

—● A Poor Man's Smart Doorbell - deel 1 ●—

André Reinink

Of toch maar ouderwets belletje trekken?



Bron: <http://saltooo.be>

Waarom?

Waarom zou je een dure 'smart' deurbel kopen? Omdat je voor het gemak gaat? Of omdat je graag wilt weten wie er aan jouw (voor)deur staat? Je hebt geen zin in een hobbyproject? Geen zin om zelf te knutselen? Bovenstaande kan allemaal kloppen! Maar misschien ben je best wel nieuwsgierig en wil je ontdekken of het ook anders kan. Misschien wel goedkoper. Misschien wel leuker. Misschien wel met meer privacy. Dan is het de moeite waard om het artikel waarin je nu begonnen bent te lezen, toch maar helemaal te lezen.

Hoe kom ik op het idee?

Bij mooi weer zit ik vaak in de tuin. Je kent het wel: buiten schijnt het zonnetje en de temperatuur is ook aangenaam. Een krantje en een kopje koffie er bij. Vrienden en bekenden weten dat we een tuin hebben. En als de voordeur niet open gedaan wordt, komen ze wel achterom. Maar soms belt er iemand aan en dan hoor ik de bel van de voordeur niet. Dan hoor ik later: 'Ik was bij je aan de deur, maar er werd niet open gedaan'. En ook de postbode wil wel eens een pakje afleveren. En dergelijke lieden hebben steeds minder tijd en steeds meer haast. Tijd is geld. Dus als je niet binnen 10 seconden open doet, krijg je een dreigmail waarin staat dat de buurman jouw pakje heeft aangenomen. En laat dat nou net de buurman zijn die een hekel heeft aan anderen's pakjes. Daar moet toch een oplossing voor te vinden zijn.

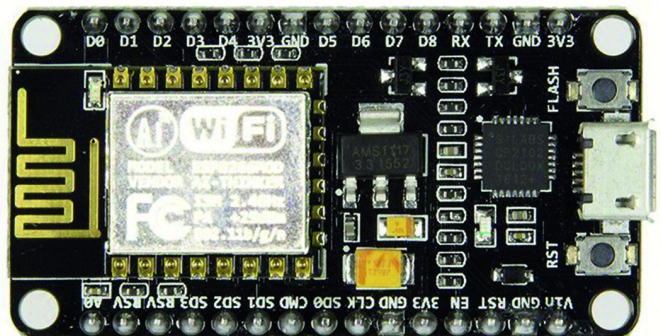
En wat als...

iemand aanbelt en je per omgaande een bericht op je mobiele telefoon krijgt? Als dat te regelen is kun je alsnog naar de voordeur lopen. En wat als je dat allemaal met simpele en goedkope middelen zelf kunt realiseren? Oké, er komt ook nog een stukje software aan te pas. En ja, je hebt wifi en internet nodig.

Ik liep al een tijdje met een idee rond om zelf iets in elkaar te flansen en te proberen het idee te realiseren. Nu is dat goed voor je grijze hersencellen, maar ik liep tegen een artikel aan waarin beschreven werd hoe na het indrukken van een knop een e-mail verstuurd werd. En zo kwam van het een het ander.

Het artikel van Rick Gouin¹

Ik stootte in eerste instantie op een artikel met daarin drukknoppen en een NodeMCU. Een NodeMCU is een veel gebruikt stukje elektronica. Je zou kunnen zeggen dat het een uitgebreide wifi-chip is. Tegenwoordig inclusief Bluetooth. Natuurlijk hoort daar dan ook nog een processor met in- en uitgangen bij. Ten slotte is er een geheugenchip toegevoegd. De chip is in bijna alle gevallen van Espressif. Je kunt kiezen uit een print met voorgesoldeerde aansluitpennen of voor een print zonder voorgesoldeerde aansluitpennen. Dat laatste is van belang als je weinig plek hebt om de print in te bouwen. Alhoewel, zo groot is de print nou ook weer niet. Onlangs vond ik een handjevol in de uitverkoop omdat er weer nieuwere modellen waren. Ik betaalde ongeveer drie euro per stuk voor een module met 4 MB geheugen en een Espressif ESP8266². Voor mijn doel meer dan toereikend.



