

## ROBOBITS<sup>-85</sup>

### VAN DE BESTUURSTAFEL

Beste lezer,

Er is weer voldoende gebeurd deze zomer. In navolging van de Summer Tech Day 2018 in Hengelo is een aantal van ons op bezoek geweest bij de HCC!3D in Apeldoorn tijdens de Zwitsal Summer Tech Day op 3 augustus 2019.

Ik hoop dat hieruit meer samenwerking met andere HCC interesse groepen ontstaat. Het volgende evenement waar we dit proberen is tijdens de HCC!kennisdag op 12 oktober in Houten. Hier gaan we een demonstratie geven van de RoboRama wedstrijd, dus kom vooral met je Robot naar Houten. Kun je alvast oefenen voor de RoboRama in november.

Verder heeft Joep tot onze spijt aangegeven dat hij stopt als penningmeester. Wel zal hij actief blijven binnen onze interessegroep. We danken Joep voor zijn inspanningen en heb na wat navragen Ed Buzzi bereid gevonden om deze taak binnen het bestuur in te vullen. Hij gaat samen met Joep de begroting voor het komende jaar opstellen zodat de kennis wordt over gedragen.

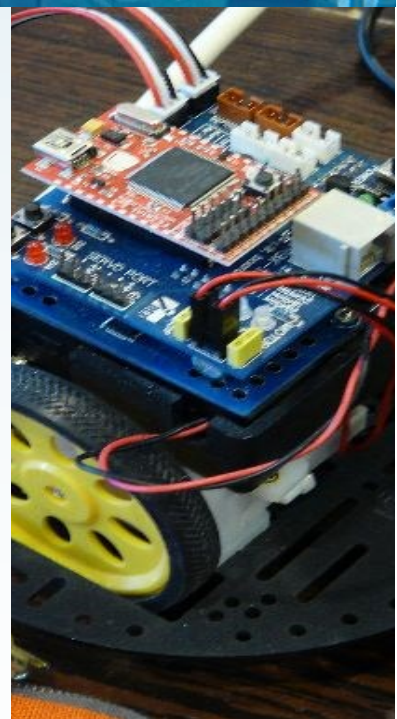
Samen met Edith heb ik in september de kast van de HCC!robotica maar eens open gemaakt om te kijken wat er nog bruikbaar is van de zaken die er in liggen. Uit de nalatenschap van Hinnie zijn er nog wat robots die we in de toekomst weer nieuw leven in willen blazen. Als je interesse hebt om hier bij te helpen laat het ons weten. We kunnen deze robots dan gaan gebruiken bij demonstraties.

Ook zijn er nog wat bakken met onderdelen. Deze gaan we de komende maanden op tafel zetten zodat we deze kunnen gebruiken. In tegenstelling tot de "is gratis" spullen willen we de onderdelen als voorraad van de club gebruiken.

In deze uitgave van onze Robobits wordt stilgestaan bij de apollo 11 maanlanding en de esp32-cam robot van Abraham. Als je een beetje kunt schrijven of een leuk idee hebt om een artikel te maken voor in de Robobits, laat het ons weten! De Robobits ontleent zijn bestaansrecht aan artikelen voor en door leden.

Met vriendelijke groet,

Wim de Boer.

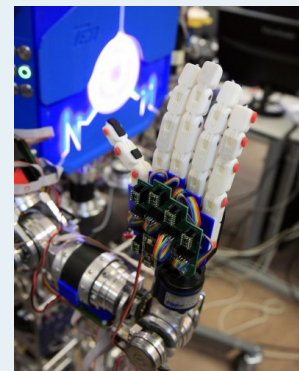


### IN DIT NUMMER

Van de bestuurstafel.....	1
The Apollo 11 maanlanding.....	2
Maanlander programma.....	3
Aan de slag met de ESP32-Cam.....	4
HCC Agenda.....	8

### robot-gebarentaal

Onderzoekers van het Robotica Lab van de UC3M (Madrid) hebben een humanoïde robot geprogrammeerd om in gebarentaal te communiceren.



# The Apollo 11 maanlanding

Richard J. Gran and Ossi Saarela, MathWorks

Waarschijnlijk is de aandacht in de media voor deze historische gebeurtenis, 50 jaar geleden, niemand ontgaan. In een recent artikel van Richard J. Gran and Ossi Saarela, MathWorks, wordt ingegaan op de maanlanding op 20 juli 1969 waarbij 'auto pilot software' is toegepast.



