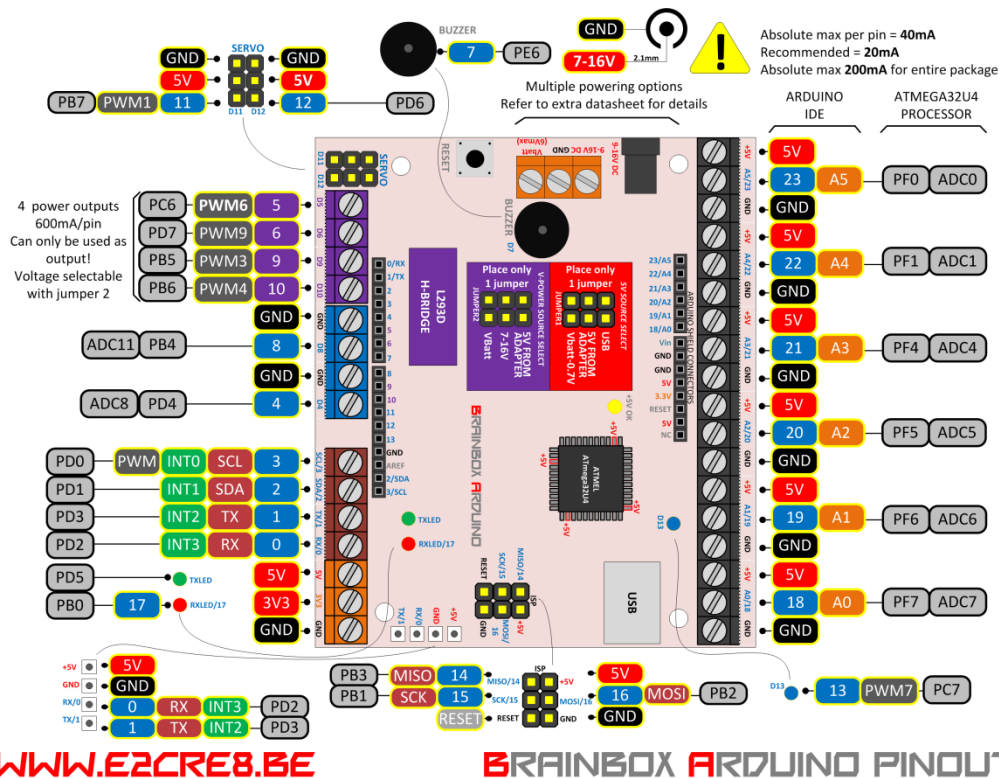
Stevige schroefconnectors
geen breadboard nodig
Analoge en digitale IO pins met eigen voedingspinnen (5V en GND)
4 x 600mA power outputs
2 servo connectors
Connector voor HC06 Bluetooth module (communiceer met je smartdevice)
I2C, RS232 en SPI connectors
Compatibel met Arduino shields
Voeding mogelijk via Adapter, USB, Batterij (6V, 9V, 12V),
Voeding

Programmeerbaar met Arduino IDE,
Flowcode, C en S4A
Veel en gratis Nederlandstalig lesmateriaal – zie onder
Gratis support door de ontwikkelaar zelf via mail
Graphics worden in High-Res aangeleverd



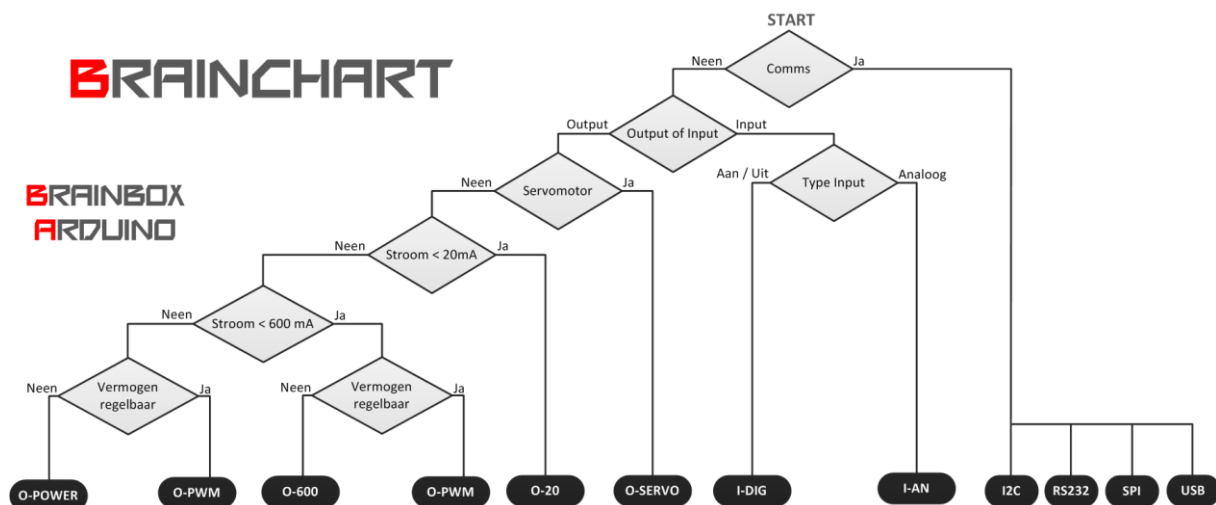
PINOUT

Onder Ardublock gebruiken we de gekleurde PIN nummers (niet die in de grijze kadertjes)



BRAINCHART

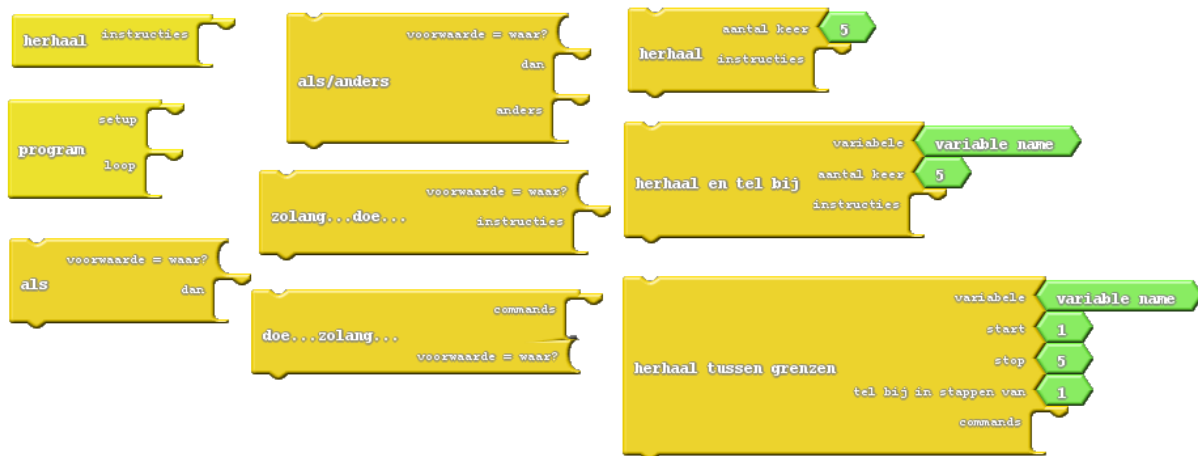
Volg deze flowchart om voor elk deelprobleem een gepaste oplossing te bekomen.



ARDUBLOCK

CONTROLE

Controle instructies worden gebruikt om bepaalde instructies een aantal keer uit te voeren, of om afhankelijk van bepaalde voorwaarden bepaalde instructies wel of niet uit te voeren



- **Herhaal** : Herhaalt onbeperkt de instructies die hieronder geplaatst worden – eeuwige lus.
- **Program** : Dit is de typische arduino methode waarbij setup instructies aan het begin van het programma geplaatst worden en daarna alle andere instructies in een eeuwige lus gezet worden.
- **Als...dan** : Als de voorwaarde 'waar' is (niet 0), dan worden deze instructies uitgevoerd.
- **Als...dan...anders** : Als de voorwaarde 'waar' is (niet 0), dan worden deze instructies uitgevoerd, anders worden de instructies onder 'anders' uitgevoerd.
- **Zolang...doe** : Zolang een bepaalde voorwaarde waar is, worden deze instructies uitgevoerd
- **Herhaal** : Herhaal deze instructies een beperkt aantal keer
- **Herhaal en tel bij** : Herhaal deze instructies een beperkt aantal keer en verhoog de variabele x telkens met 1
- **Herhaal tussen grenzen** : (als start > stop) Geef variabele x bij de start de waarde 1, blijf deze instructies uitvoeren zolang x kleiner of gelijk is aan 5, en verhoog de variabele x telkens met 1
(als start < stop) Geef variabele x bij de start de waarde 5, blijf deze instructies uitvoeren zolang x groter of gelijk is aan 1, en verlaag de variabele x telkens met 1

Subroutines worden door iets meer ervaren programmeurs vaak gebruikt om wat langere programma's op te delen in deelprogramma's of om veel voorkomende zelfde blokken met instructies samen te voegen in één blok dat telkens kan worden aangeroepen.



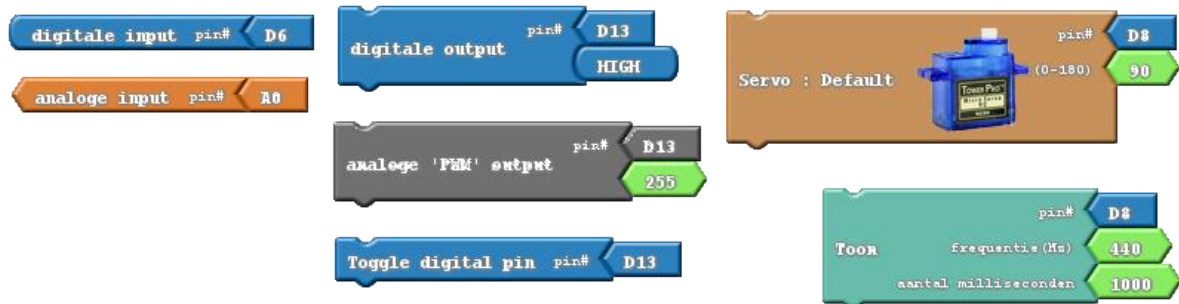
TIJD



- **Delay MILLIS** : zal er voor zorgen dat het programma gedurende dit aantal milliseconden geen andere instructies uitvoert
- **Delay MICROS** : zal er voor zorgen dat het programma gedurende dit aantal microseconden geen andere instructies uitvoert
- **Aantal msec sinds start prog:** geeft in een groot getal weer hoeveel milliseconden gepasseerd zijn sinds de start van het programma
- **Aantal usec sinds start prog:** geeft in een groot getal weer hoeveel microseconden gepasseerd zijn sinds de start van het programma

PINNEN

Deze instructies zijn specifiek om input pins in te lezen of output pins aan te sturen.