Voorbeeld van een NodeMCU

De print is, afhankelijk van de fabrikant, tussen de 50 en 60 mm lang en zo'n 30 mm breed. Ook qua aansluitingen van pinnen zijn er varianten. En dat geldt ook voor de USB-aansluiting.

Hoe nu verder?

De print landt vrij snel na het doen van een bestelling³ op de deurmat. Maar voordat ik met het artikel van Rick Gouin aan de slag ga, moet ik nadenken hoe ik het signaal van de deurbel op de print aangesloten krijg (daar had ik natuurlijk al wel iets voor in gedachten). Anders geformuleerd: als er iemand op de knop van de bel drukt, hoe komt dat signaal dan op de ESP-print binnen?

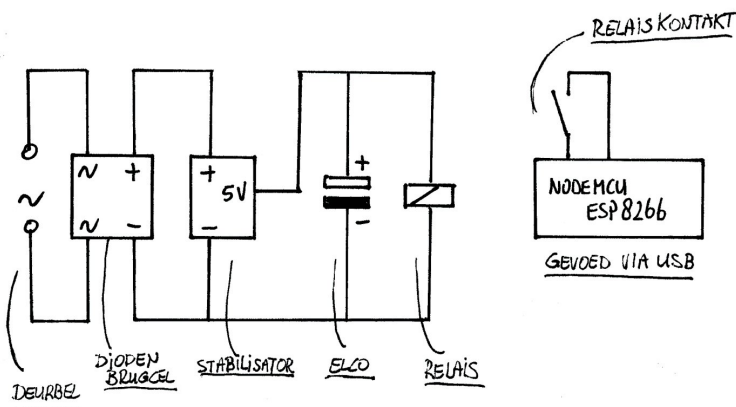
Even een stukje elektronica theorie.

In veel gevallen is er in de woning een zogenaamde beltrafo aanwezig. Een trafo is een veel gebruikte verkorte aanduiding voor transformator. Vlamingen spreken altijd over een 'transfo'. Mooi taaltje dat Vlaams. Op de primaire zijde van de trafo is de voedingsspanning van 230V aangesloten. De drukknop is aan de andere, secundaire zijde van de trafo aangesloten, samen met de drukknop. Op het moment dat je

de drukknop indrukt komt er een spanning te staan op de eigenlijke deurbel: dingdong! Een deurbel heeft veelal een voedingsspanning ergens tussen de 8V en 12V wisselspanning. Elektronica zoals de ESP8266 werkt op een gelijkspanning. We moeten dus het belsein aanpassen en omzetten naar een gelijkspanning. Tot zover de inleidende theorie.

Een opzet voor een circuit.

Links in het schema het wisselspanningssignaal van de deurbel. Het signaal moeten we omzetten van wisselspanning naar gelijkspanning. Dit kun je doen met minimaal één zogenaamde diode, twee is beter. Nog beter, gemakkelijker en uiteindelijk goedkoper is het om gebruik te maken van een diodebrugcel⁴. Een zogenaamde Greatzschakeling.



Van een wisselspanning hebben we nu een pulserende gelijkspanning gemaakt.

Mogelijk voldoende, maar in de elektronica doen we er graag een schepje boven op. Daarom plaatsen we achter de diodebrugcel een zogenaamde stabilisator, een type 7805. Als kers op de taart plakken we er nog een zogenaamde afvlakcondensator (elco) overheen om de laatste rimpels weg te werken. Daarmee creëren we een extra fraaie gelijkspanning van 5V. En die 5V hebben we nodig voor de laatste component: een relais⁵. Relais van 5V zijn goed leverbaar in de plaatselijke elektronikawinkel. De gelijkspanning die de ESP8266 zelf kan leveren is 3.3V. Deze 3.3V is een onhandige spanning om een relais aan te sturen. Bovendien zijn de uitgangen van de ESP8266 te zwak om dat te doen.