*Met is bijna niet voor te stellen met welke middelen men toen moest werken en wat we nu, 50 jaar later, tot onze beschikking hebben. Onze hobby bestond niet eens! Om te laten zien hoe verweerd we zijn met de mogelijkheden voor de bouw van onze robots (soms voorzien van meerdere Raspberry's, Arduino's en andere krachtige micro-controllers) heb ik een korte samenvatting gemaakt van het artikel. Ter afsluiting heb ik nog wat code toegevoegd van een 'maanlander spelletje', herschreven in Python. Ik zie al mogelijkheden en krijg inspiratie om mijn recent ontworpen drone te voorzien van deze auto pilot!*



*... Ik zie al mogelijkheden en krijg inspiratie om mijn recent ontworpen drone te voorzien van deze auto pilot! .....*

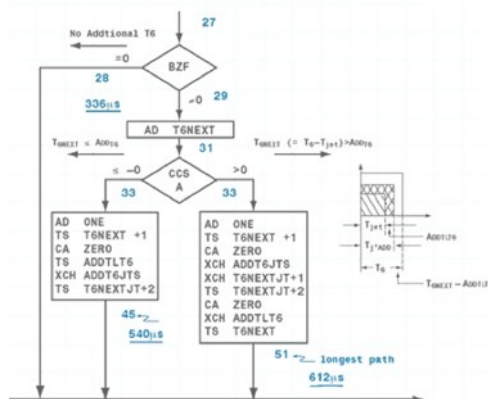


Richard J. Gran was een van de ontwerpers van de software voor de auto pilot die is toegepast bij de maanlander, waarbij voor het eerst de berekeningen gebaseerd werden op natuurkundige modellen volgens wetten van Newton ( $S=v.t + \frac{1}{2}.a.t^2$ ). Regeltheorieën en publicaties over model ondersteund regelen bestonden nog nauwelijks.

Het oorspronkelijke ontwerp van de auto pilot van de maanlander was een analoge variant die werd gebruikt om de stuur-/remraketjes van de lander te bedienen. Vanwege de digitale beperkingen van die tijd zoals rekensnelheid en opslag capaciteit werd besloten de digitale versie van de auto pilot als back-up systeem te gebruiken. De toegepaste computer had eveneens maar één interrupt en er was geen digitaal naar analog interface.

George Cherry, een van de ingenieurs van de bouwer van de maanlander (MIT) wist de NASA te overtuigen dat de rotatie bewegingen van een voorwerp in de ruimte redelijk nauwkeurig te voorspellen zijn. Dit zou betekenen dat de te ontwerpen digitale besturing kon worden ontworpen met een relatief lage bemonsteringsfrequentie (sample rate) en dus alsnog uitgevoerd zou kunnen worden met de bestaande boordcomputer van de maanlander. Als optimalisatie criterium was gekozen voor het minimaliseren van de gemiddelde tijd en de hoeveelheid brandstof.

Onderstaand flowchart uit 1966 laat een stukje algoritme zien die voor de keuze van een van de 16 remraketjes zorgt. Geschreven in assembly. De ontwikkeling hiervan heeft circa een jaar geduurd.

