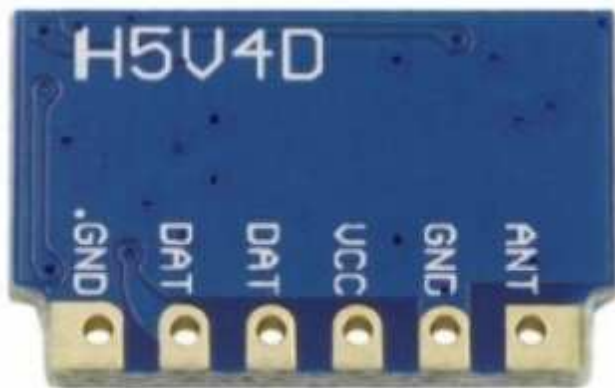


Draadloze besturing met noForth

Er zijn eerste stappen gemaakt voor een domotica werkgroep bij de Forth gg. Lampen en sensoren, etc. kunnen zo vanuit een centraal punt bestuurd worden.

Een van de leuke dingen is dat we in deze onderdelen een zendertje kunnen stoppen en b.v. noForth. Dat geeft eindeloos veel mogelijkheden. We kunnen allerlei verschillende protocollen met elkaar doorverbinden. Een voorbeeld:

- 433 Mhz zend en/of ontvang module
- 2,4 Ghz nRF24L01+ transceiver module
- IR afstand bediening zender en ontvanger (b.v. RC5)
- 2,4 Ghz Wifi modules



433 MHz Receiver



433 MHz Transmitter

Twee experimenten:

- 1) Afstand bediening met 433 Mhz zend- en ontvang-modules.
- 2) Drie nRF24L01 modules, die data aan elkaar doorsturen met een basis idee voor opzetten van een netwerk.

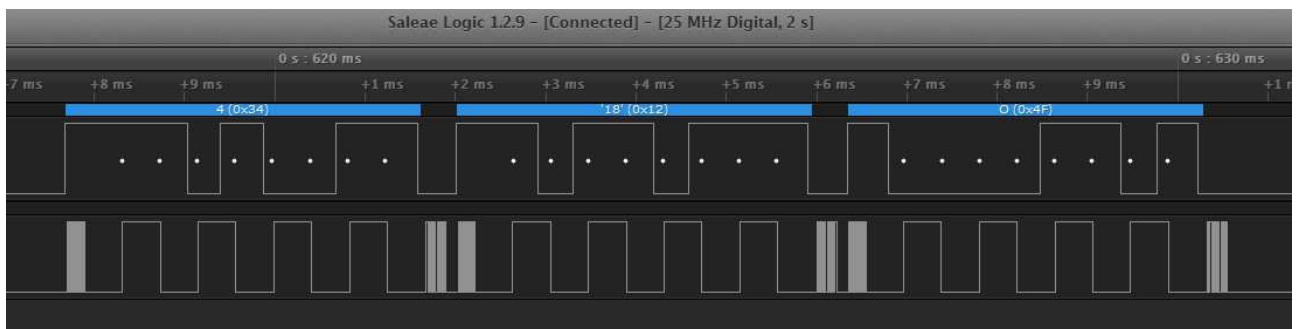
Alle code is geschreven in noForth en voor de experimenten zijn de Egel kit en de Micro Launchpad gebruikt...

1 - Een 433 Mhz ontvanger

Voor dit experiment gebruik ik een oud prototype met een goedkope 433 Mhz zendmodule. De zender krijgt zijn data, de sleutel en commando via een UART aangeleverd.

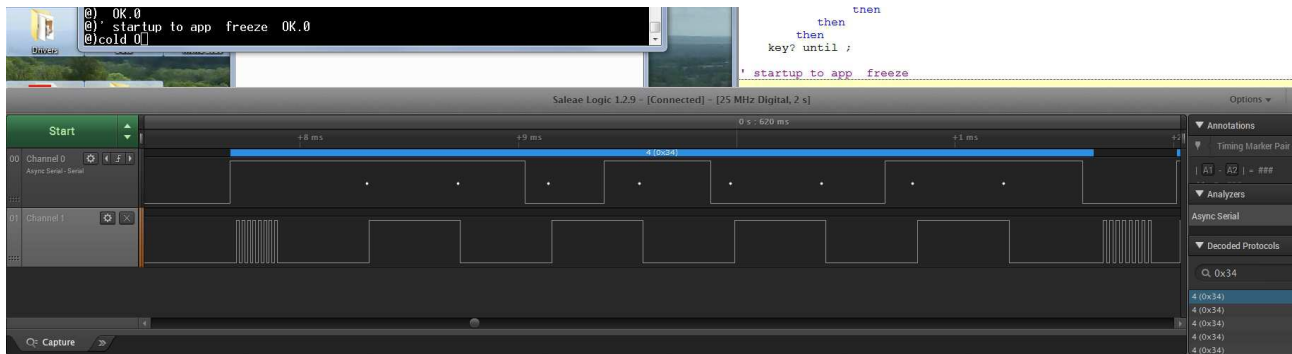
Dit is de ontvanger, hij maakt gebruik van de in machine code geschreven primitieven RS-ON RS-KEY? en RS-KEY om de binnengekomen UART signalen ontrafelen:

```
: CALLER      ( -- )
  rs-on flash
  begin
    rs-key? if      \ Wait for signal
      rs-key 34 = if \ Check key high part
        rs-key 12 = if \ Check key low part
          rs-key ch 0 = if \ Check command
            beep1 \ Command-1
          else
            beep2 \ Command-2
          then
        then
      then
    then
  key? until ;
```

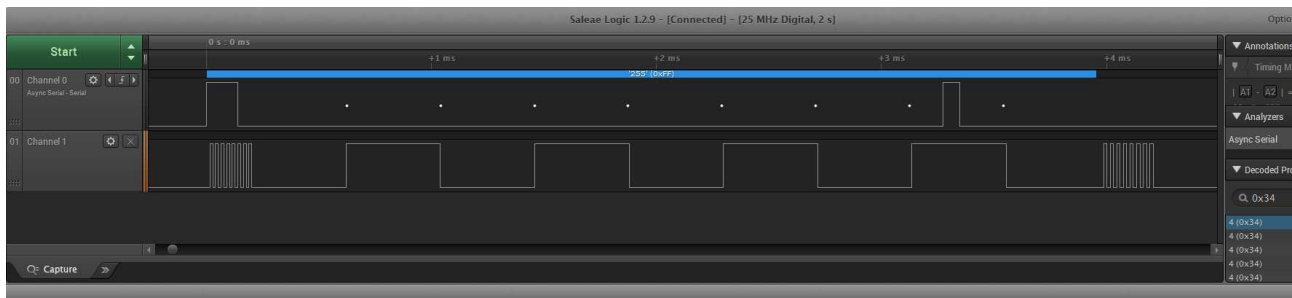


Drie ontvangen bytes (sleutel en commando)