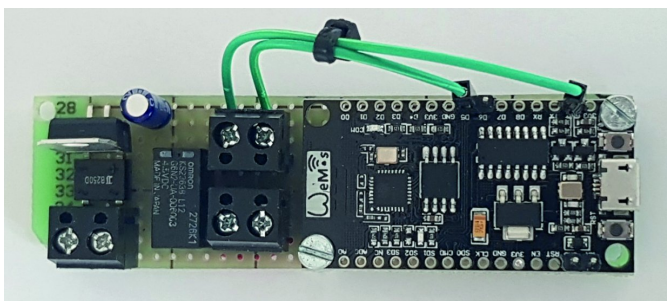
Een relais bestaat uit een spoel en een contact.

Als er een spanning op de spoel gezet wordt, wordt het contact geschakeld. Belangrijkste eigenschap van een relais: het zorgt voor een galvanische scheiding tussen het circuit en de ingang van de ESP8266.

Daarmee lijkt het alsof de drukknop op de ingang van de ESP8266 aangesloten is. Goed, genoeg theorie.

En nu naar de praktijk.

Je kunt alle componenten met draadjes aan elkaar verbinden. Ik ga voor een ietsje fraaiere oplossing. Ik zoek een stukje experimenteerprint en monteer alle componenten op de print.



Experimenteerprint met rechts de ESP8266

De print maak ik iets groter en daardoor heb ik dan de mogelijkheid om de ESP8266 ook op de experimenteerprint te monteren.

Links op de print een aansluitconnector voor de deurbel. In directe nabijheid de Greatzschakeling en de stabilisator, condensator en relais. Direct naast het relais twee connectoren om naar de ESP8266 te brengen. Een van de connectoren is al uitbedraad: twee groene draden zijn aangesloten op de ingangen van de ESP8266. Bij het schrijven van dit artikel stootte ik op nog een interessante mogelijkheid. Daarom heb ik op de print een tweede connector gemonteerd. Ik kom daar later op terug.

De NodeMCU heeft een USB-connector. Deze is in eerste instantie nodig om de print te voeden en daarnaast om via de USB-aansluiting het programma over te zetten.

Rui en Sara.

Het artikel van Rick blijkt een springplank naar een andere website met nog veel meer projecten: Random Nerd Tutorials (RNT)⁶. Rui en Sara Santos zijn de beheerders van een ware schatkamer aan projecten.

RANDOM NERD TUTORIALS

Rui Santos begon Random Nerd Tutorials in 2013 om zijn Arduino-projecten te delen. Hij studeerde toen 'Electrical and Computer Engineering', en op aanraden van een vriend begon hij met een Arduino.

Toentertijd dacht hij dat het een goed idee was om niet alleen zijn Arduino-projecten te delen, maar ook aanwijzingen te geven hoe de lezers zelf aan de slag konden met zijn projecten.

Zijn website met projecten werd door veel bezoekers gewaardeerd. En dat enthousiasmeerde hem om er mee verder te gaan. Het bleef niet bij de Arduino. Ook de Raspberry Pi, de ESP8266, Node-RED, ESP32 en Domotica kwamen aan bod.

ESP32 ESP8266 ESP32-CAM RASPBERRY PI MICROPYTHON ARDUINO

Vandaag de dag zijn er meer dan 300 gratis, duidelijk gedocumenteerde, projecten beschikbaar.

Naast de broncode van de projecten zijn er ook talloze video's te bekijken. Ook hebben Rui en Sara vele e-Books uitgebracht met daarin de een scala aan projecten. Voor zo'n 25 euro kun je dan een compleet e-Book downloaden. Updates van het door jou gekochte e-Book zijn gratis. Sommige e-Books zijn gratis te downloaden na aanmelding met je e-mailadres.

Behalve e-Books met projecten zijn er ook cursussen⁷ te volgen. Deze zijn ook tegen betaling, maar wel voor een, vind ik, schappelijke prijs. Neem eens een kijkje op de website en trek zelf je conclusie.

Terug naar Rick!

Op de website van Rick staat duidelijk beschreven hoe je het een en ander configureert en aansluit.