INPUT

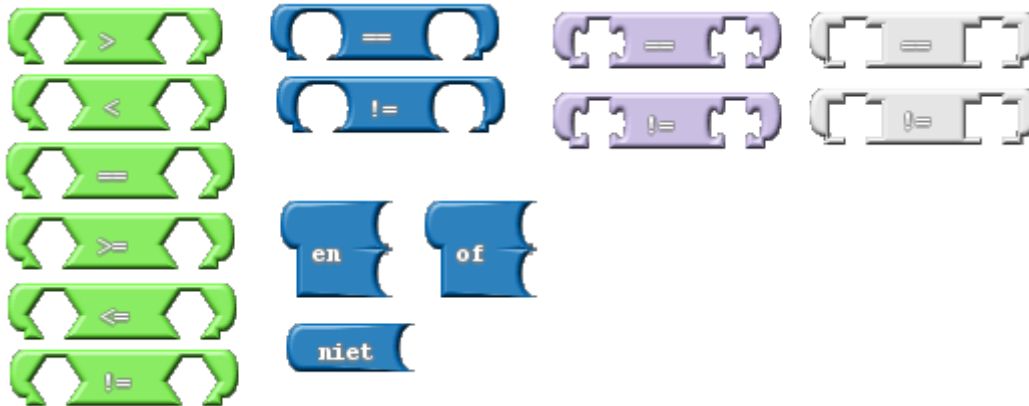
OUTPUT

SPECIALE FUNCTIE

Digitale input	: lees de toestand in, van een digitale inputpin – 1 of 0 – van het type Bool
Analoge input	: Lees de toestand in van een analoge inputpin – 0-1024 – van het type integer
Digitale output	: Maak digitale output pin D2 hoog
Analoge PWM output	: Zet het signaal van de analoge output pin op waarde van 0 tot 255 (PWM signaal)
Toggle digital pin	: verander de waarde van deze digitale pin (0->1 of 1->0)
Servo	: bepaal de hoek van een servomotor op deze pin tussen 0 en 180°
Toon	: genereer een hoorbare toon van 440Hz op deze pin gedurende 1000msec

TESTS

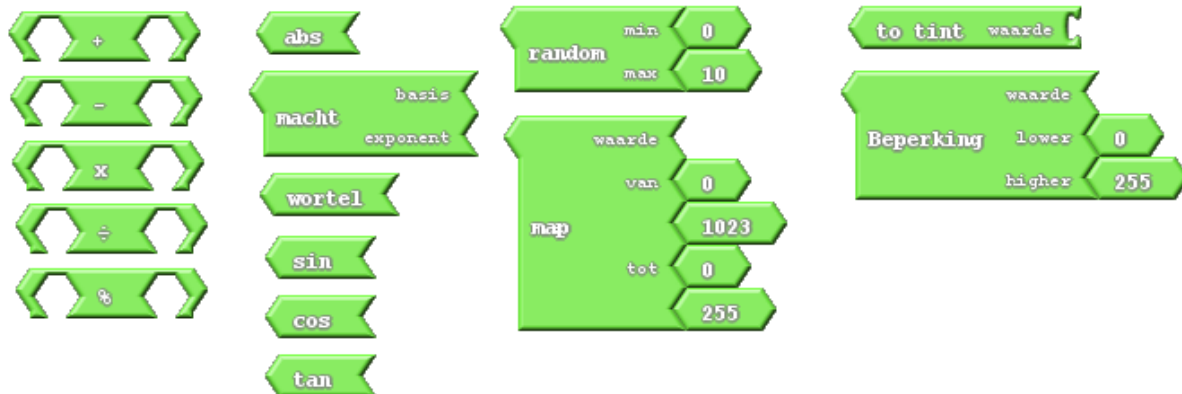
Met deze test instructies kunnen we twee variabelen vergelijken. Voor elk type variabele bestaan er dus ook verschillende instructies met bijpassende vormen. Het resultaat van een vergelijking kan enkel 'waar' of 'niet waar' zijn en is dus steeds van het type 'bool'.



INTEGER	BOOL	CHAR	STRING
• > kleiner dan • < groter dan • == is gelijk aan • >= groter of gelijk aan • <= kleiner of gelijk aan • != niet gelijk aan	• == is gelijk • != is niet gelijk • EN – om tests samen te voegen die beide waar moeten zijn • Of (of fctie 2 tests) • Niet fctie 2 tests	• == is gelijk • != is niet gelijk	• == is gelijk • != is niet gelijk

WISKUNDE

Onze microcontroller kan ook de meest gebruikte wiskundige functies uitvoeren, al moet hier wel gezegd worden dat complexe functies op grote getallen heel wat rekenkracht en dus ook tijd kunnen vragen.



- **+/-/x/:** : normale wiskundige functies op integers
- **%** : Modulus = rest van een deling ($10\%3 = 1$)
- **Abs** : Absloute waarde van een getal
- **Macht** : macht van een getal
- **Wortel, sin, cos, tan** : wortel, sinus, cosinus, tangens van een getal
- **Random** : genereer een willekeurig getal tussen min en max
- **Map** : zet de waarde – die betallen kan bevatten van 0 tot 1024 om naar een evenredige waarde tussen 0 en 255
- **To int** : zet een karakter om naar een waarde
- **Beperking** : beperkt de min en max van een bepaald getal x – als $x > 255 \rightarrow x=255$; als $x < 0 \rightarrow x=0$

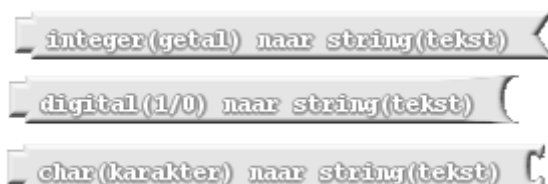
VARIABLEN

Variabelen zijn kleine geheugenplaatsen in de microcontroller die gegevens kunnen onthouden om ze later in het programma terug te gebruiken. Voor verschillende soorten gegevens bestaan er verschillende soorten variabelen:



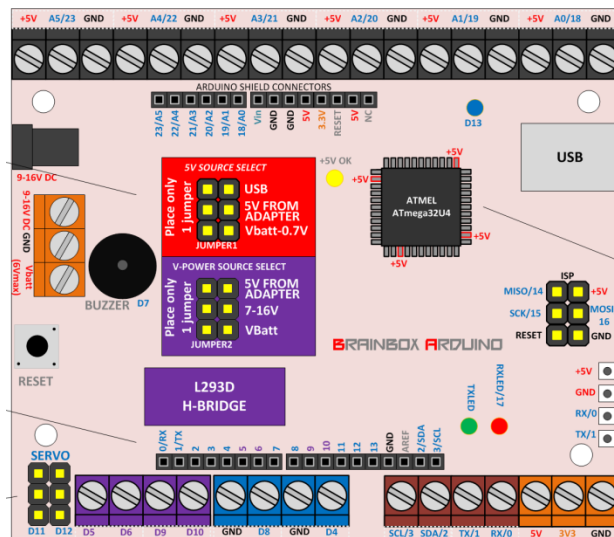
- **Integer** : kan waarden opslaan tussen -32,768 en 32,767 (merk op dat de vorm driehoekig is)
- **Bool** : kan enkel 1 of 0 opslaan – waar of niet waar (merk op dat deze vorm rond is)
- **Decimal number** : kan kommagetallen opslaan (**ook double genoemd**)
- **Char** : kan een karakter opslaan
- **String** : kan een tekst (opeenvolgende karakters) opslaan

Variabelen kunnen ook worden omgezet naar andere types variabelen, meestal met het doel om ze ergens als tekst te tonen.

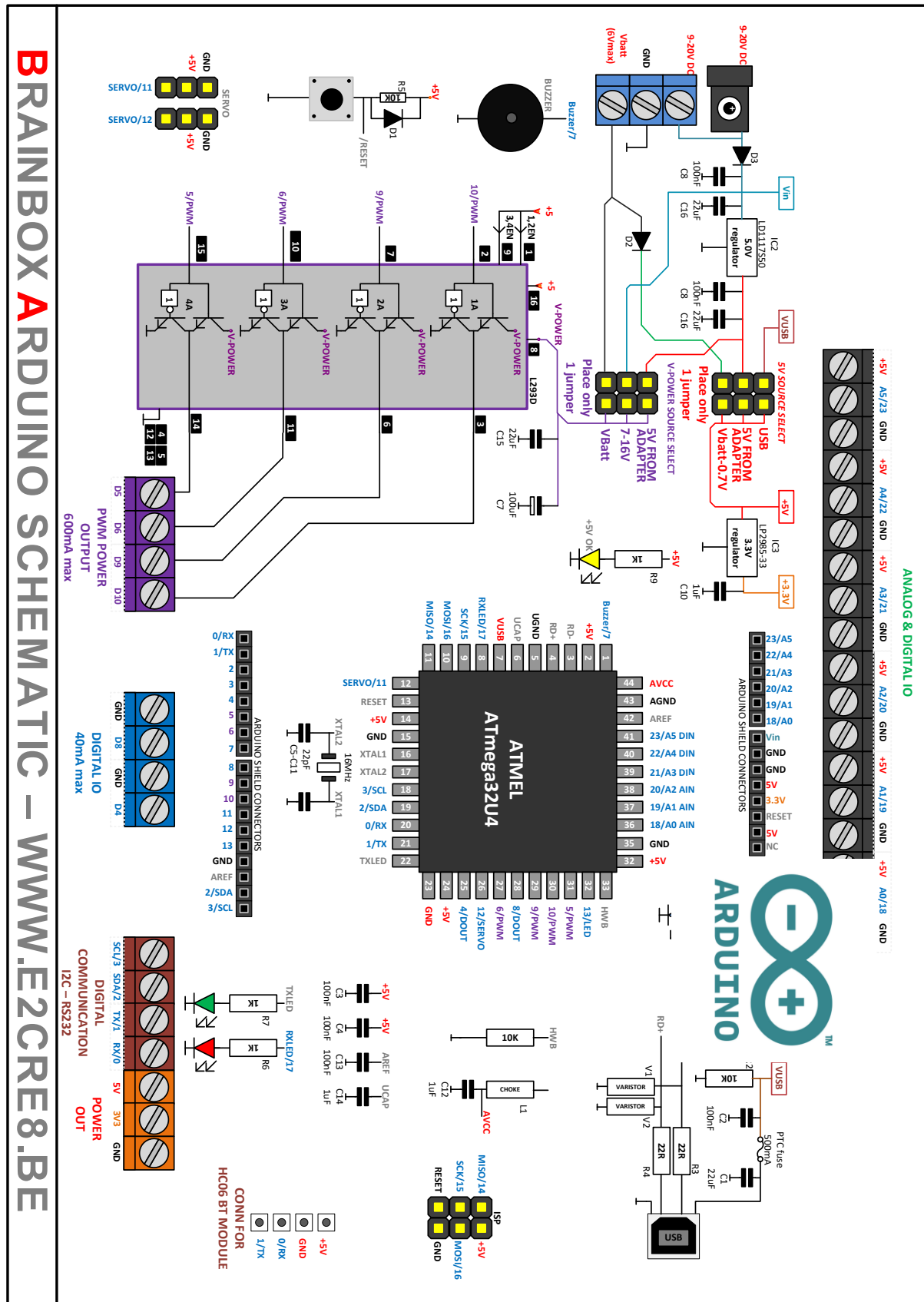


- **Integer naar tekst** : zet een getal van het type integer om naar een tekst zodat deze op een scherm of LCD kan getoond worden.
- **Digital naar tekst** : zet een getal van het type digital om naar een tekst zodat deze op een scherm of LCD kan getoond worden.
- **Char naar tekst** : zet een karakter van het type char om naar een tekst zodat deze op een scherm of LCD kan getoond worden.