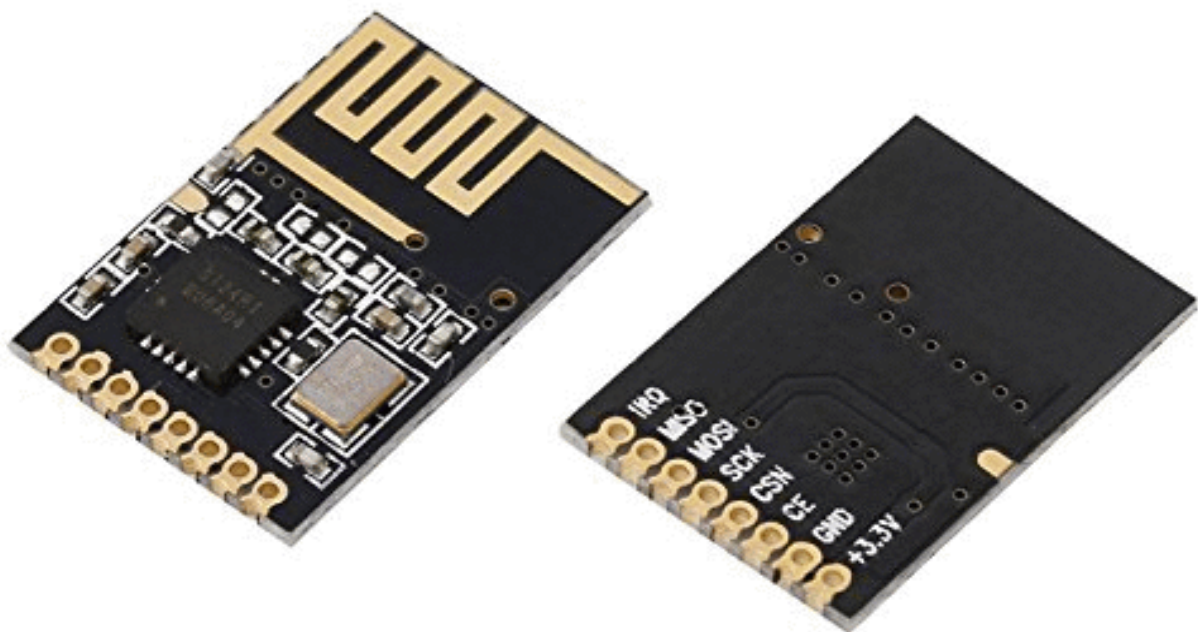
De ontvangen UART data in detail



Een correct ontvangen byte



Een verminkt ontvangen byte



SMD variant van nRF24L01+ module (1,7x18x12)

2 - Een 2,4 Ghz zend/ontvanger experiment

Hier wordt de nRF24L01+ transceiver gebruikt, met als basis de code van Egel hoofdstuk 43. Daarbovenop zijn wat aangepaste woorden gemaakt, om het experiment mee uit te voeren.

Experiment 1, Heeft 3 nodes die data aan elkaar door sturen.
Op de laatste node wordt een led aan/uit gezet.

```
: ONE          ( -- )
  private1
  begin
    s? 0= if
      red-on write-mode ch T xemit \ Send 'T' char
      200 ms red-off              \ to node 2
    then
  key? Until ;

: TWO          ( -- )
  begin
    private2
    read-mode xkey ch T = if      \ Wait for 'T' char
    08 29 *bic                    \ receive mode off
    red-on 200 ms private3 \ retransmit to node 3
    write-mode ch T xemit
    red-off
  then
  key? Until ;

: THREE        ( -- )
  private4 10 2A *bis 10 29 *bic
  begin
    read-mode xkey ch T = if      \ wait for 'T' char
    red-on 200 ms red-off         \ Led on
    08 29 *bic 10 29 *bix        \ receive mode off
  then
  key? until 10 29 *bic ;
```

Experiment 2, Heeft ook drie nodes, maar stuurt de data uiteindelijk terug naar de eerste node. Daar wordt een led aan/uit gezet.

```
: ONE      ( -- )
  private1 10 2A *bis 10 29 *bic
  begin
    s? 0= if
      red-on  write-mode ch T xemit \ Send 'T'char
      200 ms  red-off  10 ms
      read-mode xkey ch T = if      \ Wait for 'T' char
      08 29 *bic 10 29 *bix        \ Activate led
    then
  then
  key? until 10 29 *bic ;
```

Experiment 2b, gaat een stap verder, op elk van de nodes kan iets aan/uit gezet worden. Dit is de eerste stap naar een netwerk met kant-en-klare zend/ontvang modules.

```
: XEMIT?      ( c -- +n )
  0 10 0 do      \ Try it 10 times
    over xemit)   \ Send payload
  tx? if leave then \ Ack?
    reset ch - emit 1+ \ Reset, show & add failures
    flush-tx 70 >channel \ Empty pipeline, clear loss
  loop
  nip reset ;
```

03 constant ME

```
: >NODE      ( node -- )
  >r F0 F0 F0 F0 r@ 0A write-addr \ Receive address P0
  F0 F0 F0 F0 me 0B write-addr   \ Receive address P1
  F0 F0 F0 F0 r> 10 write-addr ; \ Transmit address
```

```

: PRIVATE1      ( -- )
  setup24L01 1 >node      \ Transmit address
  get-status . 10 ms ;    \ Print status & wait

: READ?         ( -- f1 )
  300 0 do
    1 ms rx? if true unloop exit then
  loop false ;

: SW?           ( -- 0|1|2|4 )    \ Test two switches & combine them
  s? 0= 1 and
  1 28 bit* 0= 2 and or
  2 28 bit* 0= 4 and or ;

: COMMAND       ( -- c )          \ Switch to char
  sw? 1 = if ch 1 exit then
  sw? 2 = if ch 2 exit then
  sw? 4 = if ch 3 exit then FF ;

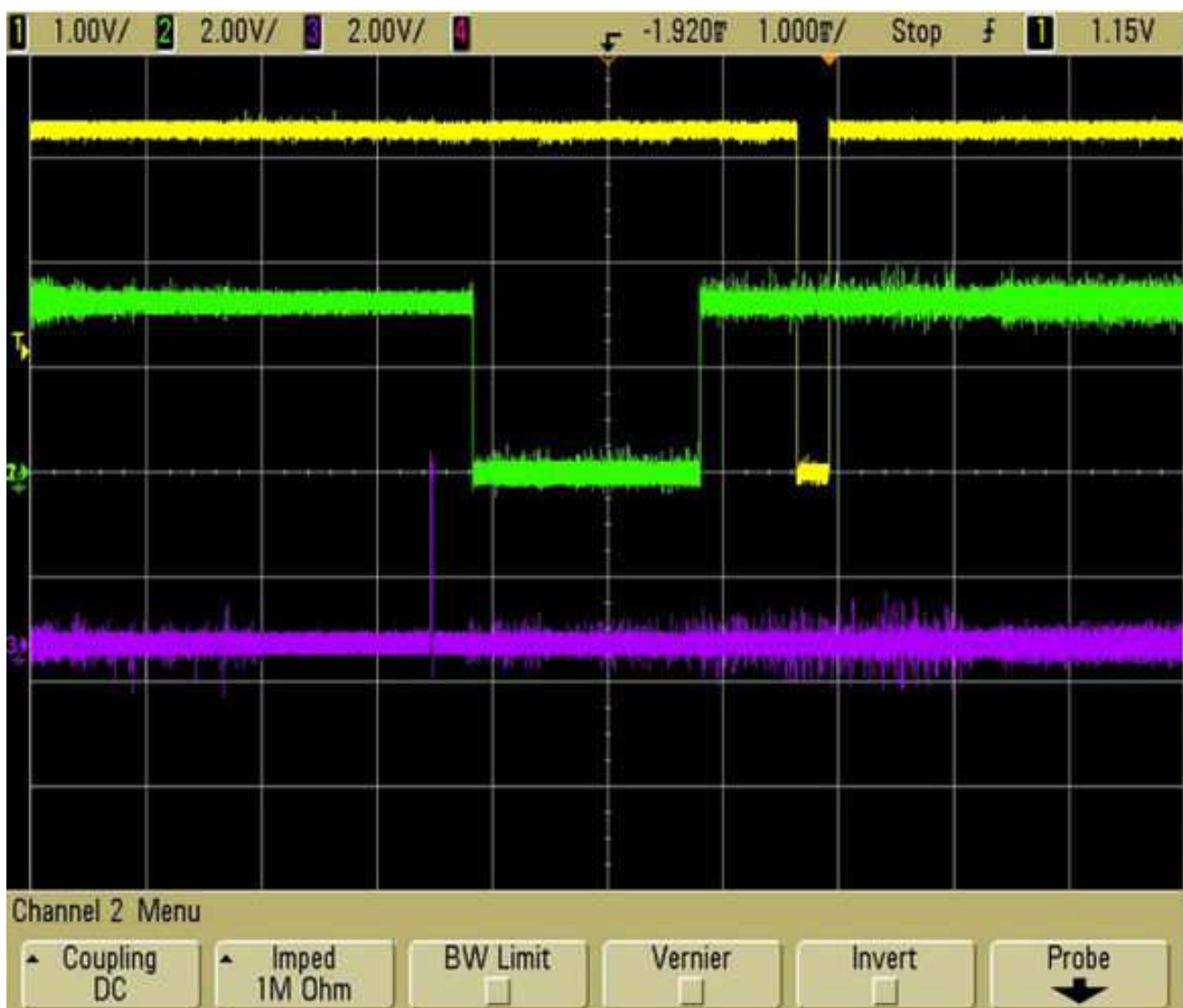
: ONE           ( -- )
  private1 10 2A *bis 10 29 *bic \ P2.4 is output
  3 2A *bic 3 29 *bis 3 2F *bis \ P2.0/1 in with pullup
  begin
    sw? if                                \ Switch pressed
      red-on 30 ms write-mode \ Show transmit action
      command xemit? 10 < if \ Command, succeed?
        100 ms red-off 10 ms \ Yes, led off
        read-mode read? if \ Data received?
          xkey ch 1 = if \ Yes, handle ascii 1
            10 29 *bix
          else
            red-on 10 ms red-off
          then
            then
              08 29 *bic \ Receive mode off
            else
              cold \ No Ack received, restart
            then
              then
                key? until 10 29 *bic ;
  ' one to app freeze

```

Netwerk opbouwen, het netwerk, van transceiver modules wordt automatisch opgebouwd. Een van de nodes de master genoemd, kan een scan starten naar nodes die binnen zijn bereik liggen.