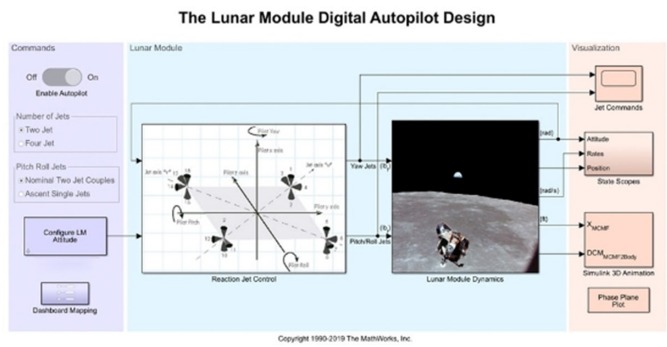


# The Apollo 11 maanlanding (vervolg)

Remraketjes zorgen voor de beweging en besturing van de maanlander. >>

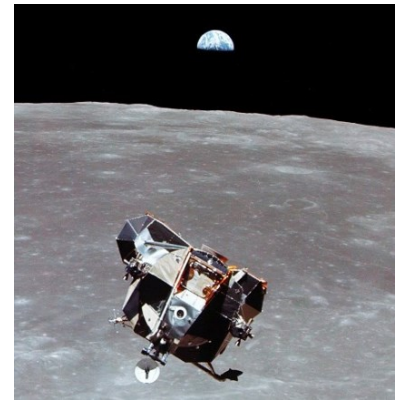
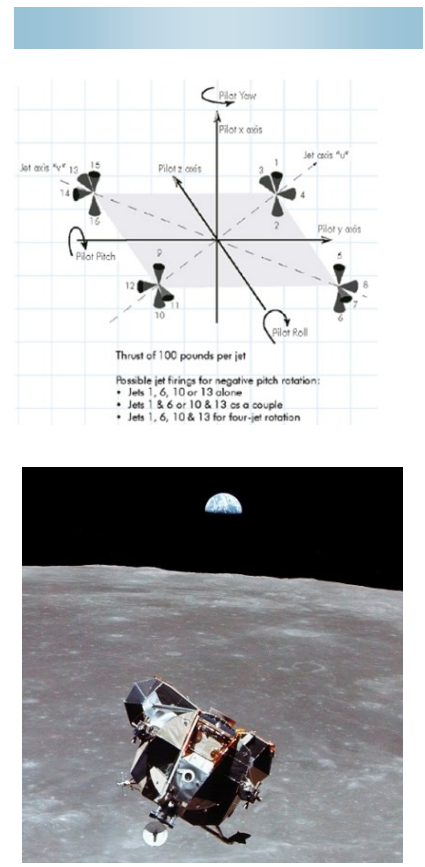
Inmiddels heeft Richard Gran de mogelijkheid gehad om, bijna 50 jaar later, dezelfde auto pilot te programmeren in [Matlab Simulink](#), echter nu binnen een uur denk- en programmeerwerk!

Via deze [studentenversie](#) is het mogelijk om de Mathlab software zelf te gebruiken.



Het volledige artikel kun je lezen via deze [link](#).

Zeno Otten



# Maanlander

Een andere routine om een zachte landing op maan uit te voeren.

Tijdens een opruimactie vond ik een aantal oude boeken waarin computerspellen staan beschreven. De spellen zijn geschreven in 'Basic'. Een van de spellen beschrijft een Maanlander programma. Ik dacht meteen aan het bewerken van het programma en het geschikt te maken voor een van mijn zelf gebouwde drones! Weer een uitdaging?!

Het programma heb ik vertaald naar Python zodat ik het straks kan gebruiken in een microcontroller (bijvoorbeeld een raspberry). De routine laat de piloot een Lunar Module op de maan zetten. De routine gebruikt als basis de bekende kogelbaan vergelijking  $S = SO + v.t + \frac{1}{2}.a.t^2$  (middelbare school, natuurkunde les). Er is geen luchtweerstand.

Er is geen grafische weergave van de resultaten. Uitsluitend worden de numerieke gegevens over de daling naar het maanoppervlak weergegeven na elke nieuwe invoer. Als invoer vraagt het programma een tijdsinterval dt (in seconden), de richting van de lander (in graden) en de stuwdruk (in %). De uitvoer van het programma geeft de hoogte (y) en toont de rest van de variabelen.

De liefhebber kan de routine verbeteren en uitbreiden met een userinterface. Alle parameters en variabelen staan in het twee dimensionale array 'flight'. Ook de berekende resultaten komen in het array te staan.

Het landen op de maan is een lastige klus om te klaren. Wat ontbreekt in deze versie van het programma is de optimalisatie routine, die de richting en stuwdruk als functie van de hoogte (y) vaststelt voor een zachte landing. Wie heeft er een idee !??

Zeno



'Basic computerspellen'  
M. Th. Vijftigschild,  
ISBN 90 201 1601 0

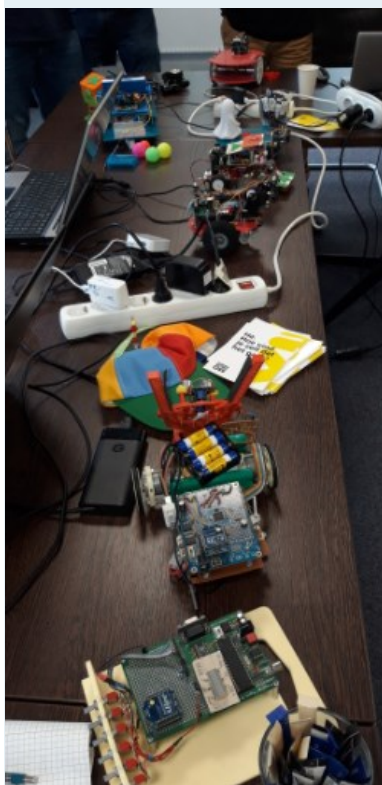
[Het volledige programma 'maanlander.py' kun je downloaden van onze website via deze link](#).