In plaats van de knoppen die hij in zijn project gebruikt, sluit ik het relais aan dat op mijn print is gemonteerd.

Verder in detail.

Het project is op Arduino-software gebaseerd.

Arduino wordt veel gebruikt voor dergelijke projecten en is goed gedocumenteerd.

Downloads voor verschillende OS'en voor Arduino

Windows	Win 10 and newer, 64 bits
Windows	MSI installer
Windows	ZIP file
Linux	Appimage 64 bits (X86-64)
Linux	ZIP file 64 bits (X86-64)
macOS	Intel 10.14, 'Mojave' or newer, 64 bits
macOS	Apple Silicon, 11: 'Big Sur' or newer, 64 bits

De belangrijkste smaken zijn beschikbaar

Onder Windows is de 2.x versie al een tijdje uit. Als ik in de repository van Zorin (Linux) kijk, zie ik dat men onder Linux iets achter loopt. Mijn versie van de Arduino IDE is momenteel 1.8.19. Maar dat hoeft op zich geen probleem te zijn. Ook prettig: er is een groot aantal gebruikers van de software. En dat betekent dat er via diverse fora een goed en snel antwoord te krijgen is op jouw vragen. *Tip:* als je Arduino geïnstalleerd hebt, kies dan Engels als interfacetaal. Dit is zoveel makkelijker bij het zoeken op fora en het volgen van instructies.

Drivers, Boards en Libraries: Drivers

Na installatie van de Arduino-software zijn er drie belangrijke aandachtspunten. Ten eerste: drivers. Hier zijn Linux-gebruikers duidelijk in het voordeel. De NodeMCU printen (en andere printen ook) zijn uitgerust met een driver IC. Dit 'integrated circuit' zorgt voor de communicatie tussen pc en print. In principe zijn er twee uitvoeringen voor wat betreft de driver IC's: CH340 en CP210X. Onder Linux werken, voor zover mijn ervaring reikt, beide driver IC's. Onder Windows werkt de CH340 meestal niet 'out of the box'. De CP210X zou beter ondersteund worden, naar men zegt. Vergewis jezelf er eerst van dat jouw pc het desbetreffende driver-IC herkent. Een simpele methode is om de print aan te sluiten op jouw systeem terwijl je de hardware-manager van Windows 'open' hebt staan. Als Windows de component herkent na aansluiten en een com-poort tevoorschijn tovert, mag je aannemen dat je geen driver hoeft te installeren. Is dat niet het geval probeer dan het drivertype te vinden en installeer die driver. Voer dan de 'trial and error'-methode nog eens uit.

Drivers, Boards en Libraries: Boards

Ervan uitgaande dat de hardware van de print en jouw computer nu met elkaar 'babbelen', gaan we verder met de print zelf. Omdat er honderden, zo niet duizenden, verschillende boardjes zijn, moeten we Arduino een beetje helpen. Ga daarvoor in Arduino naar 'File' en vervolgens naar 'Preferences'. Daar vind je:

Additional Board Manager URLs:

Hier vul je in:
http://arduino.esp8266.com/stable/package_esp8266com_index.json⁸

Sluit nu het menu met 'OK'.

Als het goed is komen straks alle ESP8266-boardjes in een keuzemenu tevoorschijn. *The proof of the pudding is in the eating*, zeggen de Engelsen.

Ga in het hoofdmenu naar 'Tools'. In het submenu zul je de com-poort terugvinden waarop de print is aangesloten. Soms moet je handmatig de com-poort selecteren.

Ook vind je er het submenu 'Get board info'. Vaak, maar niet altijd, vind je er welk board is aangesloten. Bij niet-originele boardjes, zeg maar no-name Chinese boardjes, werkt dit vaak niet of is de info beperkt.

Onder het submenu 'Board' vind je de 'Boards Manager'. Hier selecteer je je board indien nog niet automatisch gevonden. In mijn Boards Manager vind ik pakweg een 50 verschillende boardjes. Selecteer het juiste boardje. En dat is soms makkelijker gezegd dan gedaan.