TEKEN UW EIGEN AANSLUITSCHEMA



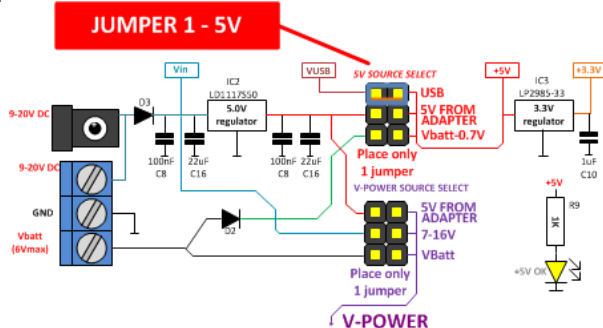
SCHEMA BBA



VOEDINGSOPTIES

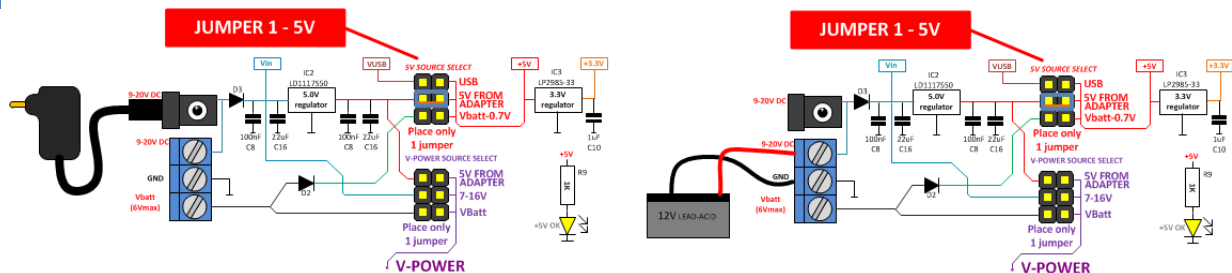
JUMPER 1: 5V SELECTIE

JUMPER 1 OP POSITIE: "USB"



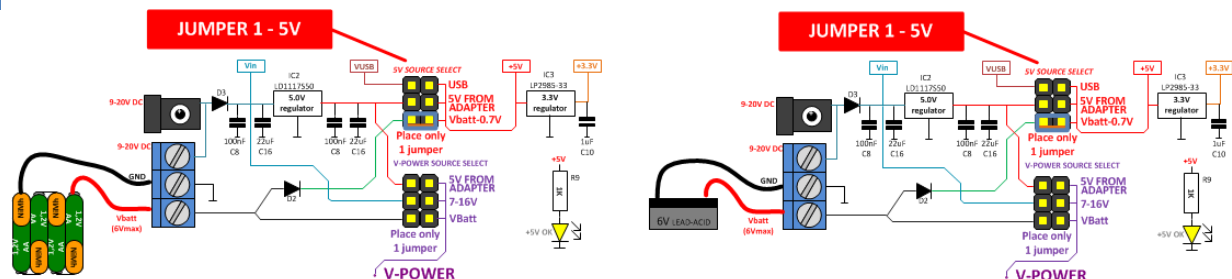
- De 5V komt nu vanuit de USB poort van de PC. De voorwaarde is wel dat de USB kabel is aangesloten.
- I_{max} voor een USB poort ligt op 500mA – een self resetting fuse op de Brainbox zal deze stroom beperken tot die 500mA.
- De stroom voor de H-brug kan nooit uit de USB poort getrokken worden

JUMPER 1 OP POSITIE: "5V FROM ADAPTER"



- De voeding komt van een adapter of van een batterij.
- De adapter moet minimaal 6V en mag maximaal 15V leveren.
- De batterij moet minimaal 6V en mag maximaal 15V leveren.
- De LD1117S50 spanningsregelaar moet hiervan een stabiele 5V maken. De maximale stroom door deze regelaar is bepaald op 500-600mA.

JUMPER 1 OP POSITIE: "VBATT-0.7V"



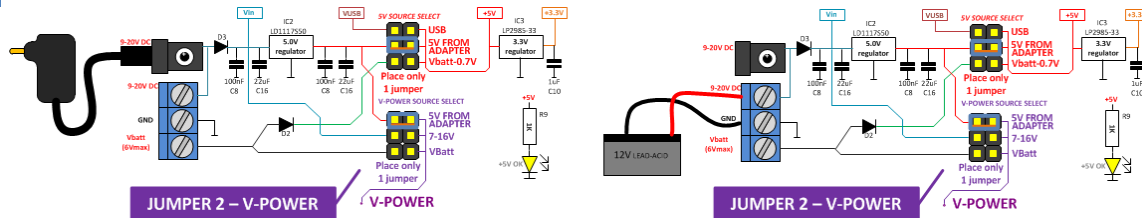
- Links: Situatie met 4x AA batterij in serie (opgelet: volle oplaadbare AA batterijen hebben helemaal in het begin een spanning van 1.55v/cel. $4 \times 1.55V/cel = 6.2V$)
- Rechts: Situatie met een 6V loodzuur accu

Enkel spanningen die tussen 4.5V en 6.2V liggen mogen op deze manier worden aangesloten. De diode vlak na deze ingang zal deze spanning met 0.7V verminderen om zo de uC nooit boven de 5.5V te voeden.

JUMPER 2: VOEDING VOOR POWER UITGANGEN

Deze 3 jumperpositie dienen enkel om de keuze te maken voor de voedingsspanning van de H-brug IC.

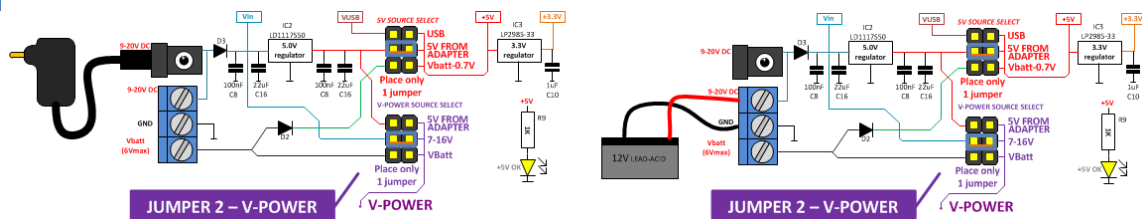
JUMPER 2 OP POSITIE: "5V FROM ADAPTER"



Met de jumper op deze positie wordt de stabiele spanning van 5V, die uit de LD117S50 spanningsregelaar komt, gekoppeld aan de V-POWER ingang van de H-brug. De belastingen aan de H-brug zullen in deze situatie dus met 5V signalen bestuurd worden.

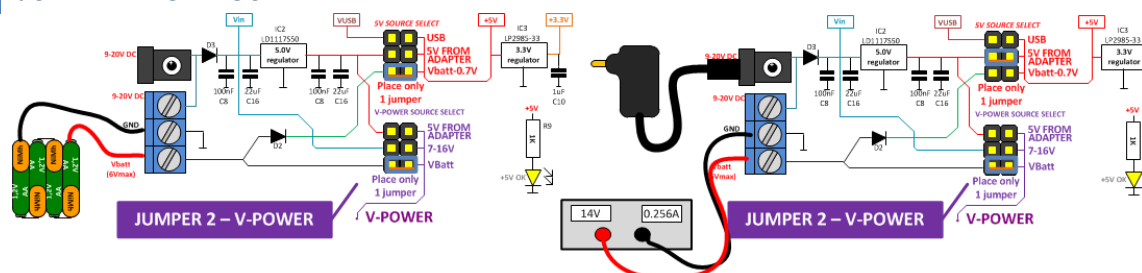
Let er op dat in deze situatie de maximale stroom uit de 4 uitgangen van de L293D, samengeteld, de 500mA niet mag overschrijden.

JUMPER 2 OP POSITIE: "7-16V"



De voeding voor de H-BRUG wordt vlak voor de LD117S50 spanningsregelaar afgetakt. De spanning is nu afhankelijk van de uitgangsspanning van de netspanningsadapter of van de batterij, maar moet steeds boven de 6V liggen voor de 5V voor de uC. Diode D3 haalt er nog 0.7V af.

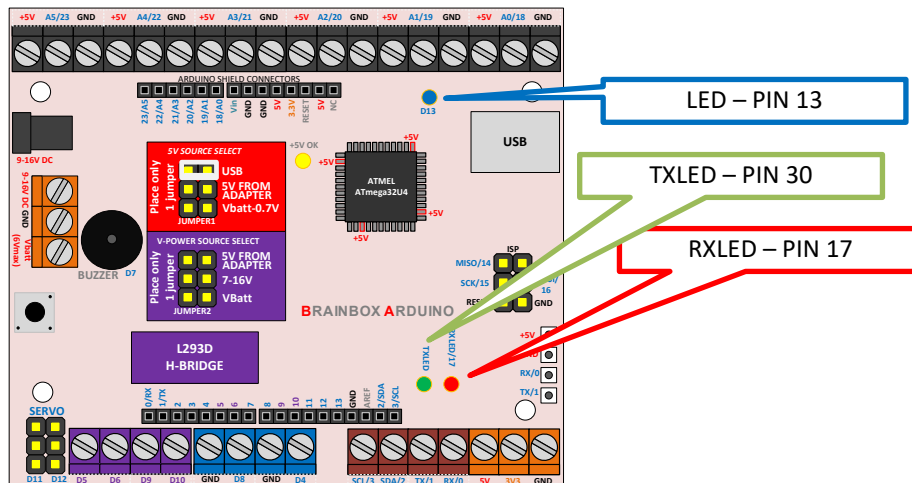
JUMPER 2 OP POSITIE: "VBATT"



VASTE ACTUATOREN

3 VASTE LEDS OP DE BRAINBOX ARDUINO

De led aan **PIN13** en de RXled aan **PIN17** en de TXLED aan **PIN30** zijn vast aangesloten op de uC. We gebruiken deze leds voor onze eerste programmeerervaringen en later ook om fouten op te sporen in grotere programma's.



In een eeuwige loop zullen de 3 leds gedurende 500msec aan zijn, en dan weer 500msec uit zijn.

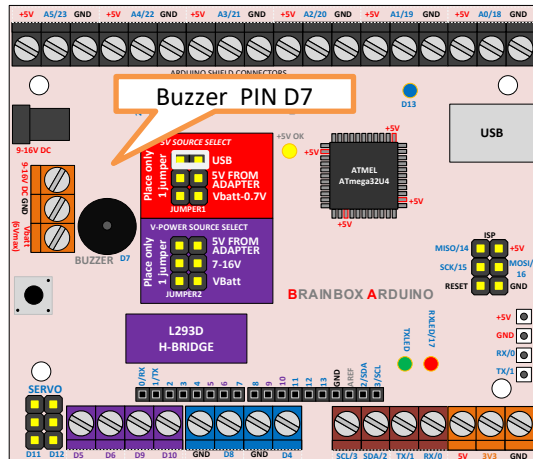


Oef1: maak een looplichtje bl-ro-ge-ro-bl-ro-ge...

Oef 2: test hoe klein je de delays kan maken zodat je geen flikkering meer ziet – je hebt nu de traagheid van je oog bepaald. Wist je dat films ook maar 25 beelden per seconde tonen?