## Zwitsal Summer Tech Day

Zaterdag 3 augustus 2019  
10.00 tot 16.00 uur, in de  
Energiefabriek, gebouw 144 op  
het Zwitsalterrein in Apeldoorn.

Bouw je eigen Gearbug met hcc13D4Kits  
Programmeer een robot. Leer vliegen met een drone  
Maak en print je eigen 3D ontwerp. Laat een full-  
body scan van jezelf maken. Of volg de  
presentaties: Robotica, drones en programmeren.

Ook de HCC Robotica was vertegenwoordigd:



Batterijen, Lipo's, electronica, microcontrollers,  
robots, computers, programma's, goede ideeën,  
adviezen en vooral veel gezelligheid op de  
Zwitsal Summer Tech Day in Apeldoorn.



4

## Aan de slag met de ESP32-Cam



*Binnen de Robotica club worden er heel veel verschillende robots  
gebouwd en hierbij worden veel verschillende processoren ge-  
bruikt. Van oudsher zijn dat de Atmel-AVR en Microchip-PIC  
chip's. Daarna zijn er een heel scala aan andere processoren  
bijgekomen, we denken aan de ST, Propeller, ARM en ga zo  
maar door.*

*Een aantal jaar terug is de ESP8266-01 module op de markt gekomen. De kracht van  
deze module is dat het een complete Wifi toepassing is, met hierop een seriele aanslui-  
ting. Door middel van ATcommando's is deze module te gebruiken om een willekeurig  
ucontrollerbord van Wifi te voorzien. Op zich een mooie toepassing, maar "onder de  
motorkap" van deze module schuilt een 32-bits processor met 1 – 4 MB geheugen en  
dat voor maar een paar Euro!!*

*Hier dromen we van als we over microcontrollers praten, en we gebruiken het "alleen  
maar" voor communicatie. Het grote nadeel van de ESP8266 serie was dat het pro-  
grammeren hiervan niet heel makkelijk ging. Maar voor de IOT toepassingen is het  
heel grote doorbraak geweest. We zijn nu jaren verder, de ontwikkelingen hebben  
niet stil gestaan. Verschillende ontwikkel platforms hebben de ESP-familie omarmd. Het  
bedrijf Espressif heeft een opvolger uitgebracht, de ESP32.*

*Tijd om de zaak opnieuw te evalueren.*

### De ESP32.

*De ESP32 bevat een Xtensa dual-core processor  
met een kloksnelheid van 160 of 240 Mhz, met  
512 KiB SRAM, Wifi en Bluetooth en de nodige IO  
aan boord. Dit is best wel indrukwekkend en dat  
nog steeds voor maar een paar Euro. In de afgelo-  
pen paar jaar zijn er rond de ESP heel erg veel  
verschillende bordjes en module's op de markt  
gekomen. Met hierop 4 – 8 MB geheugen, meer  
en minder IO-pinnen. Verschillende vormen en  
standaarden in bordjes. Op basis van de Arduino-  
layout of de Wemos D1, met of zonder display en  
vrij recent met een cmos camera module erop.  
Dat is de ESP32-CAM.*

*Deze module heeft een 2 Mpixel cameramodule,  
de OV2640-camera, 512 KiB SRAM en 4 MB extra  
geheugen erop. En dit alles voor rond de 6,50 Euro !!!*

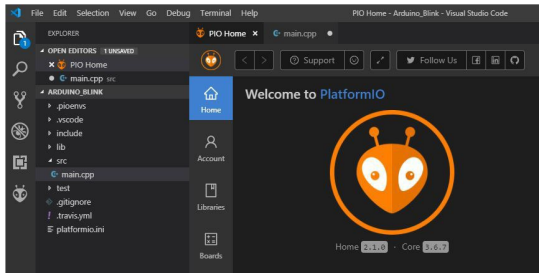


### Ontwikkelomgeving.

*Van oudsher is er voor de ESP8266 en later de ESP32 door het ontwikkelbedrijf Esper-  
riffs een eigen ontwikkelomgeving op de markt neer gezet.  
De ESP-SDK. [www.espressif.com](http://www.espressif.com). Een hele tool-chain met alle toeters en bellen erom-  
heen.*

*Voor de "professionele" ontwikkelaar geen probleem maar voor de hobbyist voelt het  
een beetje als met een kanon op een mug schieten. Daarna zijn er door allerlei verschil-  
lende partijen verschillende ontwikkelomgevingen opgezet.*

Een heel mooie is op basis van Visual Studio met hieroverheen PlatformIO.