Tip: zeer regelmatig komen er updates. Deze komen na korte tijd als melding in beeld na opstarten van Arduino. Het is een kwestie van 'Good Housekeeping' om na opstarten van Arduino even te wachten op een eventuele update en deze vervolgens te installeren. Alles wijst zich verder vanzelf.

Drivers, Boards en Libraries: Libraries

In de Arduino omgeving is veel, heel veel, beschikbaar om een programma of project te ontwikkelen. Voor een heel simpel programma heb je geen extra libraries nodig. En in de meeste boardjes die te koop zijn draait al een programma: Blink. Als je spanning op het boardje zet, bijvoorbeeld via USB, dan zal de status-LED gaan knipperen.

```
// the setup function runs once when
// you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over
// and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);
  delay(1000);
  digitalWrite(LED_BUILTIN, LOW);
  delay(1000);
}
```

Voorbeeld programma Blink

In het hoofdmenu van Arduino vind je nog veel meer voorbeelden onder 'File', 'Examples'.

De broncode die Rick beschikbaar stelt gaat verder dan een simpel programma: we gaan een e-mail versturen. Voor het versturen van e-mail is al een kant en klare library beschikbaar: ESP Mail Client.

Deze library, gemaakt door Mobizt, is simpel te integreren in de Arduino-IDE:

Ga daarvoor in het hoofdmenu naar 'Sketch', vervolgens 'Include Library' en ten slotte 'Manage Libraries'. In het zoekvenster typ je 'esp mail client' in. Als laatste is het dan zaak de ESP Mail Client van Mobizt te selecteren. Na installatie sluit je af met 'Close'.

Tip: ook libraries krijgen zeer regelmatig een update. Deze komen na korte tijd als melding in beeld na opstarten van Arduino. Dus na opstarten van Arduino even wachten op een eventuele update en deze vervolgens installeren.

Aan de slag met de broncode.

Je ziet dat de broncode al voorbereid is op de opvolger van de ESP8266: de ESP32.

Mocht je de ESP32 willen gebruiken dan moet je de Board Manager URL uitbreiden met:

https://dl.espressif.com/dl/package_esp32_index.json⁹

Alle wijzigingen die je voor jouw project moet doorvoeren zijn duidelijk gedocumenteerd op de website van Rick.

Feitelijk komt het erop neer dat je de gegevens voor jouw situatie moet aanpassen. Denk daarbij aan de gegevens van de door jou te gebruiken smtp-server, e-mailadres van de afzender, e-mailadres van de ontvanger en onderwerp en tekst van de e-mail.

Omdat ik bij CompUsers een mail-account heb gebruik ik de gegevens van CUMail.

De broncode compileren en uploaden.

De meest elegante methodiek is om eerst de broncode die je in jouw Arduino-IDE hebt staan te controleren.

Ga daarvoor in het hoofdmenu naar 'Sketch' en vervolgens 'Verify/Compile'. Arduino controleert de syntax en geeft aan of er fouten zijn. En als deze er zijn, geeft Arduino aan waar de fout zit en wat de fout is. Als alles correct is kun je de gecompileerde code naar de ESP8266 versturen. Dit gaat via 'Sketch' en vervolgens 'Upload'.

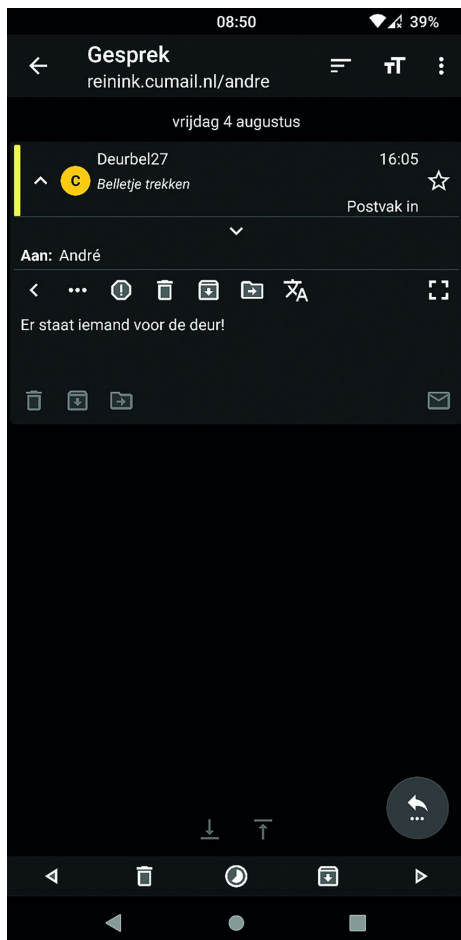
De Arduino-software geeft aan dat het programma correct is overgekomen en gereed is voor gebruik.