RINGTONES - GELUID

De kleine zwarte cilinder is een luidspreker die hoorbaar geluid kan produceren. Deze hangt aan **PIN7**



Frequentie

- Aantal golven per seconde
- Wordt uitgedrukt in hertz (Hz)

Dit programma is een politiesirene



	Hz	Hz	Hz	Hz	Hz	Hz	Hz	Hz	Hz	Hz
C / B#	16,4	32,7	65,4	130,8	261,6	523,3	1046,5	2093,0	4186,0	8372,0
C# / Db	17,3	34,6	69,3	138,6	277,2	554,4	1108,7	2217,5	4434,9	8869,8
D	18,4	36,7	73,4	146,8	293,7	587,3	1174,7	2349,3	4698,6	9397,3
D# / Eb	19,4	38,9	77,8	155,6	311,1	622,3	1244,5	2489,0	4978,0	9956,1
E / Fb	20,6	41,2	82,4	164,8	329,6	659,3	1318,5	2637,0	5274,0	10548,1
F / E#	21,8	43,7	87,3	174,6	349,2	698,5	1396,9	2793,8	5587,7	11175,3
F# / Gb	23,1	46,2	92,5	185,0	370,0	740,0	1480,0	2960,0	5919,9	11839,8
G	24,5	49,0	98,0	196,0	392,0	784,0	1568,0	3136,0	6271,9	12543,9
G# / Ab	26,0	51,9	103,8	207,7	415,3	830,6	1661,2	3322,4	6644,9	13289,8
A	27,5	55,0	110,0	220,0	440,0	880,0	1760,0	3520,0	7040,0	14080,0
A# / Bb	29,1	58,3	116,5	233,1	466,2	932,3	1864,7	3729,3	7458,6	14917,2
B / Cb	30,9	61,7	123,5	246,9	493,9	987,8	1975,5	3951,1	7902,1	15804,3
Octaaf	0	1	2	3	4	5	6	7	8	9

Oef 1: maak één stabiele toon en verhoog de frequentie telkens met 500Hz tot je de toon niet meer kan horen. Jonge kinderen horen in principe tonen tot 20.000Hz, maar door sterke geluiden neemt je gehoor af. Hoe zit het bij u?

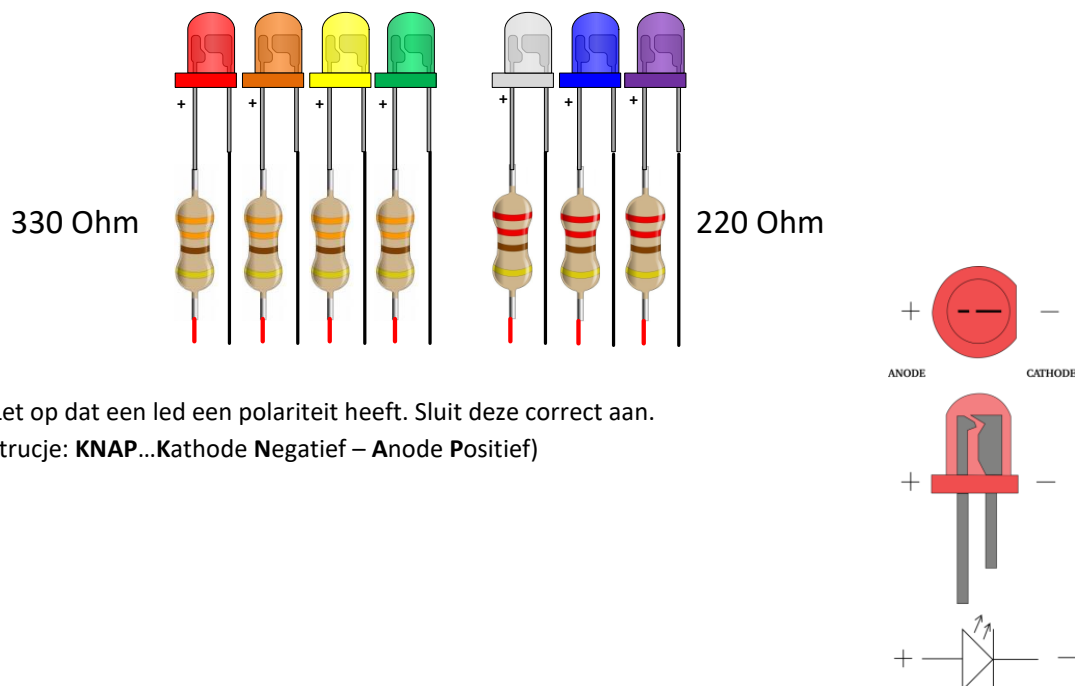
Oef 2: maak je eigen ringtone – zoek een kort muziekstukje op het internet en gebruik deze tabel om dit om te zetten naar een programma. Veel succes.

OUTPUTS

O-20 LED AANSTUREN

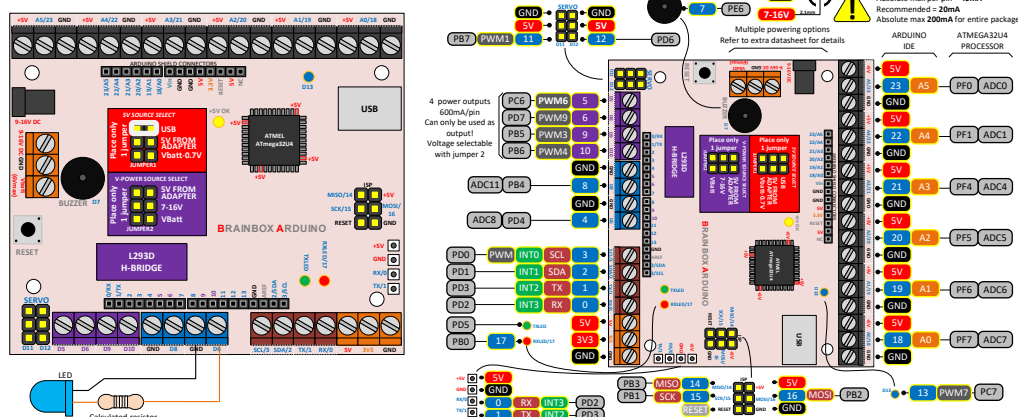
Een led mag nooit meer dan 20mA uit een pin van een microcontroller trekken!

1. Kies een kleur van led.
2. Sluit de juiste voorschakelweerstand aan. Hier horen normaal berekeningen bij, maar met deze vuistregel werkt alles ook behoorlijk en blijft de stroom beperkt tot 10mA ongeveer. Gebruik hiervoor een weerstand van 330 Ohm bij rode, oranje, groene en gele leds en gebruik een weerstand van 220 ohm bij alle andere leds.



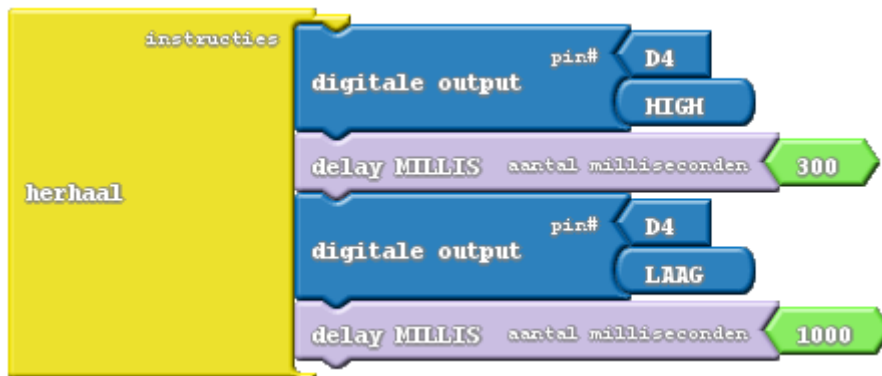
3. Let op dat een led een polariteit heeft. Sluit deze correct aan.
(trucje: **KNAP**...Kathode **N**egatief – Anode **P**ositief)

4. Sluit de led aan op één van de IO pins van de Brainbox Arduino



5. Indien u de led ook wil dimmen, sluit de led dan aan op één van de **PWM** pins. De power output PWM pins mogen hiervoor ook gebruikt worden (D5, D6, D9, D10) maar houd er wel rekening mee dat u de spanning op deze pins moet instellen met jumper 2 (best ook op 5V).

Led tussen D4 en GND zal aan en uit-gaan – telkens gedurende 300msec.



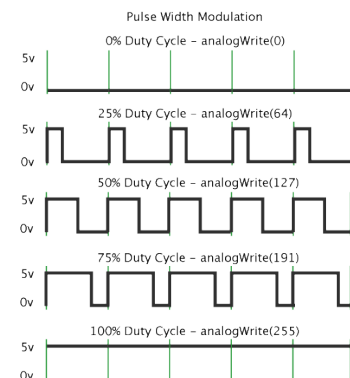
Led tussen SCL/3 (D4 heeft geen PWM functie) en GND zal rustig aan harder gaan branden. 0= helemaal uit, 255 is helemaal aan. In 255 stappen, telkens met een delay van 10msec zal de led harder gaan branden – Na 2.55 seconden herhaalt alles zich terug.



PWM:

PWM staat voor Pulse Width Modulation en dit zal de pin gedurende een bepaalde tijd hoog maken en daarna gedurende een bepaalde tijd laag maken. Hoe langer de pin hoog is – hoe harder de led zal branden. PWM is regelbaar tussen 0 en 255.

Met PWM kunnen we led dimmen, maar ook de snelheid van DC Motoren regelen.

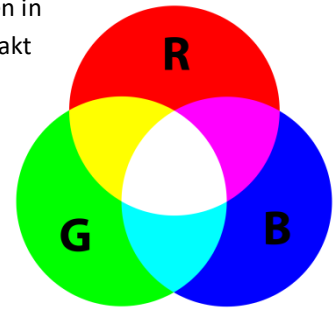
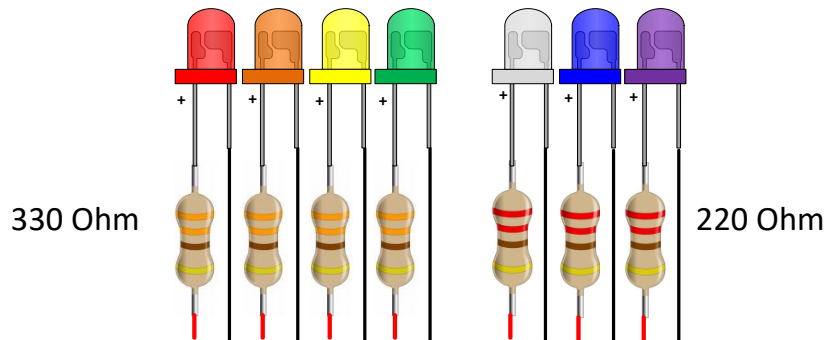


Opgaven:

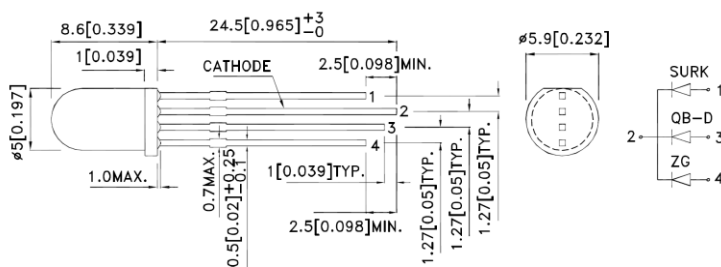
- Maak een verkeerslicht met een rode, een groene en een oranje led
- Gebruik ook een lichtsensor – als het donkerder wordt moet de led harder gaan branden en omgekeerd.
- Gebruik een temperatuursensor – als de temperatuur goed is moet een groene led branden, te hoog = rode led, te laag = blauwe led

O-20 RGB LED AANSTUREN

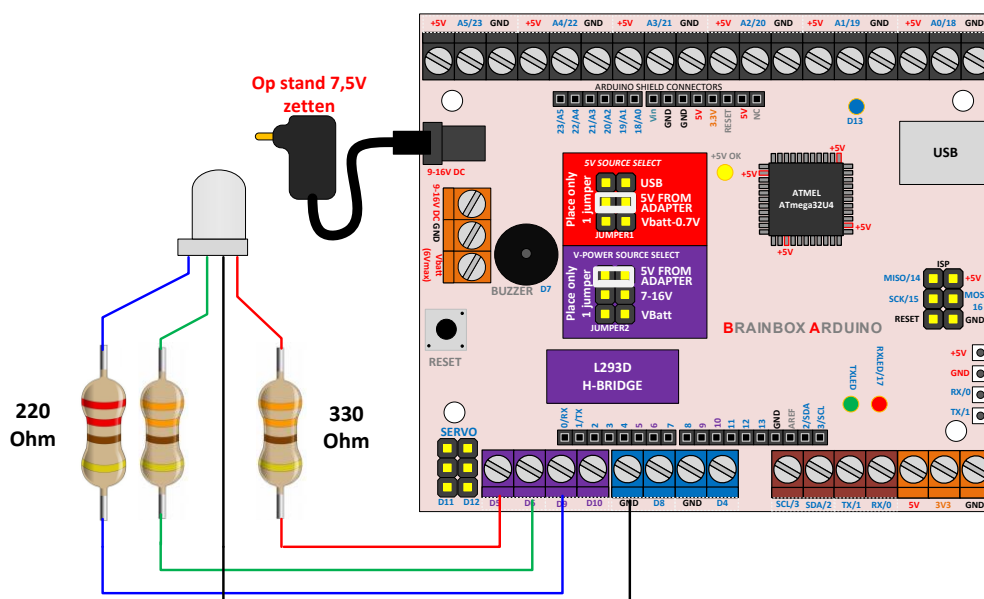
1. RGB leds zijn niet meer dan 3 leds (een rode, een groen en een blauwe led) samen in één behuizing. Door deze 3 kleuren te mengen kunnen alle andere kleuren gemaakt worden.
2. We gebruiken weerom onze vuistregel voor de bepaling van de waarde van de voorschakelweerstand.