Zie <https://fasani.de/taq/esp32/> voor een opzet voor dit systeem.

Dit systeem is nog volop in ontwikkeling en is heel mooi, alleen zelf liep ik zo nu en dan tegen het "probleem" aan dat ik een foutmelding kreeg, die verdween nadat ik PlatformIO opnieuw opgestart had. Dat was best wel even zoeken naar fouten die er niet altijd bleken te zijn. Daarnaast zijn de ESP8266 en ESP32 ook in de Arduino-IDE geïmplementeerd. Dit betekent dat we in de voor veel mensen vertrouwde Arduino-omgeving een ESP32 bordje en library's kunnen kiezen en dat we hiermee gewoon aan de slag kunnen.

Dan wordt de "moeilijke" ESP32 processor opeens voor veel mensen 'heel makkelijk' bereikbaar.

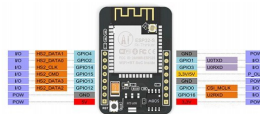
#### Hardware ESP32-CAM.

De ESP32-CAM module wordt standaard met een normale lens uitgevoerd. Deze lens heeft een beeldhoek van ongeveer 60 graden. Willen we aan beeldherkenning gaan doen op de wat kortere afstand, lijnvolgen bijvoorbeeld, dan moeten we de module op een vrij grote afstand van de vloer monteren.

Er is ook een Fish-Eye module beschikbaar. Deze heeft een beeldhoek van 120 graden. In dat geval kan de afstand tot de vloer korter worden. Er zijn 2 nadelen/verschillen met de normale camera module. Het beeld is 90 graden gedraaid en de randen van het beeld verlopen van kleur als we de onboard LED aanzetten om het beeld te verlichten. Deze randen worden minder belicht en verlopen hierdoor meer naar grijs.



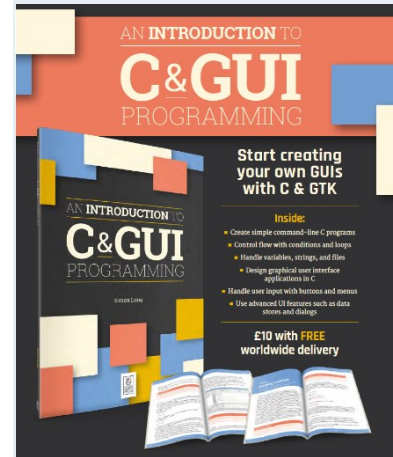
De Esp32-CAM heeft 16 aansluitingen, 3 daarvan zijn GND en er is 1 5V aansluiting die we voor de voeding gebruiken. De 3,3 V aansluiting gebruiken we niet. Ook de uitgaande 3,3/5V aansluiting gebruiken we niet.



Voor het programmeren gebruiken we een seriële poort op GPIO1 en GPIO3. En om de module in de programmeermodus te zetten moeten we de GPIO0 aan GND leggen en deze pin kunnen we dan ook (beter) niet gebruiken voor andere toepassingen. GPIO4 is verbonden met de onboard LED. De GPIO12/13 zijn verbonden met de interne SPI aansluiting naar ondermeer de PSRAM en als we deze "verkeerd" gebruiken start de ESP32 module niet op. Het verbinden van een LED aan de 5V via een weerstand naar GPIO12 en 13 levert deze fout op. Zo dunt het wel lekker uit met vrije pinnen. We zouden kunnen uitzoeken of we deze GPIO-pinnen op een andere manier kunnen gebruiken, maar dat heb ik niet gedaan. Dan houden we GPIO-pinnen 2, 14, 15 en 16 over. In theorie kunnen we tot 14 mA sinken met een GPIO-pin. Willen we de ESP32-CAM programmeren dan moeten we een USB-Serieel omzetten op de GPIO1 (TxD) en GPIO3 (RxD) aansluiten. De module heeft GEEN USB aansluiting aan boord.



#### BOEKEN:



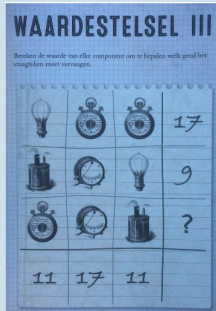
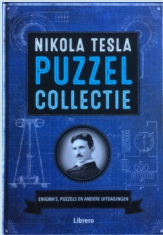
In the first half of the book, An Introduction to C & GUI Programming, we learn the basics of the C language, and see how to use it to write text-only programs to run at the command line. Looking at the keywords and library functions covered, it isn't obvious how to create everything required for a full GUI – desktop, windows, icons, mouse pointer, and so on – with them.