Eens kijken of het werkt.

In plaats van de drukknoppen heb ik het relaiscontact gebruikt van de experimenteerprint.

Ik sluit alles aan en druk op mijn eigen voordeurbel. Vreemd gevoel, want dat doe ik eigenlijk nooit...

Na enkele seconden krijg ik een e-mailbericht:



Een screenshot van het e-mail bericht

Dat werkt dus zoals gedacht en verwacht.

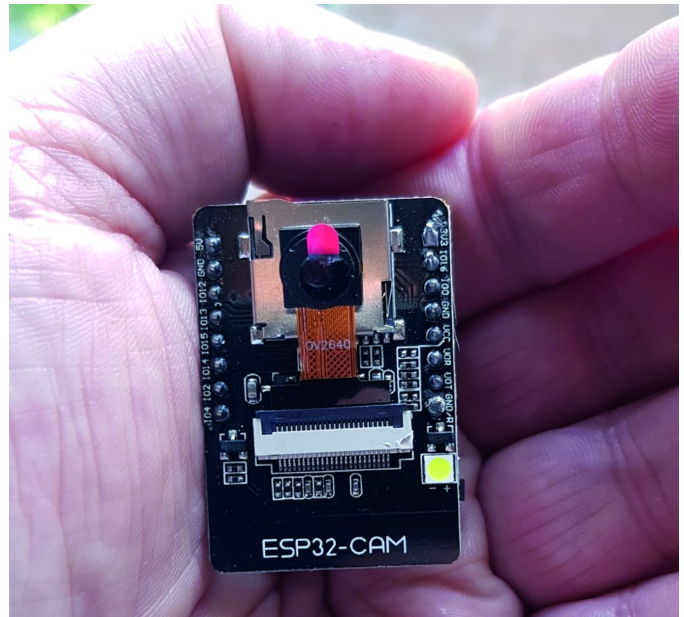
Je bent natuurlijk vrij om gegevens naar eigen smaak aan te passen.

Resumerend.

Met een handvol euro's kun je de componenten aanschaffen voor het beschreven project. Enige handigheid en kennis van elektronica is ook niet verkeerd. Ik geeft het toe, het is misschien niet voor iedereen 'appeltje-eitje'.

In de volgende SoftwareBus.

Terwijl ik aan dit artikel werk stuit ik op een mogelijkheid om dit project uit te breiden. Daarvoor heb ik de tweede connector op de experimenteerprint gereserveerd. Ik hoop in deel 2 hier meer over te schrijven. Alvast een 'teaser':



Links:

1. <https://www.rickgouin.com/build-a-device-to-send-emails-at-the-push-of-a-button/>
<http://bitly.ws/RUqs>
2. <https://www.espressif.com/>
3. <https://www.hobbyelectronica.nl/>
4. <https://nl.wikipedia.org/wiki/Bruggelijkrichter>
<http://bitly.ws/RURc>
5. <https://nl.wikipedia.org/wiki/Relais>
<http://bitly.ws/RURF>
6. <https://randomnerdtutorials.com/>
<http://bitly.ws/RUsc>
7. <https://randomnerdtutorials.com/courses/>
<http://bitly.ws/RUsc>
8. http://arduino.esp8266.com/stable/package_esp8266com_index.json
<http://bitly.ws/RUtd>
9. https://dl.espressif.com/dl/package_esp32_index.json
<http://bitly.ws/RUxk>