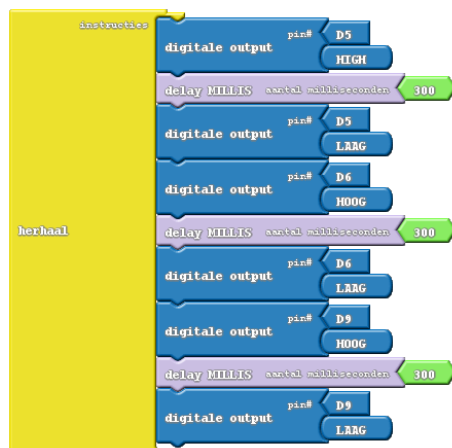
3. Let op de correcte aansluiting van anode en kathode. Er is hier één gemeenschappelijke Kathode voor de 3 leds.



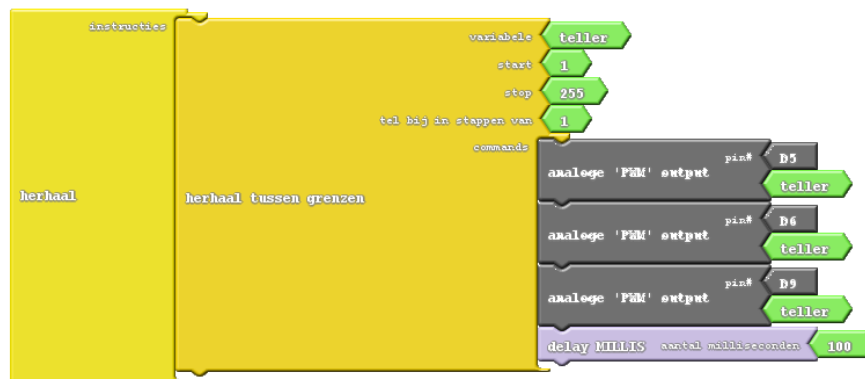
4. In principe kan je deze RGB led op eender welke 3 digitale IO pins aansluiten. Wij kiezen echter om onze RGB led op 3 PWM pins aan te sluiten. Op onze Brainbox Arduino zijn er enkel op de power outputs 3 PWM pins naast elkaar te vinden. Als we jumper 2 op positie "5V from adapter" zetten, dan gedragen deze uitgangen zich als gewone 5V uitgangen. **!! vergeet ook zeker niet dat deze power uitgangen niet werken op USB power – enkel op adapter of battery power.**



De 3 leds van de RGB led gaan afzonderlijk aan en uit.



De 3 leds zullen gelijkmatig harder gaan branden – als ze alledrie op hun maximum staan gaan ze terug uit. We maken hier gebruik van 3 PWM uitgangen. In 255 stappen, telkens met een delay van 10msec zullen de leds harder gaan branden – Na 2.55 seconden herhaalt alles zich terug.

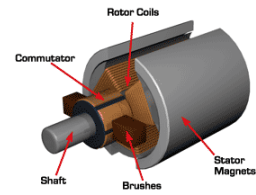


Opgaven:

- De intensiteit van elke led is in te stellen tussen 0 en 255 – door de kleuren van de 3 leds te mengen geeft dat dus $255 \times 255 \times 255 = 16.581.375$ verschillende kleuren. Maak nu je eigen lievelingskleur.
- Gebruik een temperatuursensor – als de temperatuur goed is moet de groene led branden, te hoog = rode led, te laag = blauwe led

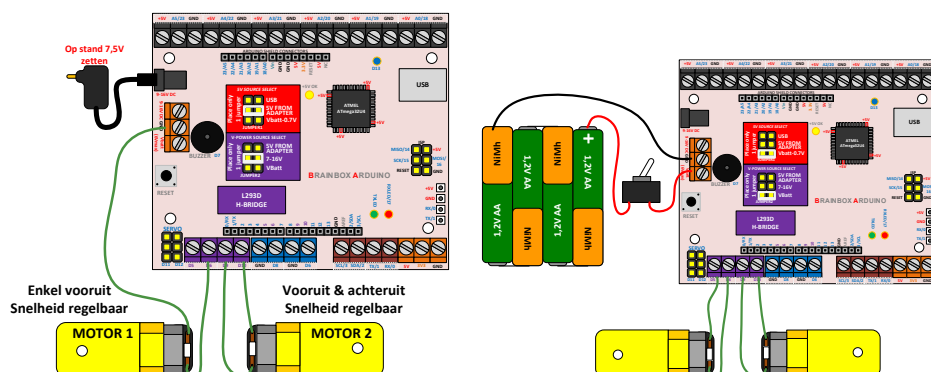
O-500 DC MOTOR

DC motortjes zijn ideaal om bepaalde voorwerpen in beweging te brengen. Ze zijn beschikbaar met of zonder tandwielkast. Ze trekken te veel stroom voor de 20mA uitgangen van de uC. Daarom gebruiken we hier een dubbele H-brug om zo 4 uitgangen van 600mA te hebben. Dat is voldoende voor de meeste kleine DC motortjes. Als je de DC motortjes ook nog in snelheid wil regelen kan je hiervoor best PWM gebruiken.



DC motor	Uit CD speler gesloopt.... (Farnell, DX.com, Conrad, Opitex)
----------	--

1. Sluit je DC motor aan op een afzonderlijke voeding en meet het stroomverbruik in belaste toestand (met de DC motor in de toepassing die je wil aandrijven). Dit stroomverbruik mag niet boven de 500mA uitkomen.
2. Zet jumper 2 in de positie 7-12V. De motor zal nu dezelfde spanning krijgen als de adapter geeft. Zet de adapter op 7Volt
3. Aansluiting DC motor: merk op dat de DC motor op verschillende manieren kan worden aangesloten. Motor 1 gebruikt maar 1 van de 4 uitgangen, maar kan daardoor slechts in 1 richting draaien. Motor 2 gebruikt 2 uitgangen, maar kan daardoor zowel vooruit als achteruit draaien.



Motor 1 zal 3 seconden vooruit draaien en dan gedurende 2 sec stoppen.



(tekening links) Motor 2 zal

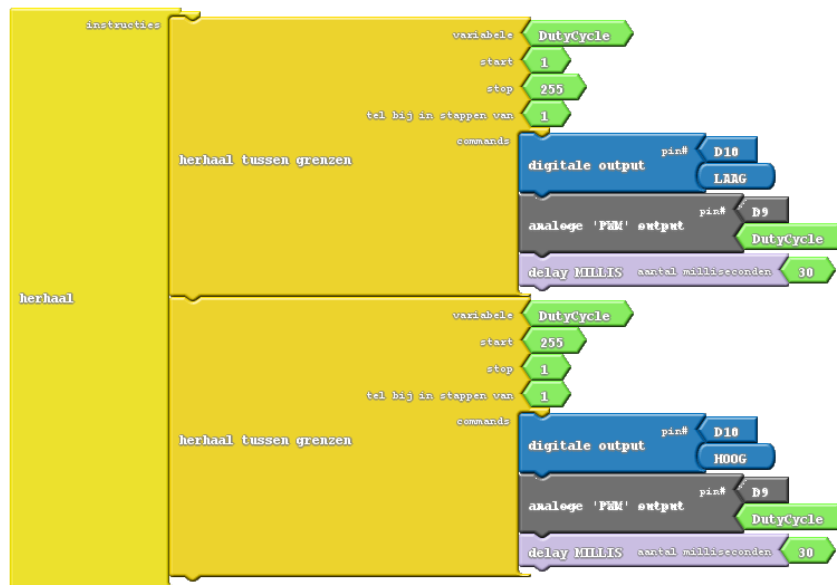
- Eerst 2 seconden stoppen
- Dan twee seconden vooruit draaien tegen maximale snelheid
- Dan terug twee seconden stoppen
- En als laatste twee seconden achteruit draaien tegen maximale snelheid



(tekening links) Motor 2 draait in twee richtingen. In elke richting start de snelheid op 0 en verhogen we die elke 30msec tot die 255 bereikt. Dan keren we de richting om. Met pin D10 bepalen we de draairichting, met pin D9 de snelheid.

Merk op dat:

- Als D10 hoog is, een PWM signaal van 255 betekent dat de motor stilstaat en een PWM signaal van 0 de motor maximaal laat draaien.
- De motor bij lage PWM waarden nog niet draait – dat komt omdat de motor dan nog niet voldoende kracht krijgt om de tandwielkast rond te draaien.

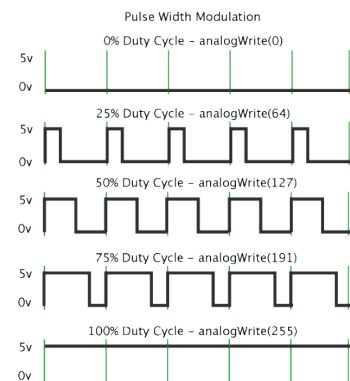


PWM:

PWM staat voor Pulse Width Modulation en dit zal de pin gedurende een bepaalde tijd hoog maken en daarna gedurende een bepaalde tijd laag maken. Hoe langer de pin hoog is – hoe harder de motor zal draaien als de DIR pin laag is. PWM is regelbaar tussen 0 en 255.