20 makkelijke en creatieve elektronica-projecten voor kinderen en volwassenen!

## BOEK

Nikola Tesla wordt gezien als een van de grootste ingenieurs en uitvinders aller tijden.



Nog een keer de maanlanding.  
One Giant Leap or Lunar Landing Navigation

Via deze link kun je lezen wat de plannen zijn om in 2024 te gaan landen op de maan en met welke techniek de landing gaat plaats vinden.



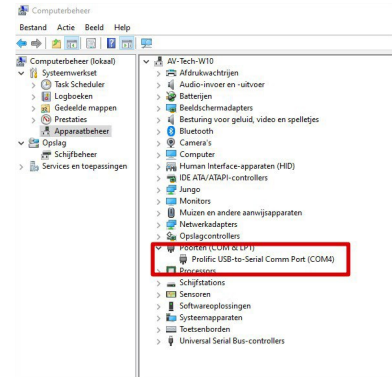
6

## Aan de slag met de Arduino-IDE.

Als we de normale Arduino-IDE geïnstalleerd hebben gaan we naar bestand-voorkeuren en dan bijna onderaan - Meer Board Manager URL's-. Rechts staat een blokje waarmee we een invul scherm kunnen openen waar we aanvullende URL's in kunnen vullen waar vandaan we andere borden kunnen installeren.



Bij [Hulpmiddelen --> Board --> Board beheren] vullen we bij Filter [ESP32] in en we krijgen de libraries die we nodig hebben. Na installatie van de borden gaan we de instellingen voor de ESP32-CAM goed zetten. Bij [Hulpmiddelen --> Board:] kiezen we -ESP32 Wrover Module - Hierna kunnen we voor deze module de volgende instellingen goed zetten:  
Upload Speed: "921600"  
Flash Frequency: "80 Mhz"  
Flash Mode: "QIO"  
Partition Scheme: "Huge APP (3MB No OTA)"  
Core Debug Level: "Geen"



Gaan we naar Voorbeelden --> ESP32 --> Camera --> CameraWebServer, dan kunnen we het voorbeeld programma installeren waarmee we onze ESP32-CAM kunnen testen. In deze sketch moeten we de SSID en wachtwoord van een Wifi netwerk invullen en hiermee verbindt de ESP32-CAM zich met het lokale netwerk en kan je aan de hand van een IP-scan programma of de uitlezing van de seriële debugger zien op welk IP-adres het beeld van de camera te zien is.

We kunnen nu in de webinterface allerlei zaken doen en bekijken. Dit is heel leuk. Een beperking in de eigen programma's is dat deze (nog) niet goed samenwerken met de webinterface in de ESP32. Gebruiken we de webinterface en gaan hier beelden streamen dan hindert dat de werking van het eigen programma. Mijn beperkte programmeer skills op het gebied van HTML en C weerhouden mij om dit op (korte) termijn op te lossen. In eerste instantie vind ik dat geen probleem. Het zou leuk zijn als hier een oplossing voor komt, maar dat staat bij mij niet bovenaan mijn lijstje van -To Do-

### Eigen programma's.

Werkt dit dan kunnen we naar stap-2, next-level, ons eigen programma en hieruit het beeld halen en in onze eigen toepassing gaan gebruiken. Op de GitHub pagina <https://github.com/AVTech1/ESP32-Cam-Walker> staan de volgende drie programma's waar we naar gaan kijken, te weten:

ESP32\_beeld.ino, ESP32\_lijnsensor.ino en WoF\_CAM.ino .

ESP32\_beeld.ino stuurt een "beeld" wat origineel 160 \* 120 Pixels is in een formaat van 32 \* 15 karakters naar de seriële poort. Deze karakters laten zien hoe zwart/grijs/wit een pixel is en zo krijgen we een (vaag) beeld van onze opname.

ESP32\_lijnsensor.ino stuurt via de seriële poort 1 lijn als karakters in "grijstinten" of met lijndetectie filter. Deze informatie zou je in een andere microcontroller als lijnsensor-Input kunnen gebruiken.

WoF\_CAM.ino is een sketch waarin 2 servo's aangestuurd worden die 4 poten van een Robot vormen.

We behandelen de sketch ESP32\_beeld.ino, hierin gaan we het principe behandelen.