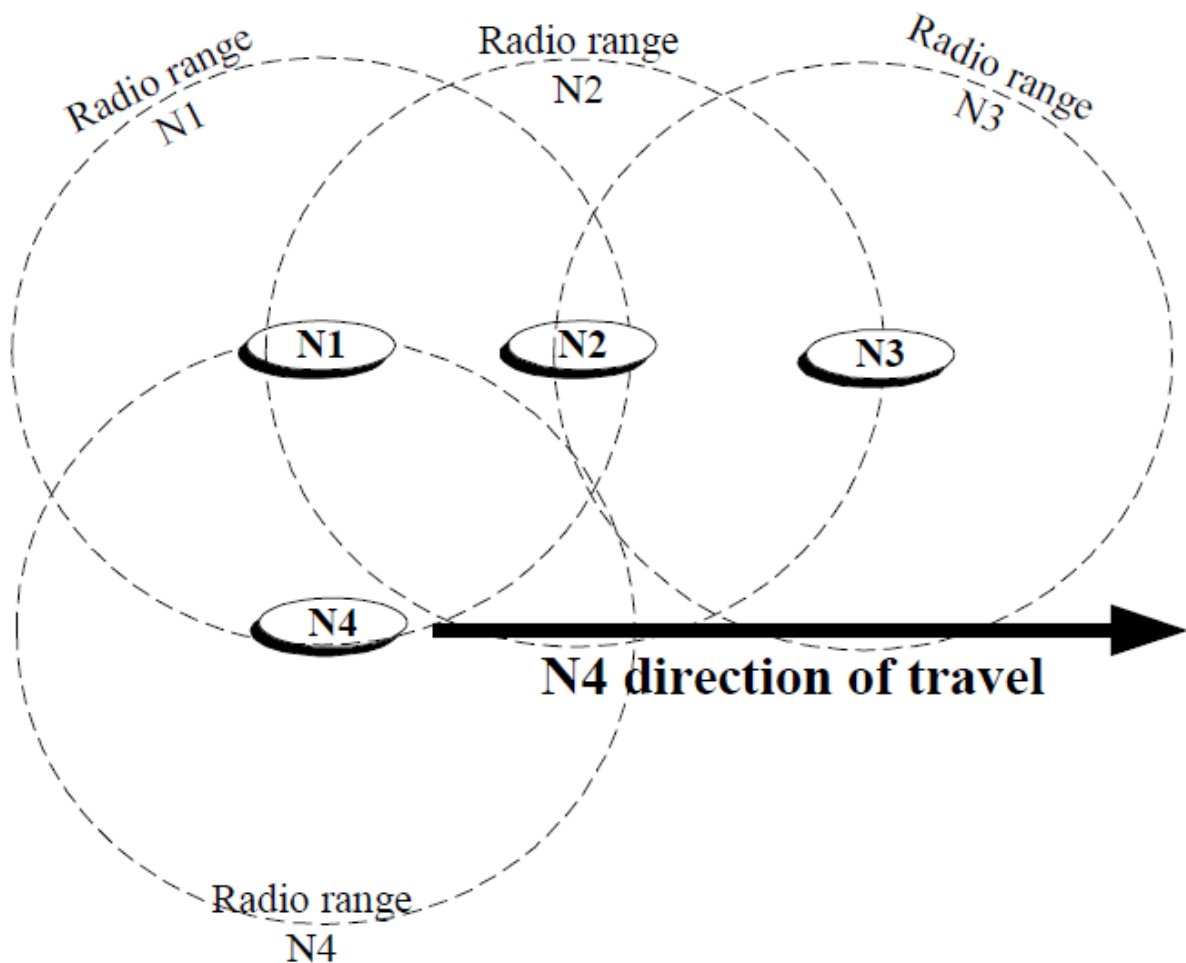
Een belangrijke voorwaarde is dat elke node een uniek adres moet hebben! Natuurlijk moeten alle transceivers in ontvangst modus opstarten.

Met de nRF24L01+ kan erg eenvoudig getest worden of een node binnen bereik ligt. Je zendt een datablok naar een node toe. Als deze binnen bereik ligt, antwoordt de node door een ACK terug te sturen. De master weet zo dat deze node bestaat!



Maar er komt meer informatie vrij, de master heeft de mogelijkheid om meerdere zend pogingen te doen. Het aantal pogingen, voor een ACK ontvangen is, wordt geteld. Dit aantal pogingen is een indicatie van de bereikbaarheid van die node.

Dit gegeven kunnen we gebruiken bij het verzenden van data via het opgebouwde netwerk. De master legt de verzamelde data voor later gebruik vast.



Voorbeeld van een bereikbaarheid en info lijst:

Node 1	0 pogingen	Power switch	Prioriteit laag	Actief
Node 2	5 pogingen	Dimmer	Prioriteit laag	Actief
Node 3	2 pogingen	8-bit sensor	Prioriteit laag	Inactief
Node 5	0 pogingen	Inbraak sensor	Prioriteit hoog	Inactief
Etc.				

Let op! Een inactieve node maakt alleen verbinding als er wat te melden is.

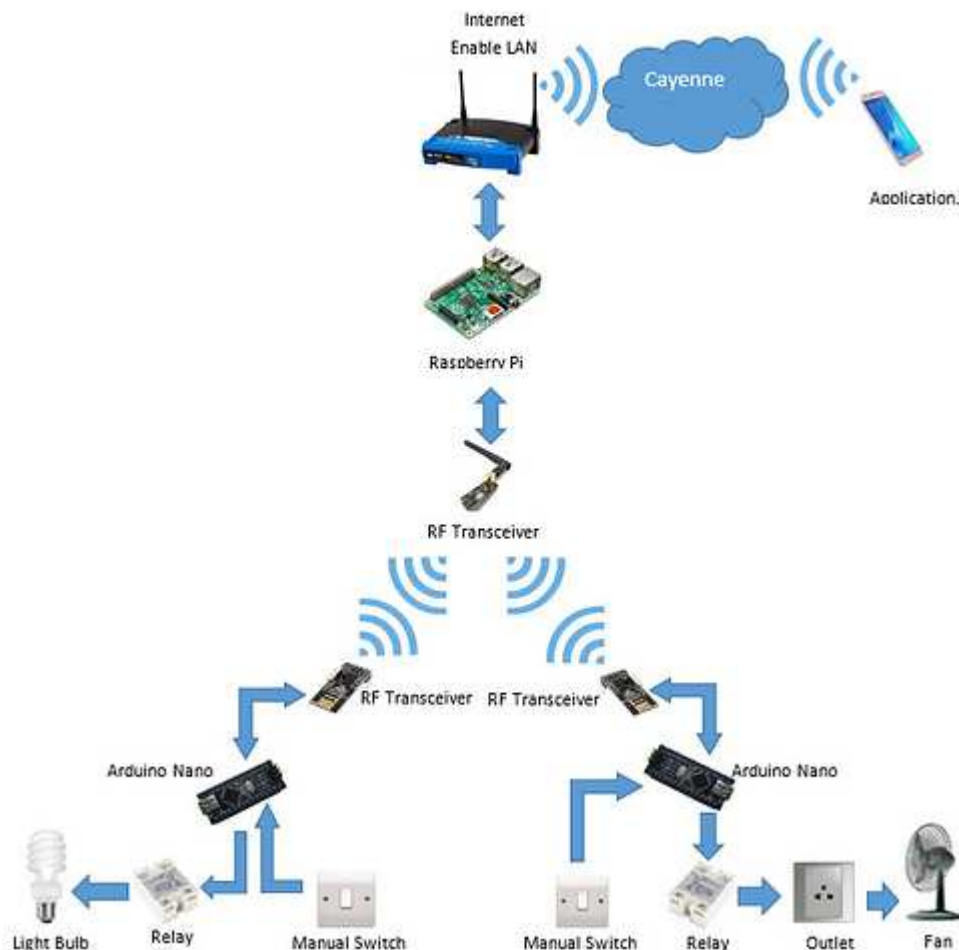
Dan kan er nog een speciaal geval zijn, b.v. Een afstandbediening. Dat zou een speciale node kunnen zijn die met elke node contact moet kunnen maken. Daarmee kan hij altijd een geschikte (binnen bereik liggende) communicatie node kiezen.