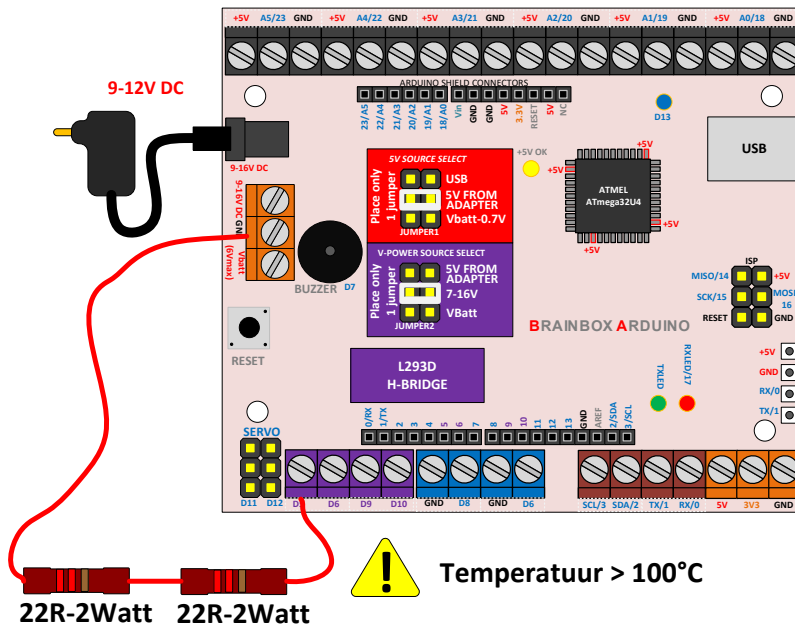
Als de DIR pin hoog is moeten we omgekeerd redeneren: hoe hoger het PWM signaal, hoe trager de motor zal draaien.

De aan-tijd tov de totale tijd is de **duty cycle** en deze wordt uitgedrukt in % of in een getal tussen 0 en 255



O-WARMTE GENEREREN

Om warmte te genereren – bijvoorbeeld om een maquette van een huis te verwarmen – gebruiken we 2 weerstanden van 22 Ohm-2Watt in serie. **Opgelet want deze weerstanden kunnen warmer worden dan 100°C!**



Berekening:

- We voeden de BBA met een adapter of voeding van 12 Volt
- De 4 paarse power aansluitingen kunnen maximaal 500mA stroom schakelen – hier willen we ruim onder blijven. We kiezen een stroom van 300mA.
- $R = U/I$ $12\text{Volt}/300\text{mA} = 40 \text{ Ohm}$.
- 2 weerstanden van 22 Ohm in serie maken een totale weerstand van 44 Ohm – OK
- $P = U \times I$ In deze weerstanden wordt een vermogen gedissipeerd van:
 $12\text{Volt} \times 300\text{mA} = 3.6 \text{ Watt}$
- We selecteren een weerstand met een vermogen van slechts 2Watt – die zal niet stukgaan bij 3.6 Watt, maar zal wel heel warm worden.

Dit programma maakt powerpin D5 – die 500mA kan leveren – hoog. Hierdoor zal er een stroom van iets minder dan 300mA door de weerstanden in serie vloeien. Opgelet, want dit kan deze weerstanden opwarmen tot boven de 100°C.

Uitschakelen doet u simpelweg door deze pin terug laag te maken.



Bij dit programma hebben we de analoge uitgang gebruikt op powerpin D5. Door een getal van 0-255 in te geven kunnen we het vermogen op deze pin regelen tussen 0 en 100%

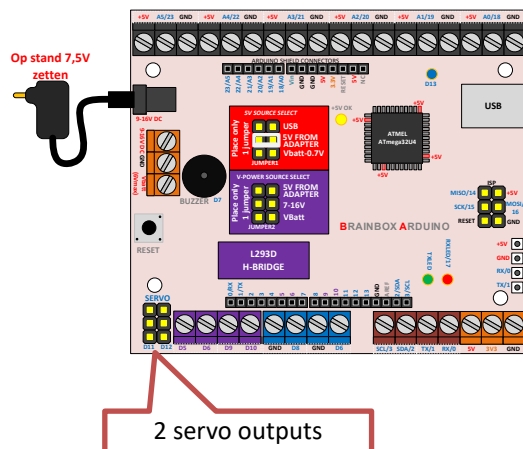
- Als we de variabele 'WARMTE' op 0 zetten, dan zal de weerstand niet opwarmen
- Bij 128 zal deze weerstand 50% van z'n maximale temperatuur bereiken
- Bij 255 zal deze weerstand z'n maximale temperatuur bereiken



Uitdagingen:

- Sluit een potentiometer aan en regel hiermee de temperatuur van de weerstanden tot die exact 60°C is.
- Sluit een temperatuursensor aan en regel daarmee de temperatuur in een schoendoos naar 30°C – of van een kamer in uw maquette naar 21°C.
- Je kan in principe 4 afzonderlijke verwarmingen maken – voor 4 kamers in uw huis....

O-SERVO – HOBBY SERVO MOTOREN



Op onze BBA kunnen we standaard 2 servomotoren aansluiten.



Hobby Servomotortjes zijn ideaal om een bepaalde relatief nauwkeurige, krachtige maar beperkte roterende beweging te veroorzaken. Onze servomotortjes kunnen ongeveer 120° verdraaien. De servomotortjes die wij gebruiken komen uit de modelbouwwereld en worden daar gebruikt om allerlei op afstand bestuurd modellen te besturen. Zo worden ze gebruikt in de stuurinrichting van wagens, in de hoogte- en rol-roeren van vliegtuigen en in helikopters.

Er bestaan 2 verschillende formaten, de mini en normale servomotoren en die zijn leverbaar in allerlei verschillende versies en kwaliteiten. Daarnaast bestaan er ook nog aangepaste servomotoren die continu kunnen blijven ronddraaien. Deze aanpassingen kan je ook zelf doen aan bestaande servomotoren a.d.h.v. de vele handleidingen die je hierover op het internet kan vinden.

Mini servo	Normal servo	Continue rotatie servo
Conrad: 6.99€ China: 1.12€	Conrad: 10.00€ China: 4.00€	Conrad: - China: 5.00€

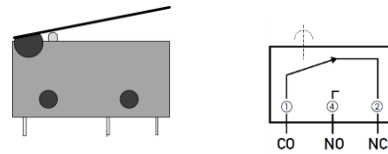
Dit programma beweegt de as van een servomotor op pin D11 traag van 0° naar 180° en daarna nog trager terug van 180° naar 0°



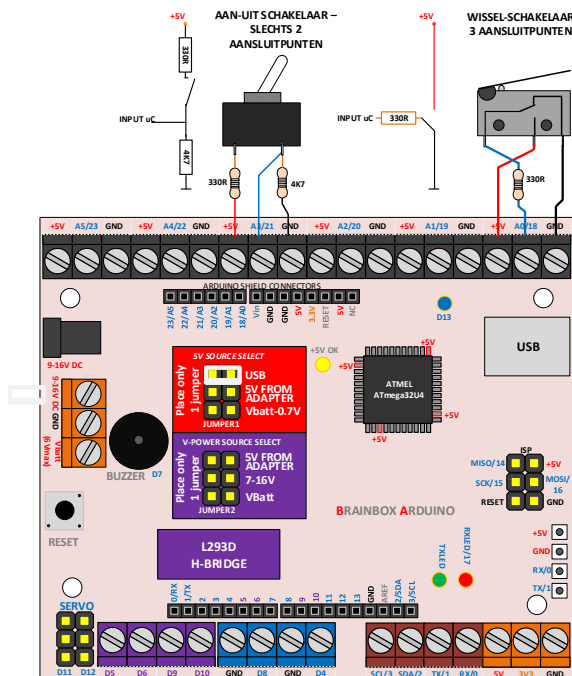
DIGITALE INPUTS – WAAR OF NIET WAAR

I-DIG SCHAKELAAR OF DRUKKNOP

5. Raadpleeg de datasheet of gebruik je multimeter om de schakelaar of drukknop door te meten. Wij hebben hier gekozen voor een eindeloopschakelaar.

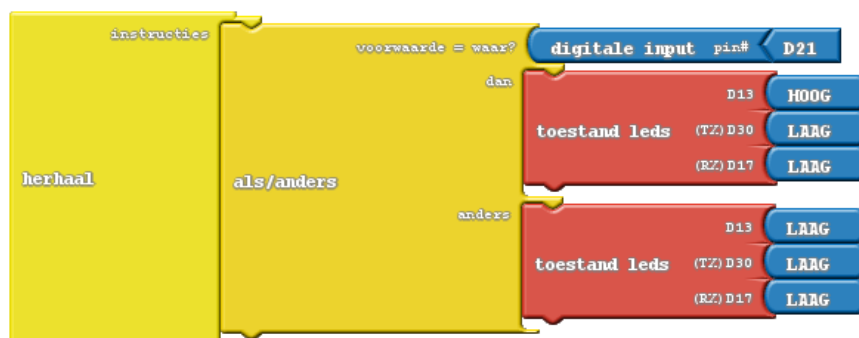


6. Schakelaars en drukknoppen worden best op onderstaande manier aangesloten om de ingangen van de microcontroller te beveiligen wanneer de pin in de software foutief als output zou geschakeld worden.
7. Sluit de schakelaar of drukknop aan – aan één van de digitale IO-pins.



Schakelaars kunnen enkel 1 of 0 doorgeven – aan of uit, waar of niet waar. In Ardublock hebben dit soort inputs een typische ronde vorm – zoals te zien is op “digitale pin #” hieronder.

Dit programma ‘kijkt’ of de schakelaar aan pin D21 is ingedrukt of niet. Als deze ingedrukt is, zal de led aan pin D13 branden.



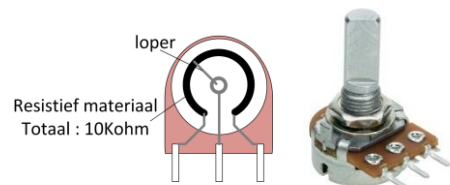
ANALOGUE INPUTS – MEETWAARDE 0-1023

Analoge sensoren meten een waarde tussen 0 en 1023. Deze waarde is recht evenredig met de gemeten spanning op die pin. Enkel de pins met een aanduiding Ax kunnen analoge spanningen meten.

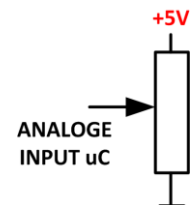
I-AN POTENTIOMETER

Potentiometers zijn ideaal om een bepaalde waarde in te stellen. Bijvoorbeeld een gewenste temperatuur of toerental. Je zou er ook een bepaalde positionering of hoekverdraaiing mee kunnen meten.

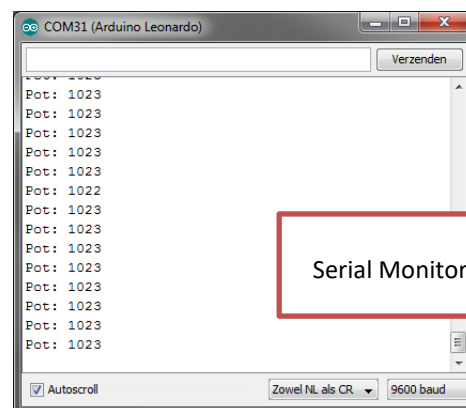
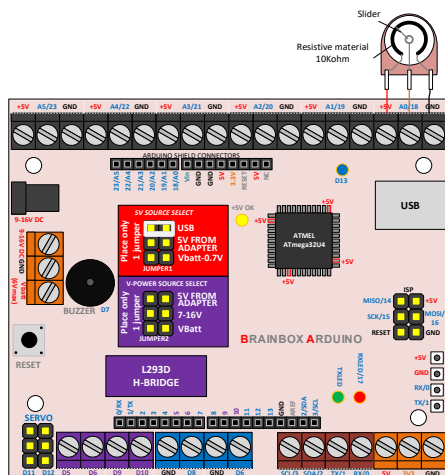
- Potentiometers zijn regelbare weerstanden en bestaan uit een baan van resistief materiaal. In ons geval meet je tussen de twee buitenste klemmen 10KOhm. De middelste pin is verbonden aan de looper. Door de looper rond te draaien kan je de weerstand tussen de middelste en één van de buitenste pinnen vergroten en verkleinen.