De sketch bestaat uit verschillende delen, we beginnen met de [void setup()]. Hierin worden alle dingen opgestart en ingesteld die maar 1 keer voorkomen. Hierin komen we het blok [camera\_config\_t config;] tegen.

Hierin staan heel veel toewijzingen en instellingen van de OV2640-camera module. Speciaal zien we de instellingen [config.pixel\_format = PIXFORMAT\_GRAYSCALE] en [config.frame\_size = FRAMESIZE\_QQVGA].

We willen eenvoudige data uit onze afbeelding halen en dat doen we als we er een Grayscale afbeelding van maken in het QQVGA formaat van 160 \* 120 pixels. Aan het einde van de setup configureren we de onboard-LED door middel van het commando: [ledcSetup(3, 2000, 8);] en [ledcAttachPin(Led0, 3);].

We wijzen de pinnummer Led0 toe aan een willekeurig nummer -3-.

Met ledcSetup stellen we deze pin in op 2000 Hz met een 8 bits PWM uitgang. Hierna kunnen we de betreffende uitgang met het commando [ledcWrite(3,100);] op 100 van de 255 delen "8-bit's intensiteit" aanzetten.

In de [void loop()] roepen we elke halve seconde de routine [camera\_beeld();] aan. Hierin zetten we met het commando: [camera\_fb\_t \* s\_state = esp\_camera\_fb\_get();] een beeld in het frame-buffer. Dit frame-buffer moet je gewoon als een blok geheugen zien van 160 \* 120 byte's met 8-bit's grijsstinten. Je kunt het ook zien als een rij van 19.200 (= 160 \* 120) byte's.

Met dit geheugen cq rij byte's kunnen we nu zelf dingen gaan doen. In [uint8\_t pixel = (s\_state->buf[iw + (ih \* s\_state->width \* 2)]);] zijn ih en iw de hoogte en breedte in het beeld-frame waarvan we de pixelgrijswaarde in de variabele [pixel] zetten.

Met de variabele [pixel] kunnen we vervolgens eigen zaken gaan doen. In ons voorbeeld gaan we het 160 \* 120 pixel bestand omzetten in een 32 \* 15 pixel/karakter beeld en sturen dit als ascii-karakters naar de seriële poort.

De andere 2 sketchen zijn uitbreidingen op de sketch ESP32\_beeld.ino en hebben verder geen uitleg nodig. In WoF\_CAM.ino worden 2 servo's aangestuurd en hiervoor staan de benodigde commentaren in de sketch.

### Opmerkingen.

Als we de FishEye camera module gebruiken dan moeten we beseffen dat we best wel breed beeld hebben als we vanaf 20 cm hoogte meten. Een lijn van 1 cm wordt dan heel smal en kunnen we bijna niet meer zien. Dan is het het overwegen waard om veel meer metingen te verrichten en dan bijvoorbeeld per 3 of 5 pixels een gemiddelde te nemen. Dan halen we veel

### Conclusie.

De ESP32-CAM is een heel leuke module voor een nog leukere prijs. In de Arduino-IDE is hij heel goed te programmeren. Het "nadeel" zijn de beperkt aantal IO-pinnen, maar dat is eventueel met I2C wel te omzeilen. Omdat ik een Walker wilde bouwen met 2 servo's was het beperkte aantal IO-pinnen voor mij geen probleem. Ik heb veel plezier beleefd aan deze ontdekkingsstocht en hoop dat anderen aan de ESP32-CAM ook veel plezier beleven.

Abraham Vreugdenhil.

Programma's en informatie om met de ESP32-Cam aan de slag te gaan

11 commits			1 branch			0 releases			1 contributor		
Branches	master	New pull request	Find File	Clone or download							
AVTech1	Create ESP32_Aan_de_slag_met.pdf										Latest commit f414dc on 9 Aug
ESP32_beeld	beeldverwerking aangepast										last month
ESP32_ljnsensor	beeldverwerking aangepast										last month
WoF_CAM	Walker met ESP32-CAM										last month
ESP32_Aan_de_slag_met.pdf	Create ESP32_Aan_de_slag_met.pdf										last month
README.md	Update README.md										last month
walk.jpg	WoF										last month

## WALKER

Abraham toont zijn esp32 cam robot tijdens de maandelijkse bijeenkomst in Hooglanderveen.

Tussen de andere robots is de esp32 met zijn eigenzinnige poten (WALKER) op de voorgrond te zien .