Voor de eenvoud wordt de eerste scan actie wordt gedaan door de master. Daarna kan er aan elk node binnen bereik v.d. master, een aanvullende scan opdracht gegeven worden, etc. Met zo'n systeem moeten alle nodes gevonden kunnen worden. Elke bevat uiteindelijk de netwerk structuur. Door deze kennis van de netwerk paden, is het nRF24-netwerk functioneel! Dit type netwerk heet in het Engels een **"gateway mapping network"**.

Een eenvoudige scanner om het principe te testen:

```
: SCAN)      ( +n -- )      \ Send 'S' to node +n, show result
    >node  write-mode  ch S xemit)  .lost ;

: SCAN      ( -- )
    private3      \ Check for node 1 to 7
    8 1 do  cr  i .  i scan) 600 ms  loop ;
```



Een prototype van een meer geavanceerde scanner:

```
: XEMIT?      ( c -- +n )
  0 10 0 do          \ Try it 10 times
    over xemit)      \ Send payload
  tx? if leave then  \ Ack?
    reset ch - emit 1+ \ Reset, show & add failures
    flush-tx 70 >channel \ Empty pipeline, clear loss
  loop
  nip reset ;

00 constant ME      \ This modules node number
: >NODE             ( node -- )
  >r                \ Save node address
  F0 F0 F0 F0 r@ 0A write-addr \ Receive address P0
  F0 F0 F0 F0 me 0B write-addr \ Receive address P1
  F0 F0 F0 F0 r> 10 write-addr ; \ Transmit address

create NODES 40 allot \ Direct nodes within range of master
create NNODES 40 allot \ Nested nodes

: TRY      ( +n1 node -- +n2 ) \ +n is number of found nodes
  dup >node red-on write-mode \ Address 'node'
  ch S xemit? 10 < if        \ Send S, attempt succeeded?
    >r 1+ r> over nodes + c! \ Yes, store node number
  then
  red-off ;

: FIND-NODES ( -- +n )
  setup24L01 0 \ No nodes found
  100 0 do i 1+ try loop \ Try all nodes
  dup nodes c! ." Nodes " . ;

: SCAN      ( -- ) find-nodes find-nested-nodes ;
```

Finish: Als het netwerk opgebouwd is staan alle nodes in de ontvangst stand. Als een node van status veranderd mag hij data verzenden. Botsingen op het netwerk worden met een gemiste ACK gedetecteerd.

Een mogelijk record formaat, max. Lengte 32 bytes:

Een voorbeeld met een record van zes bytes:

Best. node	Org. node	Soort	Command	Data-0	Data-1
------------	-----------	-------	---------	--------	--------



Diagram van mogelijke verbindingen

Voorbeeld van knooppunt (node) adressen:

00 = Meester	Kan (re)scan for nodes starten
01 = Node 1	Actieve slaaf node 1
02 = Node 2	Actieve slaaf node 2
03 = Node 3	Niet actieve slaaf node 3
FF = Afstand bediening	Besturings node

Knooppunt (node) ID:

Ofwel was is de functie van een node.

Node ID: 00	Meester node
Node ID: 01	Vermogens schakelaar (Elk voltage)
Node ID: 02	Dimmer (Elk voltage)
Node ID: 03	8-bit analoge sensor
Node ID: 04	16-bit analoge sensor
Node ID: FF	Besturings node

Losse flodders:

Ruimte detectie kan mogelijk gedaan worden door de besturings node. Die werkt dan als een soort transponder.

De besturings node, moet dan beweging kunnen detecteren. De zender moet op laag vermogen (klein bereik) afgesteld staan. Dan wordt alleen contact gemaakt met nodes die vlakbij zijn.

Er kunnen "gateways" gemaakt worden naar andere protocollen.

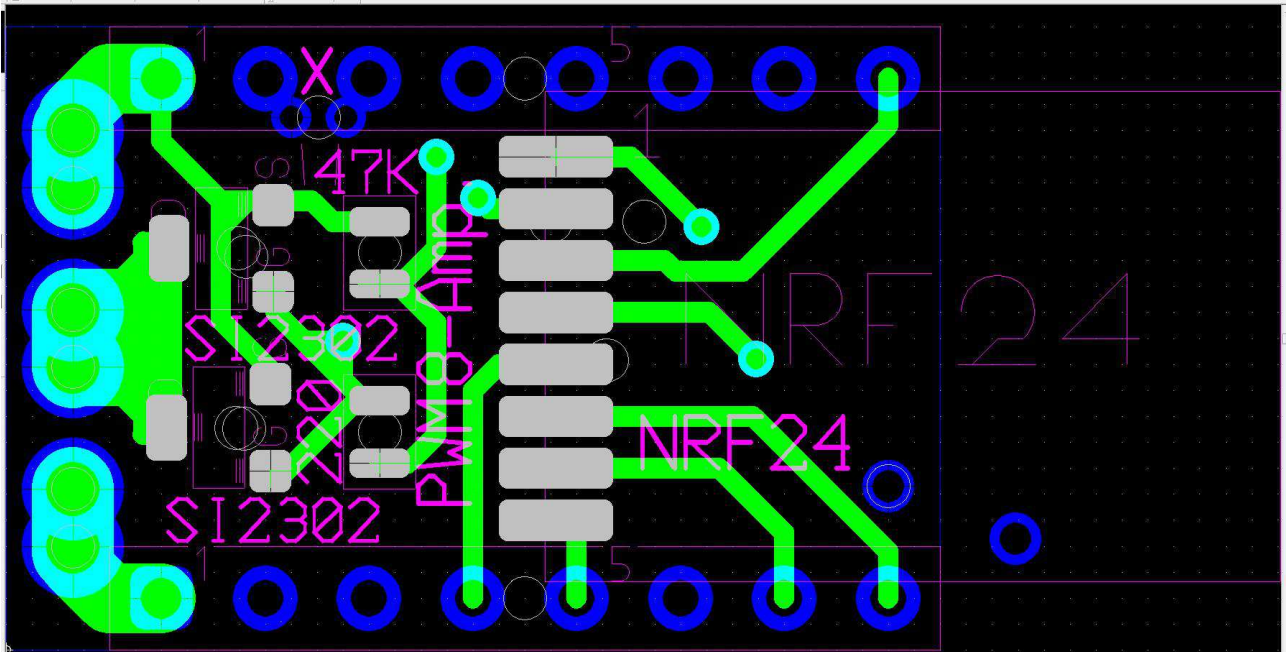
Op 433 Mhz gebaseerd: Flamingo schakelaars, KlikAan-KlikUit, Deurbel, etc.

Of naar infrarood: b.v. RC5 zodat er nog maar één afstandbediening nodig is.