- Door de twee uiterste pinnen van de potmeter te verbinden tussen 5V en GND, wordt de spanning op de middelste pin van de potmeter nu regelbaar tussen 0 en 5V.

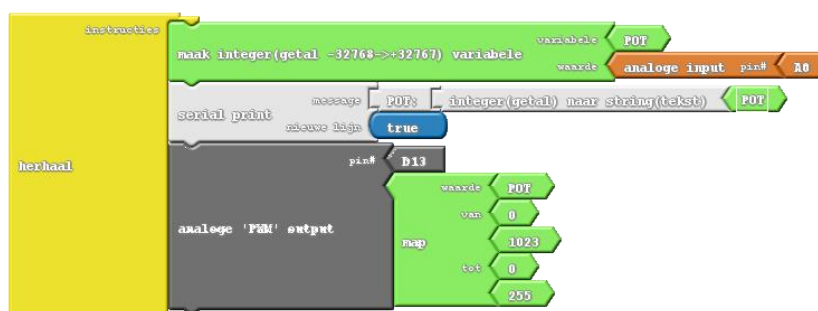


- Sluit de potentiometer aan – aan één van de 6 analoge ingangen (A0, A1, A2, A3, A4, A5)



Dit programma leest een analoge waarde in van de potmeter – dit is steeds een waarde tussen 0 en 1023. Deze waarde wordt in de variabele 'Pot' gezet.

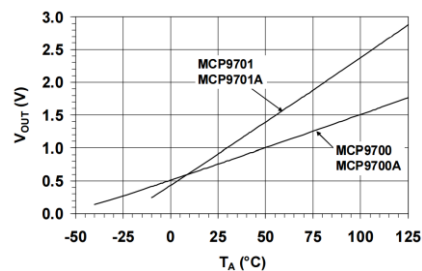
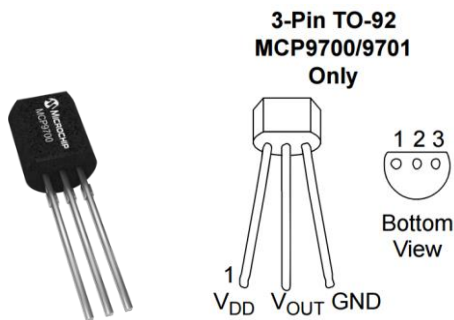
Vervolgens wordt deze waarde verstuurd – via de USB kabel naar de computer. Met 'Serial Monitor' kan je deze waarde dan volgen op het scherm.



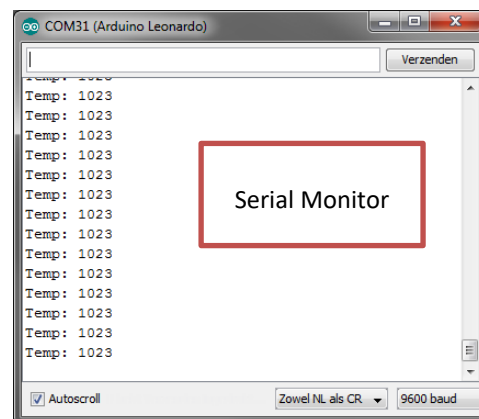
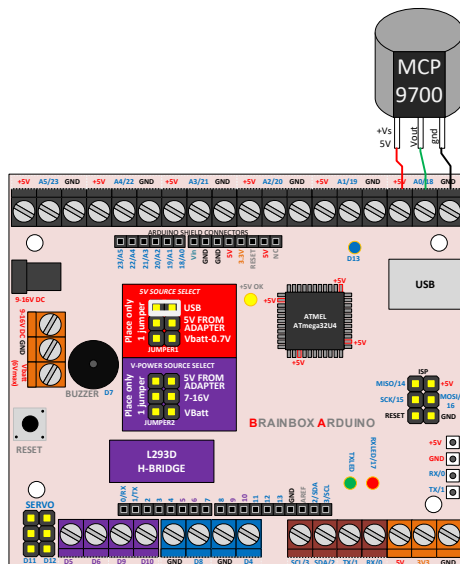
Als laatste dimmen we met deze waarde de led op pin D13. Analoge pincode werkt enkel met waarden tussen 0 en 255, dus we moeten de waarde tussen 0 en 1023 eerst nog herschalen of mappen.

I-AN TEMPERATUURMETING MET MCP9700

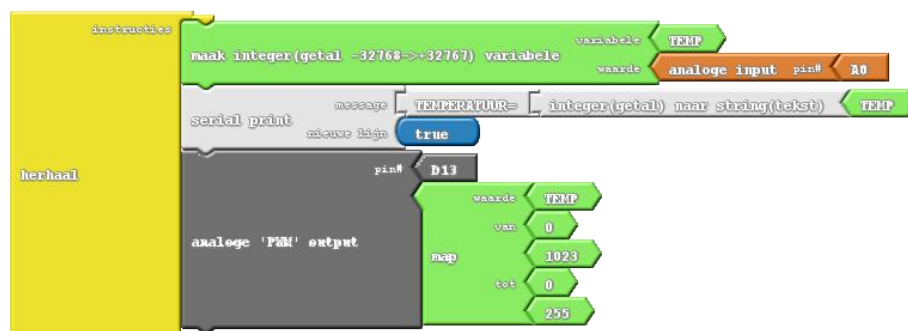
De MCP9700 is een goedkope lineaire temperatuursensor die rechtstreeks aan een microcontroller aan te sluiten is. Deze sensor is geschikt om zonder extra kalibratie temperaturen te meten tussen -40°C en +125°C. **!!Raadpleeg de datasheet van de MCP9700 voor meer info.**



De MCP9700 geeft een spanning uit die met 10mV per graad Celsius stijgt. Bij 0°C is dat 0.5 Volt. Bij 100°C is dat 1.5V. Sluit de MCP9700 aan – aan één van de 6 analoge ingangen (A0, A1, A2, A3, A4, A5)



Dit programma leest een analoge waarde in van de temperatuursensor – dit is steeds een waarde tussen 0 en 1023. Deze waarde wordt in de variabele 'Temp' gezet.



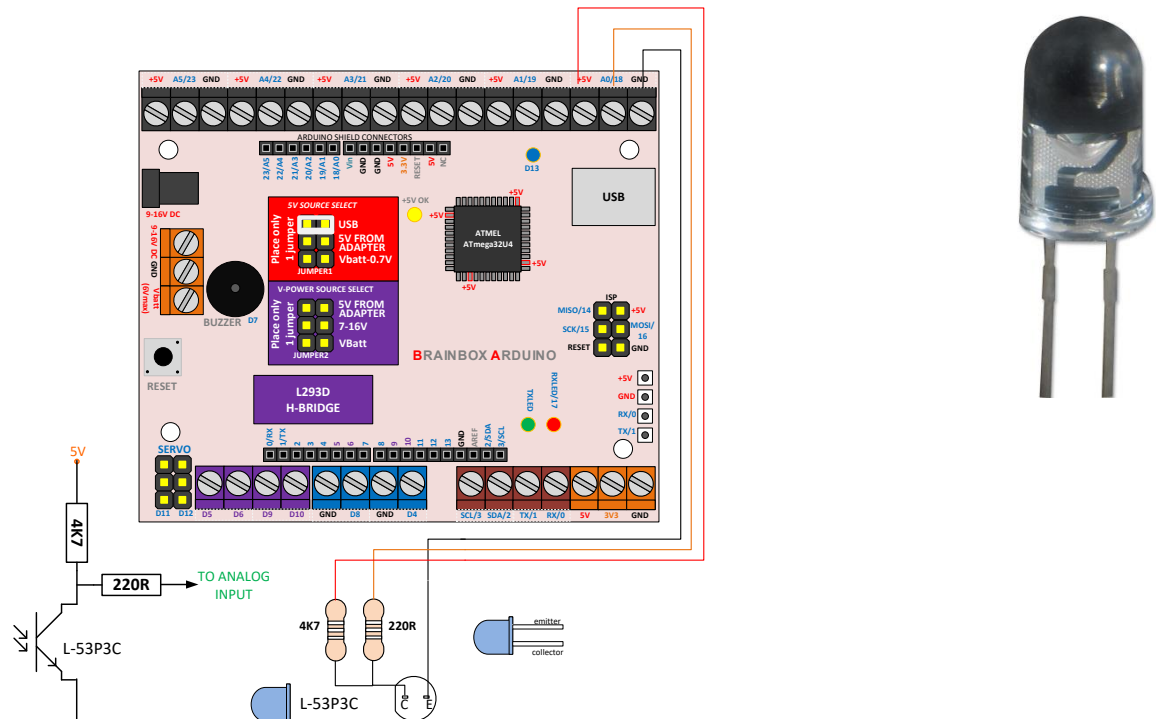
Vervolgens wordt deze waarde verstuurd – via de USB kabel naar de computer. Met 'Serial Monitor' kan je deze waarde dan volgen op het scherm.

Als laatste dimmen

we met deze waarde de led op pin D13. Analoge pincode werkt enkel met waarden tussen 0 en 255, dus we moeten de waarde tussen 0 en 1023 eerst nog herschalen of mappen.

I-AN LICHTMETING MET FOTOTRANSISTOR

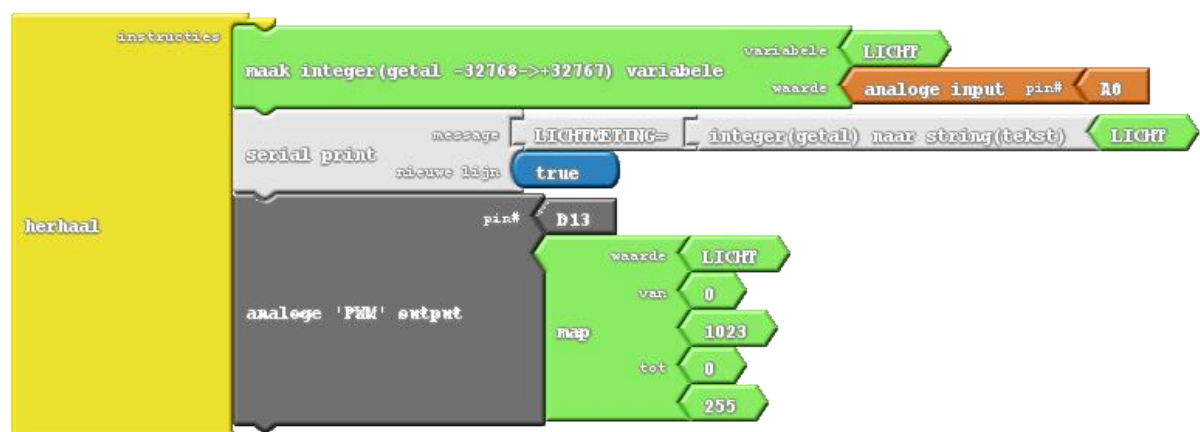
Hoe meer licht invalt op de fototransistor, hoe meer de transistor zal geleiden en hoe minder spanning er over de transistor blijft staan.