Misschien bij de komende RoboRama als lijnvolger te bewonderen.



```
#if defined(CAMERA_MODEL_WROVER_KIT)
#define PWDN_GPIO_NUM    -1
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    21
#define SIOD_GPIO_NUM    26
#define SIOC_GPIO_NUM    27

#define Y9_GPIO_NUM      35
#define Y8_GPIO_NUM      34
#define Y7_GPIO_NUM      39
#define Y6_GPIO_NUM      36
#define Y5_GPIO_NUM      19
#define Y4_GPIO_NUM      18
#define Y3_GPIO_NUM      5
#define Y2_GPIO_NUM      4
#define VSYNC_GPIO_NUM   25
#define HREF_GPIO_NUM    23
#define PCLK_GPIO_NUM    22

#elif defined(CAMERA_MODEL_ESP_EYE)
#define PWDN_GPIO_NUM    -1
#define RESET_GPIO_NUM  -1
#define XCLK_GPIO_NUM    4
#define SIOD_GPIO_NUM    18
#define SIOC_GPIO_NUM    23

#define Y9_GPIO_NUM      36
#define Y8_GPIO_NUM      37
```

## HCC!Robotica ig

HCC-Robotica is een interessegroep die zich bezig houdt met het ontwikkelen, ontwerpen, programmeren en bouwen van elektronica en mechatronica, toegepast op robots. Deze meer of minder intelligente en autonome robots en machines met verschillende sensoren, actuatoren, processoren en bewegende onderdelen worden onder andere ingezet bij de jaarlijkse georganiseerde Roborama wedstrijden. Wij komen elke eerste zaterdag van de maand bijeen in dorps huis de Dissel te Hooglanderveen. Kennis delen, kennis vergaren, presentaties en workshops bijwonen zijn terugkerende activiteiten tijdens deze bijeenkomsten.

U bent van harte welkom!



## Agenda

### Zaterdag 12 oktober HCC!kennisdag in Expo Houten

Een groot aantal interessegroepen laten u zien wat zij zoal te bieden hebben. Maak kennis met 3D, Apple, Drones, SeniorenAcademie, Flightsimulator, Genealogie, Modelbaanautomatisering, FotoVideo, Hardware Support, Open Source, Programmeren, Robotica, eCs, CompUsers, Commodore, Treinsimulator, Artificiële Intelligentie (AI) en Windows. Robotica gaat ter voorbereiding van Roborama die plaatsvindt in november een oefenwedstrijd houden. U vindt de Expo aan Meidoornkade 24, 3992 AE Houten. Er is ruimschoots gratis parkeergelegenheid en de bus stopt voor de deur.

De HCC!kennisdag is geopend van 10.00 tot 15.00 uur. De toegang is voor alle bezoekers gratis.

### Zaterdag 2 november De jaarlijkse Roborama Nederland wedstrijd

Op de eerste zaterdag van november vindt de Roborama wedstrijd plaats. Op de website van [Robotica](#) kun je de reglementen voor deelname opzoeken. Dit jaar zal er voor het eerst ook ruimte zijn voor diegenen die zelf aan de slag zijn gegaan met de robot Maqueen Micro:bit. Deze robot is uitgebreid besproken tijdens het seizoensthema in de regio's en in de PCActive. Alle bezoekers zijn welkom, toegang is gratis.

Zaterdag 2 november 10:00-16:00  
De Dissel, Disselplein 6, Hooglanderveen

### Discussiegroepen HCCROBOTICA:

[http://groups.google.nl/group/hcc\\_robotmc](http://groups.google.nl/group/hcc_robotmc)

### Blogs

<http://zotten.wordpress.com/>

<https://avretro.wordpress.com/>  
<http://www.robotblog.nl/>

### Facebook

**HCC!Robotica ook op Facebook.**

Gewoon om te laten weten, dat wij ook op Facebook actief zijn.

## Teensy 4.0

- 600 MHz ARM Cortex-M7
- 2048K Flash, 1024K RAM
- Chip NXP iMXRT1062
- 2 USB ports
- 3 CAN Bus
- 32 General DMA Channels



[Meer informatie op de website van Antratek](#)

## HCC!Robotica ig

### Dagelijks bestuur:

Voorzitter : Wim de Boer  
Secretaris : Edith van Putten  
Penningmeester : Joep Suijs

### Het Kernledenbestand ziet er als volgt uit en zal het dagelijks bestuur ondersteunen:

Redactie : Zeno Otten  
Website : Bert Berrevoets  
Techniek : Tim Woldring  
Roborama : Bert Ruben  
Public Relations : Rien van Harmelen  
Externe Contacten : Ed Buzzi

Website: <http://www.hccrobotica.nl>