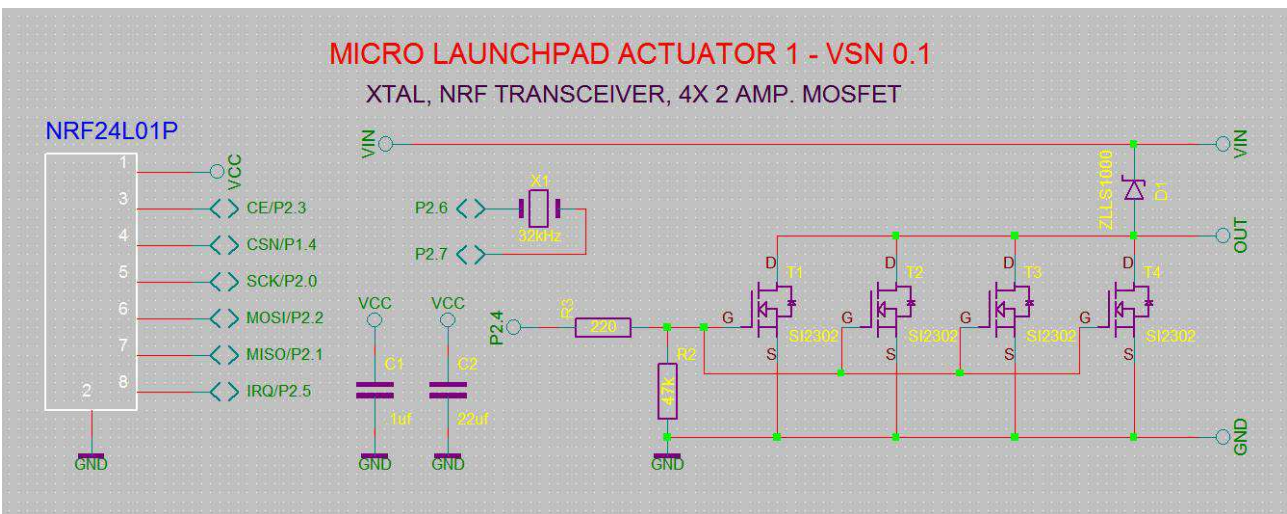
Als het bereik van de nRF24 een probleem is:

- Bit snelheid verlagen van 2 Mbit naar 1 Mbit of 250 Kbit
- Data pakket (payload) zo klein mogelijk
- Betere antenne
- Module met groter zendvermogen

Micro launchpad uitbreiding met nRF24L01+



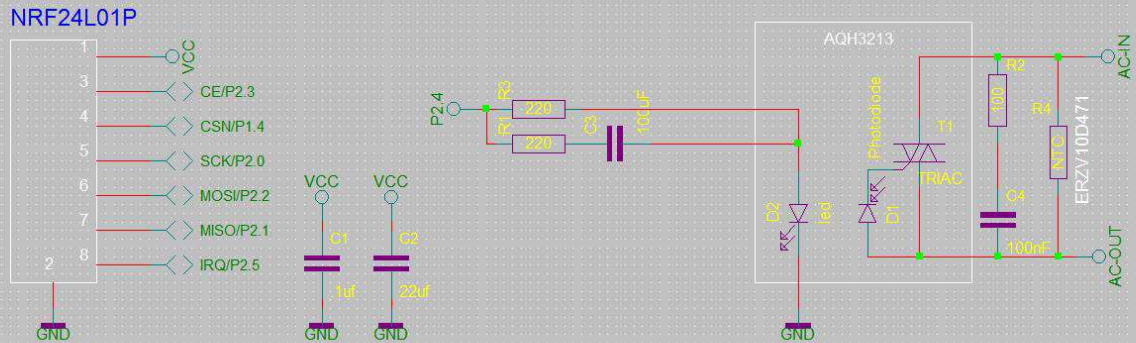
12 V DC – 8 Amp. vermogenschakelaar node