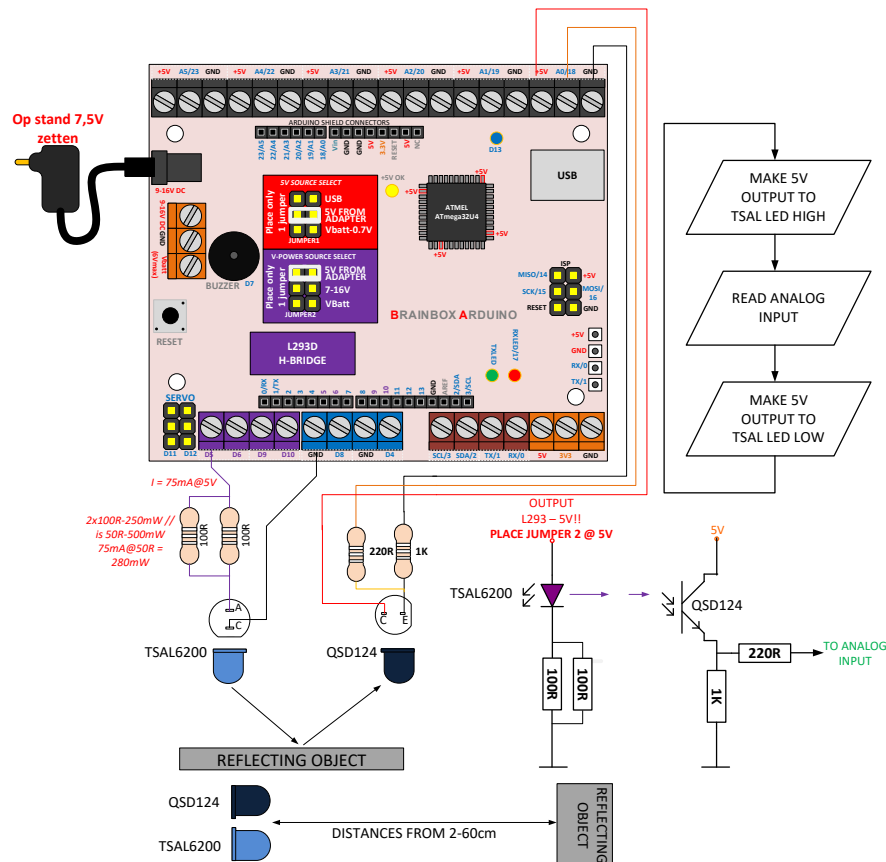
Dit programma leest een analoge waarde in van de lichtsensoren – dit is steeds een waarde tussen 0 en 1023. Deze waarde wordt in de variabele 'Licht' gezet.

Vervolgens wordt deze waarde verstuurd – via de USB kabel naar de computer. Met 'Serial Monitor' kan je deze waarde dan volgen op het scherm.

Als laatste dimmen we met deze waarde de led op pin D13. Analoge pincode werkt enkel met waarden tussen 0 en 255, dus we moeten de waarde tussen 0 en 1023 eerst nog herschalen of mappen.



I-AN AFSTANDMETING MET INFRAROOD LICHT



Deze afstandmeter werkt met een zender en een ontvanger. Een uitgezonden infrarode lichtstraal weerkaatst op een voorwerp. Hoe dichterbij dit voorwerp, hoe meer licht er weerkaatst en hoe meer licht er gemeten wordt in de ontvanger.

Componenten:

IR Emitter TSAL6200 (alt for TSAL5100)	Farnell: 3152856
IR Photo Transistor QSD124 (alt for BPV11F)	Farnell: 2453253
4 resistors – 250mW	100R, 100R, 1K, 220R

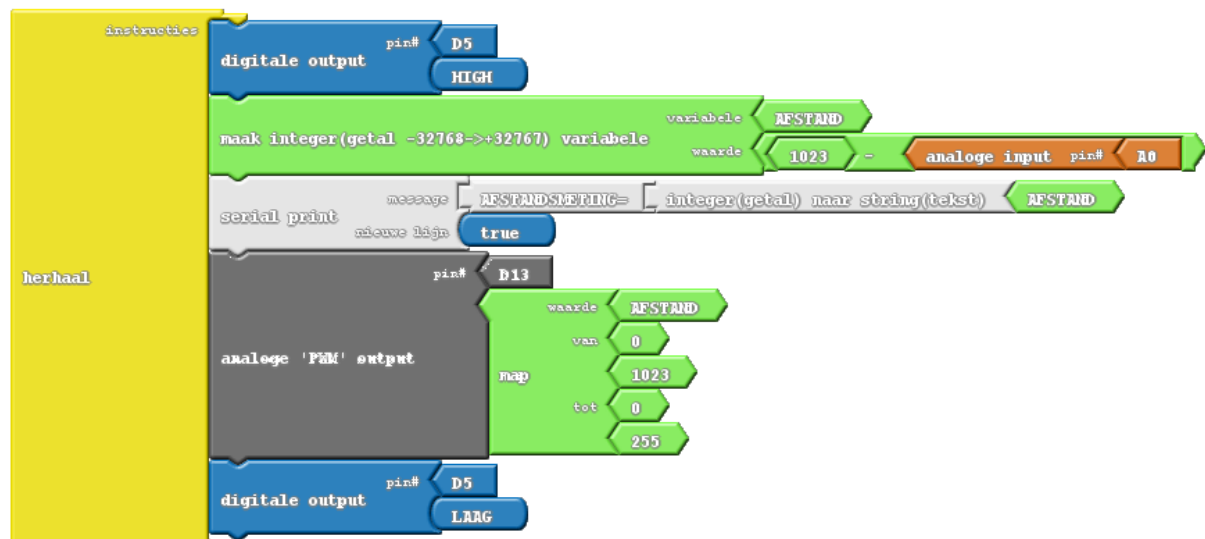
Als zender gebruiken we een TSAL56200 IR emitter van Vishay. We werken hier met IR licht omdat dat niet zo sterk wordt gestoord door omgevingslicht. Enkel zonlicht bevat een stevige IR component die deze sensor zou kunnen storen. De TSAL6200 schakelen we via één van de uitgangspinnen van de L293D H-brug omdat de stroom door de TSAL in de buurt van de 75mA ligt en dit te hoog is voor een gewone 20mA uitgang. Door deze TSAL met een uitgang te sturen kunnen we meerdere TSAL's in of uitschakelen zonder dat ze elkaar storen. Let er op dat we de TSAL6200 hier sturen met een 5V signaal – de 'VPOWER SOURCE SELECT'-jumper van de Brainbox Arduino moet hier dus op 5V staan. De H-brug werkt niet op USB power!

Als ontvanger gebruiken we een QSD124 Fototransistor. Deze sensor heeft een extra filter die enkel IR licht binnen laat. Deze sensor is een fototransistor die meer in geleiding gaat als er mee licht op invalt. De spanning over de weerstand die we binnenlezen in onze uC is dus omgekeerd evenredig met de afstand.

Kortere afstand -> meer IR licht weerkaatst -> Transistor meer in geleiding -> Spanning over R(1K) stijgt.

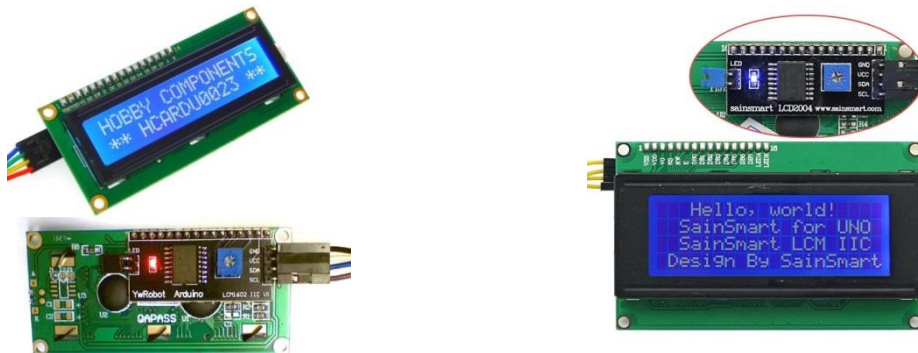
In het programma hieronder doolopen we een aantal stappen:

- We maken powerpin D5 hoog zodat de infrarode led zal beginnen stralen. Dit licht is niet te zien met het blote oog, maar wel met de meeste camera's van mobiele telefoons.
- De IR fototransistor is aangesloten op A0, die analoge waarde lezen we binnen in variabele 'AFSTAND', maar we inverteren die (grote afstanden gaven immers een kleine waarde terug en omgekeerd)
- Deze waarde sturen we naar de PC, zodat die met 'serial monitor' kan gevisualiseerd worden
- Met deze waarde dimmen we ook de led aan pinD13 – geschaald van 0-1023 naar 0-255
- Tot slot zetten we de zendled terug uit na deze meting omdat die veel stroom trekt.

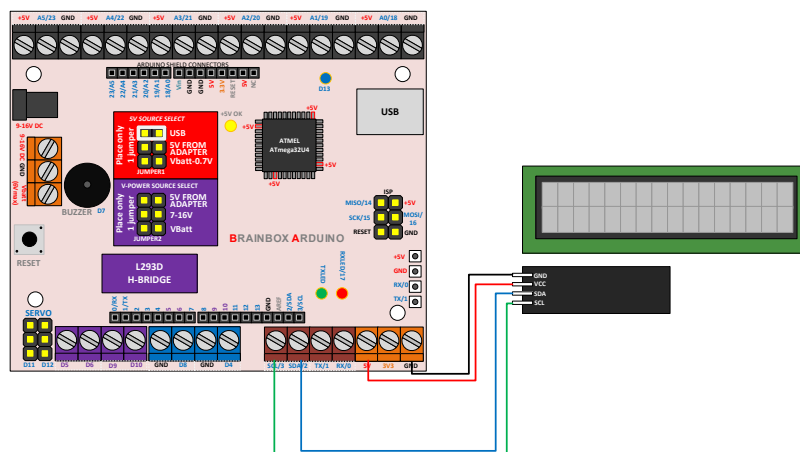


I2C COMMUNICATIE

I2C LCD



Sinds enkele jaren zijn er een aantal heel goedkope LCD's op de (Chineese-) markt. Deze LCD's hebben een extra printje achteraan dat I2C signalen omzet naar LCD signalen. I2C is een protocol dat veel gebruikt wordt om sensoren en actuatoren aan te sluiten met slechts 4 draden: GND, 5V, SCL, SDA. Sluit deze LCD correct aan zoals hieronder aangegeven.



LIBRARY INSTALLEREN

Om deze I2C LCD's te kunnen aansturen onder Ardublock / Arduino IDE moet je eerst de bijpassende library downloaden en installeren onder IDE.

- 1- download de "LiquidCrystal_I2C" library van <https://bitbucket.org/fmalpartida/new-liquidcrystal/downloads>
- 2- V1.3.4 is getest - unzippen is niet nodig
- 3- in Arduino IDE: Sketch >> include library >> add .ZIP library – selecteer de library zip file
- 4- Deze library is nu geïnstalleerd onder 'mydocs' -> Arduino : je kan deze hier ook verwijderen

I2C ADRESSEN

De meest voorkomende I2C adressen: 0x27, 0x20, 0x3F

Indien deze niet werken, dan kan je een I2C adres scanner gebruiken:

<http://playground.arduino.cc/Main/I2cScanner>

In onderstaand programma worden achtereenvolgens volgende stappen gezet:

- LCD Light on – zet de blauwe backlight aan – op I2C adres 0x27
- Set Integer: lees analoge waarde binnen van pin A0 en zet deze in var: "x"
- LCD CLEAR: Wis alle tekst op de LCD
- 16by2: Stuur de tekst 'var x=' naar de LCD en kleef hier de waarde van var 'x' bij aan, zet deze regel op lijn 1, karakter 1 – dus links bovenaan de LCD. Het I2C adres van deze LCD is 0x27
- Wacht 300msec om een duidelijk beeld te krijgen zonder flikkering.