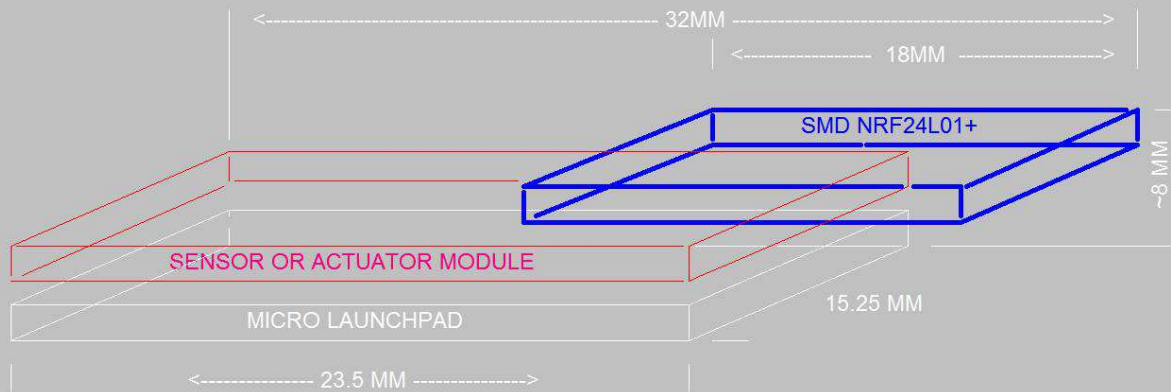
Schema DC vermogenschakelaar

MICRO LAUNCHPAD ACTUATOR 2 - VSN 0.0

XTAL, NRF TRANSCEIVER, 1 A 240V TRIAC SWITCH

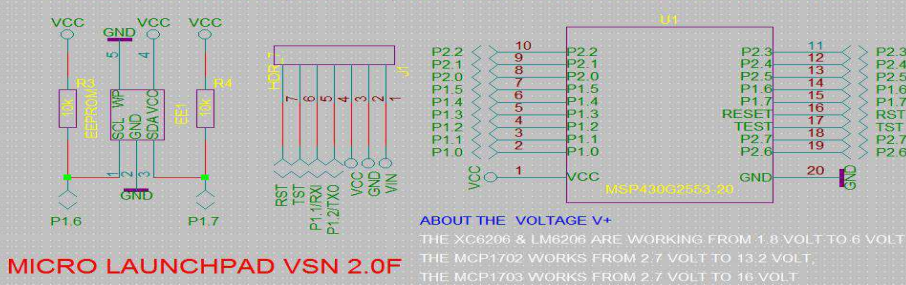


Schema AC vermogenschakelaar



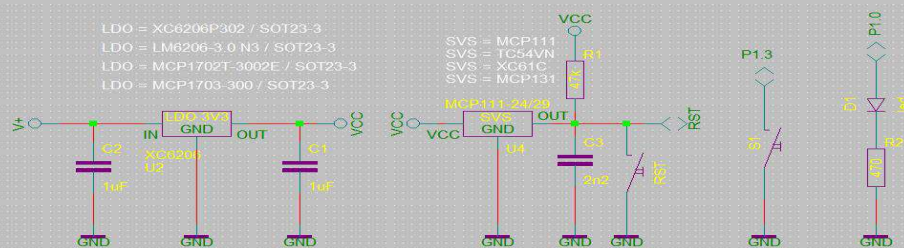
Stapelen van print modules met MLP

ONE LED, 2 SWITCHES, EEPROM, 3V3 LDO, VOLTAGE SUPERVISOR 2V4 TO 3V



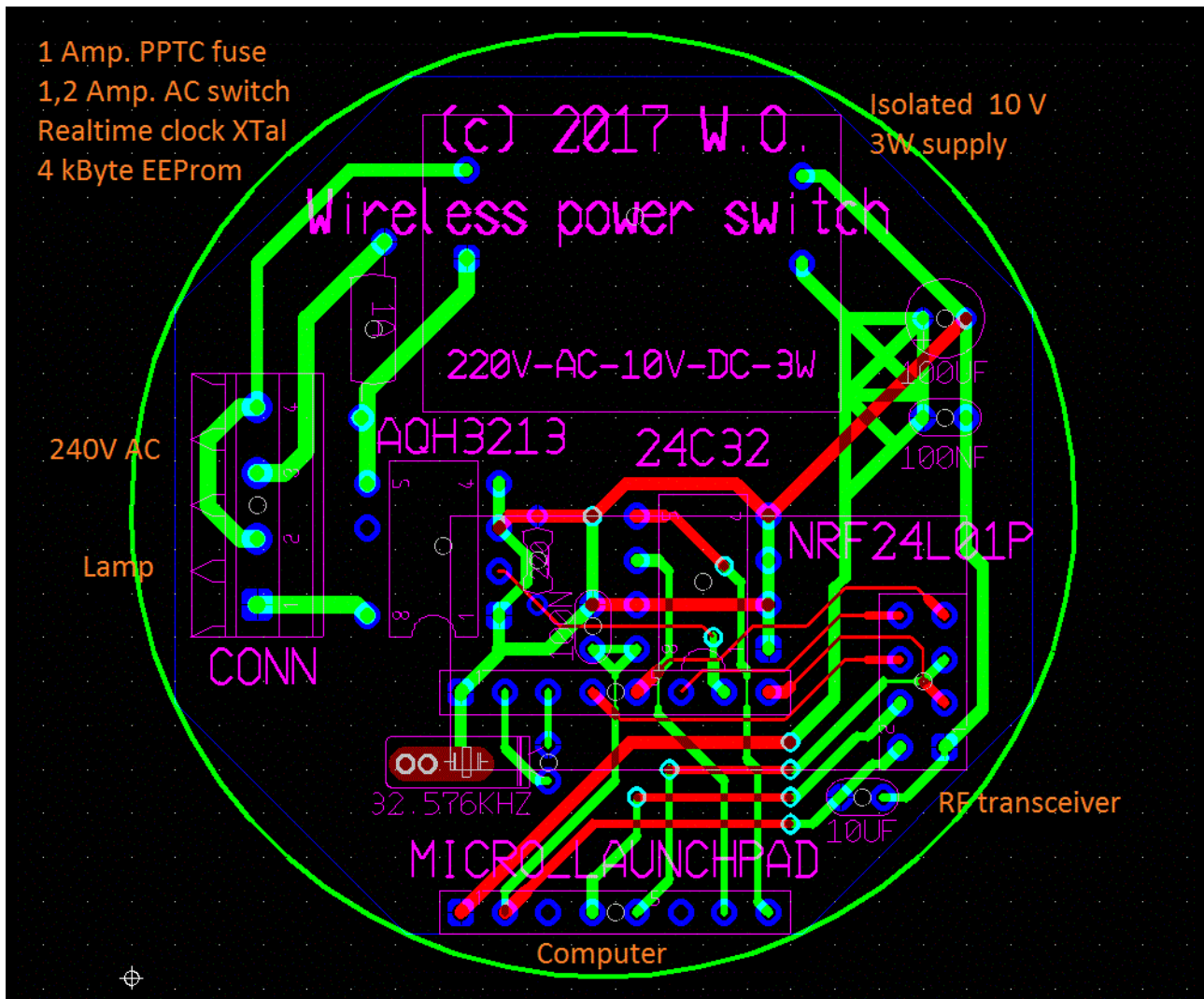
MICRO LAUNCHPAD VSN 2.0F

LDO = XC6206P302 / SOT23-3
LDO = LM6206-3.0 N3 / SOT23-3
LDO = MCP1702T-3002E / SOT23-3
LDO = MCP1703-300 / SOT23-3



Mogelijke uitbreiding MLP

240V AC schakelaar als voorbeeld:



Dit is een vermogens schakelaar en/of dimmer. De gekozen schakelaar AQH3213 schakelt op een nuldoorgang aan. De AQH3223 kan random aan/uit schakelen voor een dimmer.

Met de nRF24L01P bouwen we een draadloos netwerk.

De Micro Launchpad bevat een RTC en samen met de 24C32 een geheugen voor adressen en instellingen.

Als voeding wordt een compacte Led stroombron misbruikt. Het geheel past al in een kleine opbouw trekdoos.

Ombouw 240V AC schakelaars:



Flamingo Schakelaarset Met Afstandsbediening

Geschikt voor alle gangbare lampen

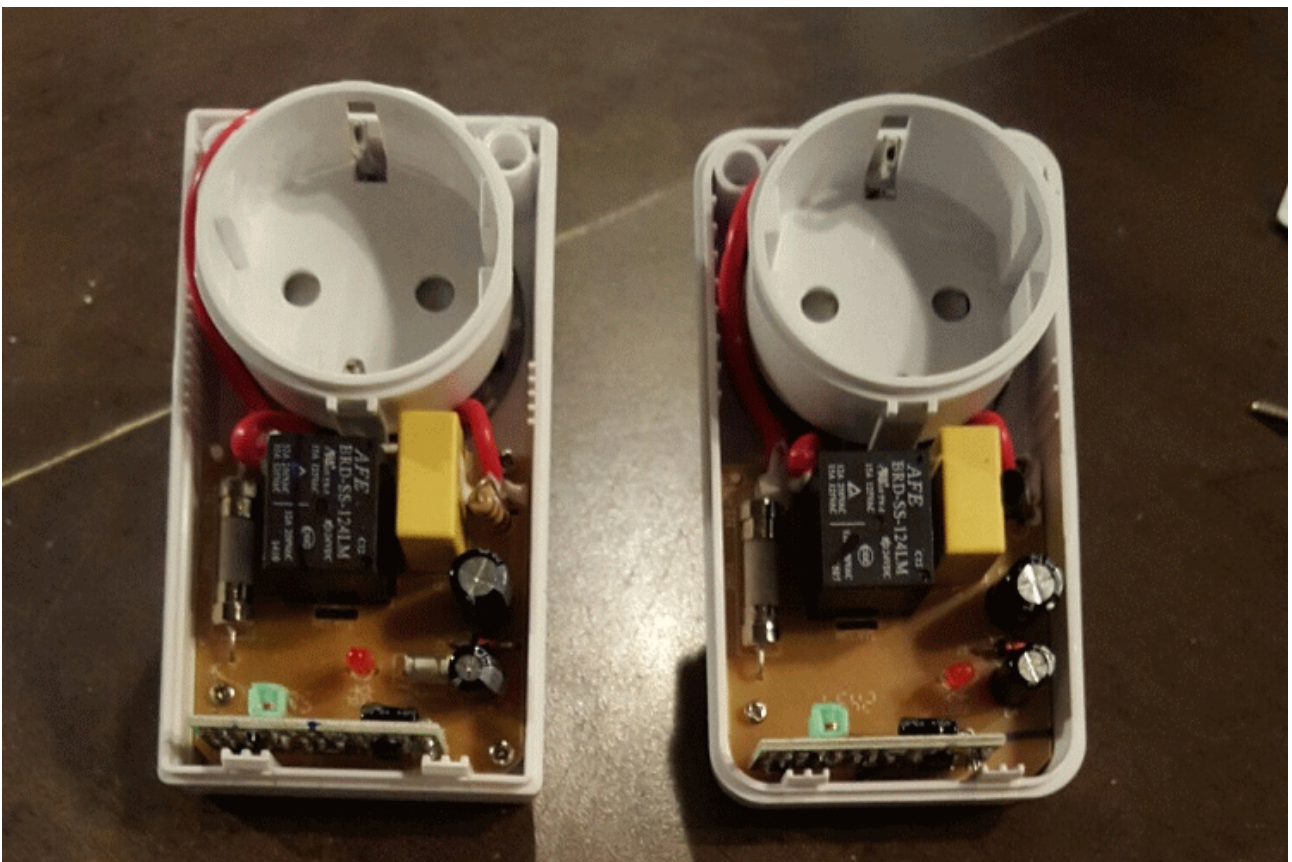
995

VOEG TOE AAN FAVORIETEN

EEN MOMENT OM TE DELEN

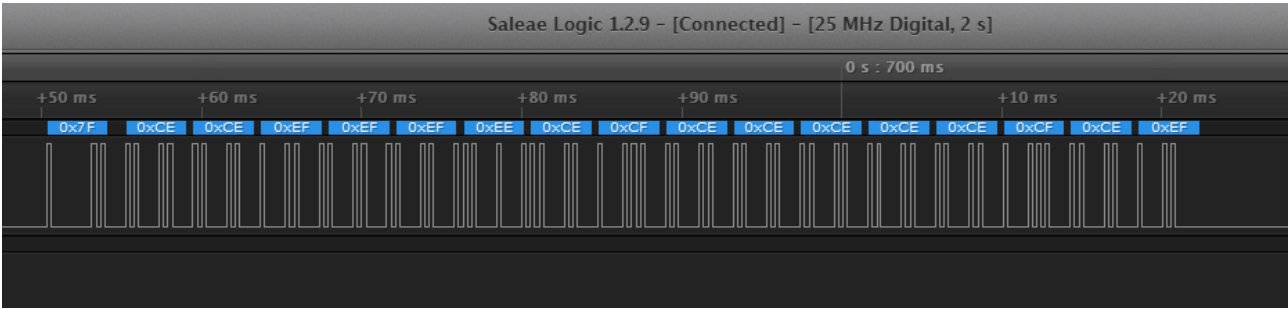


Action 240V AC schakelaar set

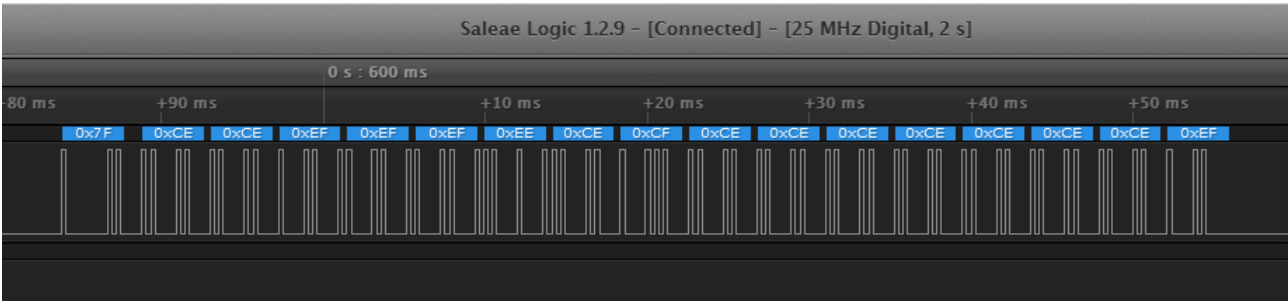


240V AC schakelaars opengemaakt

Flamingo set ontrafelt



Verzonden string voor toets A-aan



Verzonden string voor toets A-uit

De code string bestaat uit 33 puls reeksen van 1, 2 of 3 pulsen, vooraf gegaan door een startpuls. Zender één:

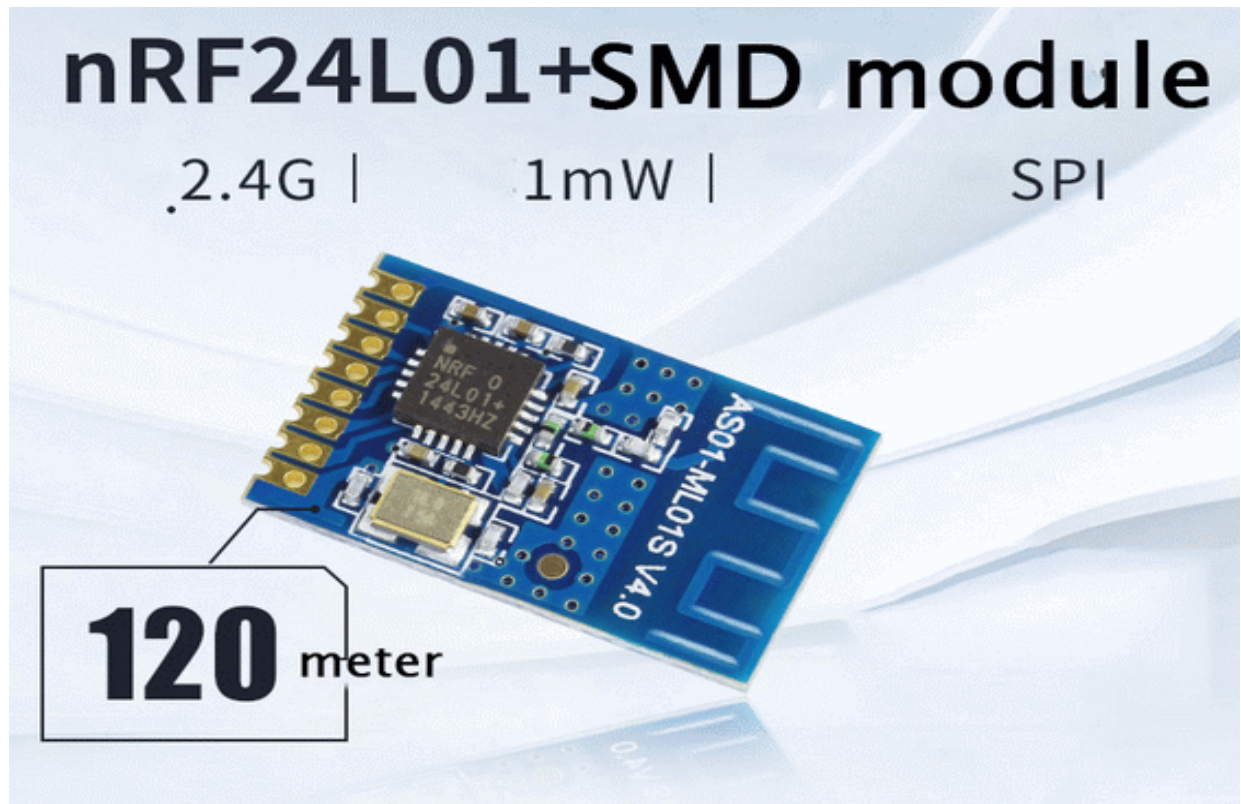
A = Aan	S22222122222313213222222222	2132212
A = Uit	S22222122222313213222222222	2222212

De laatste zeven puls reeksen zijn verschillend per toetscode, maar gelijk bij iedere zender.

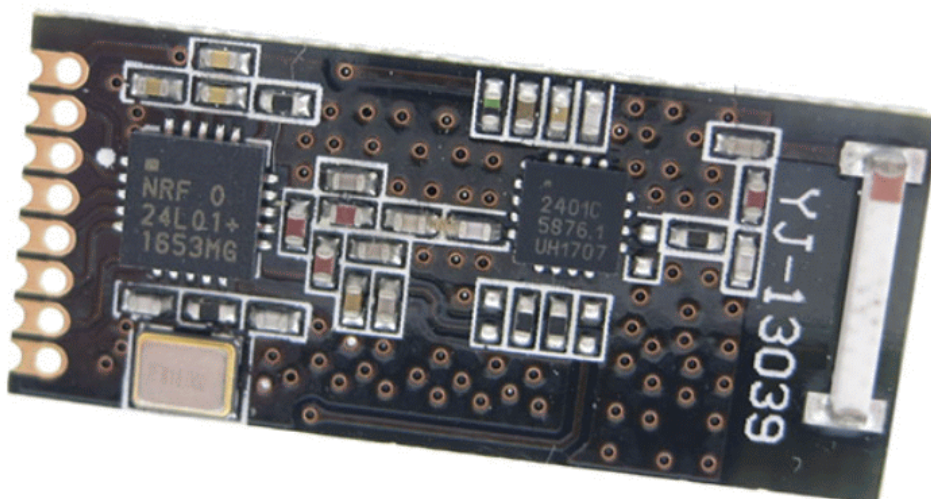
Alles aan	S22131231312232123222222222	1232212
Alles uit	S22131231312232123222222222	1322212

De eerste 26 puls reeksen zijn verschillend per zender. Zij zijn het basis adres van zend/ontvang set. Om deze sets makkelijk te kunnen gebruiken, moeten we een programma maken om het basis adres te loggen. De ontvangers onthouden het eerst ontvangen adres en toetscode.

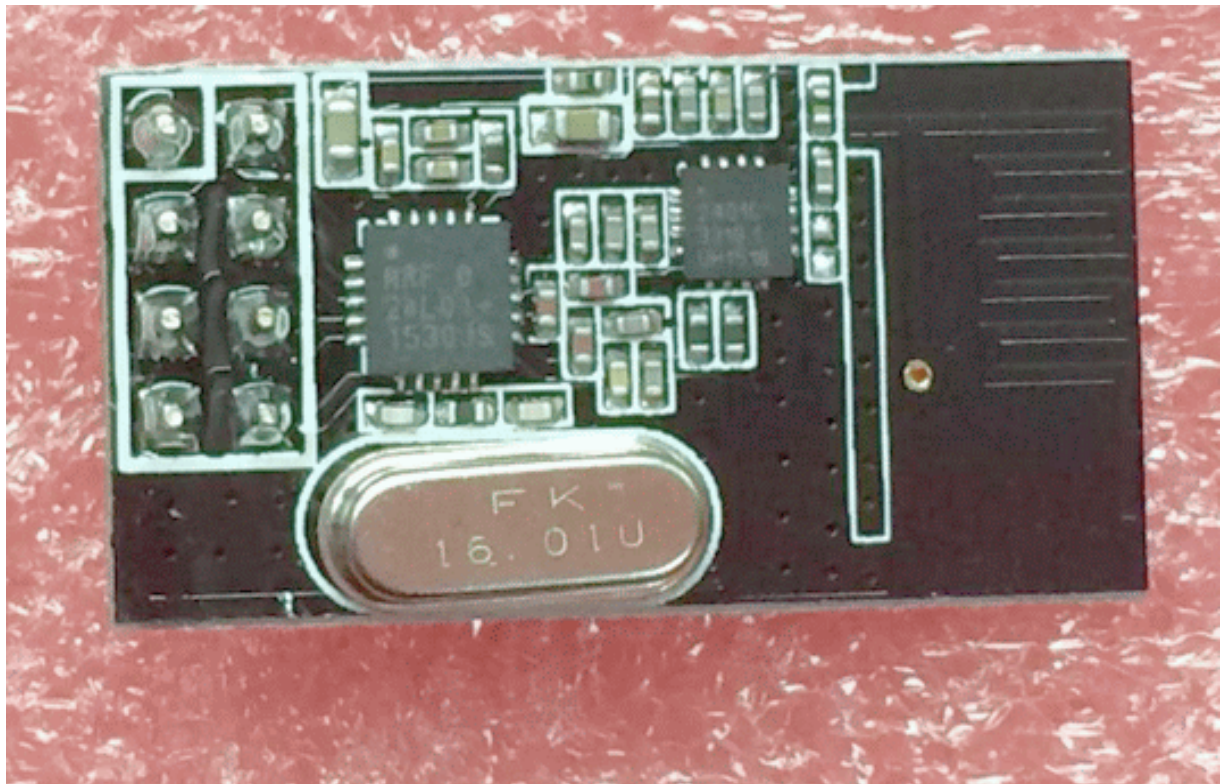
Diverse nRF24L01 modules



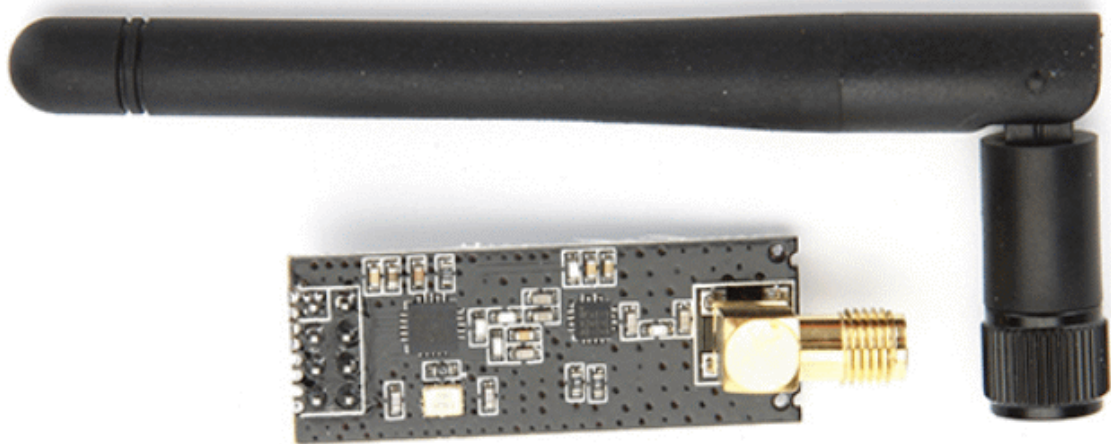
Compacte nRF



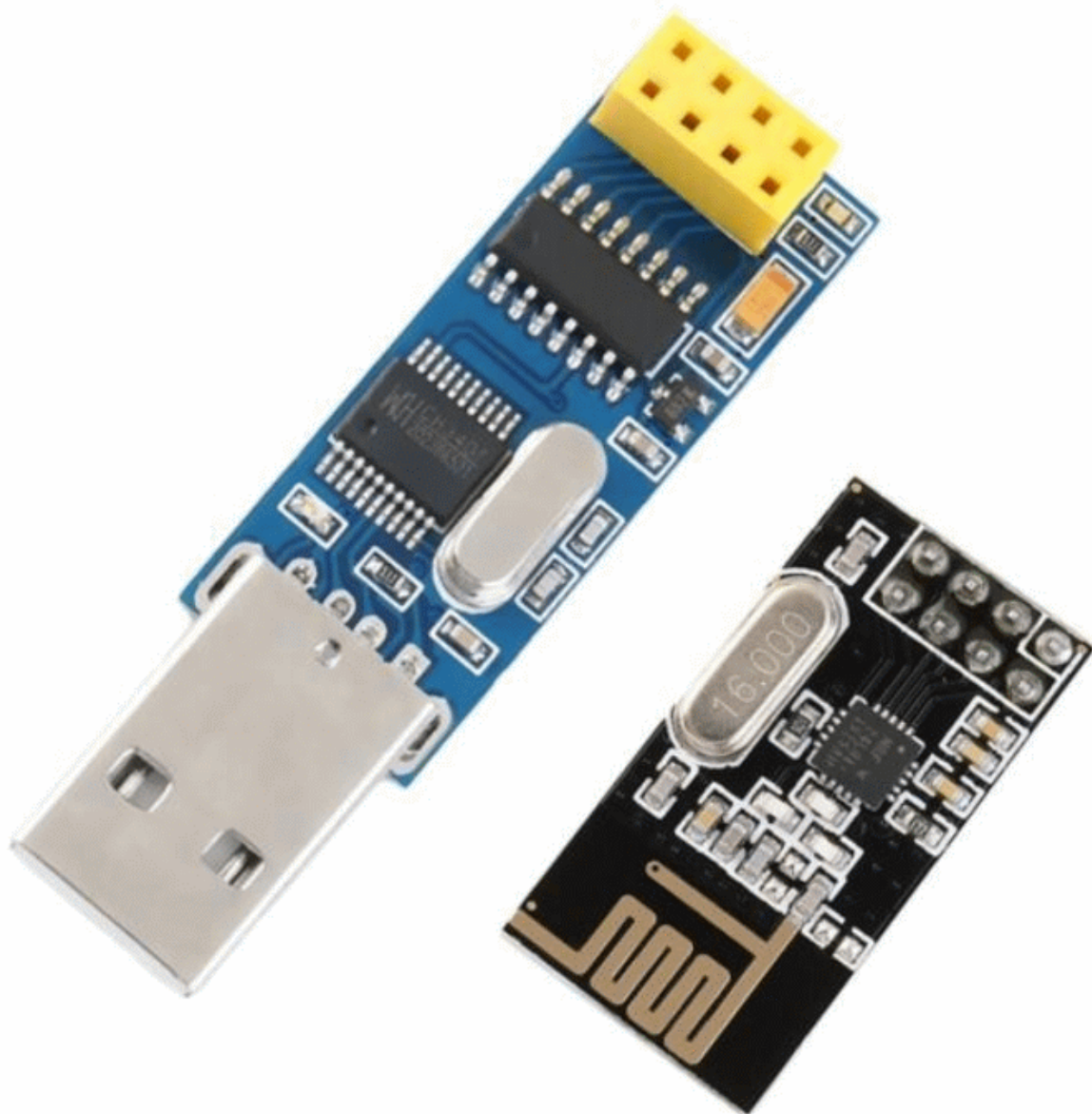
Compacte nRF met versterker & keramische antenne



nRF met versterker & print antenne



nRF met versterker & staaf antenne



nRF PC interface met standaard module