



Programma HCC!Forth komende zaterdag

**Zaterdag 13 juni 2015 op de bekende locatie:
het gebouw van peuterspeelzaal Juliaantje naast de Zuiderkapel aan de
Boslaan 1 in Bilthoven**

10:30 Zaal open en koffie voor vroege vogels

11:00 Overzicht activiteiten van de werkgroepen

11:10 One-armed-bandit door Gerard Vriens met demo op het orgeltje

12:00 Willem Ouwerkerk behandelt het bouwen van complete applicaties:

- in noForth een met Bluetooth op afstand bediende robot
- in ByteForth een miniatuur Ultrasoon sensor

13:15 Pauze

13:45 Project-ideeën door Albert van der Horst

15:00 Sluiting.

Tot ziens



Zuiderkapel Boslaan 1 Bilthoven

One-armed bandit (en robot-truc)

Gerard Vriens

"The one-armed bandit" is een muziekstukje, speciaal geschreven voor het computergestuurde orgeltje van de HCC Forth interessegroep. Als alles volgens plan verloopt, zal het op de bijeenkomst van 13 juni zijn officiële première beleven. Het muziekstuk zelf is kort, niet langer dan zo'n anderhalve minuut. Dat is voorbij voordat je het weet. De ervaring leert echter dat muziek (en kunst in het algemeen) meer genoeg geeft naarmate je er meer van weet en begrijpt.

Daarom zal ik vóór de "uitvoering" iets vertellen over de muzikale opbouw van het stukje. Daarbij komt wat elementaire muziektheorie kijken. Hopelijk zal de opletten toehoorder ook enige interessante parallellen opmerken tussen het schrijven van een muziekstukje en het schrijven van een computerprogramma.

Het tweede onderwerp (bij voldoende tijd):

Iedereen die een robot bouwt, wil het ding "iets interessants" laten doen. Daarbij kunnen we de natuur als inspiratiebron gebruiken. Het volledige gedragsrepertoire dat we bijvoorbeeld bij insecten kunnen observeren ligt op dit moment nog buiten ons bereik -- hoe ver buiten ons bereik, daarover verschillen de meningen. Maar met een simpele "goocheltruc" kunnen we onze robotjes toch een leuke show laten opvoeren. Dit wil ik met een korte demonstratie laten zien.

Voortgang met noForth, Egel, launchpad etc.

Willem Ouwerkerk

We zijn een stapje verder met Egel. Ik heb acht voorbeelden voor de Launchpad klaar, met foto en schema.

Het PWM project zit er volledig bij. De Egel pagina neemt nu vorm aan, ik moet nog wat aandacht aan de tekst besteden.

Daarnaast enkele foto's van het negende project: een kleine wandelende robot met vier servo's. Dit project is nog in de test fase, ik probeer het voor de bijeenkomst af te ronden.

Voor de Egel projecten heb ik een print van 5x5cm ontworpen die ik voor €2,00 per stuk in China kan laten maken. Op de print passen alle connectoren zoals op de hardware explorer:

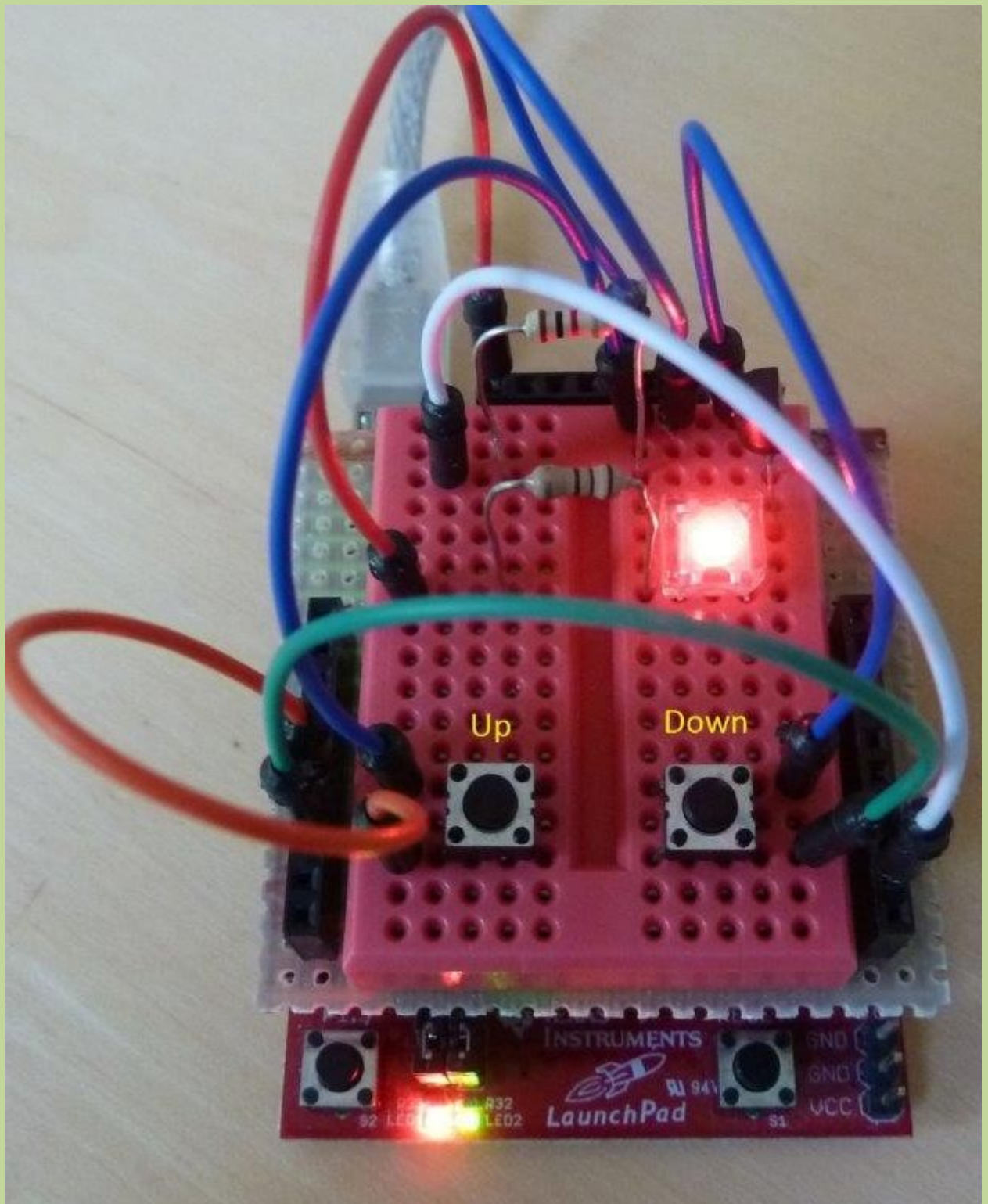
<http://home.hccnet.nl/anj/nof/hardware%20explorer.pdf>

Daarnaast kunnen er vier servo's met een aparte voeding op aangesloten worden.

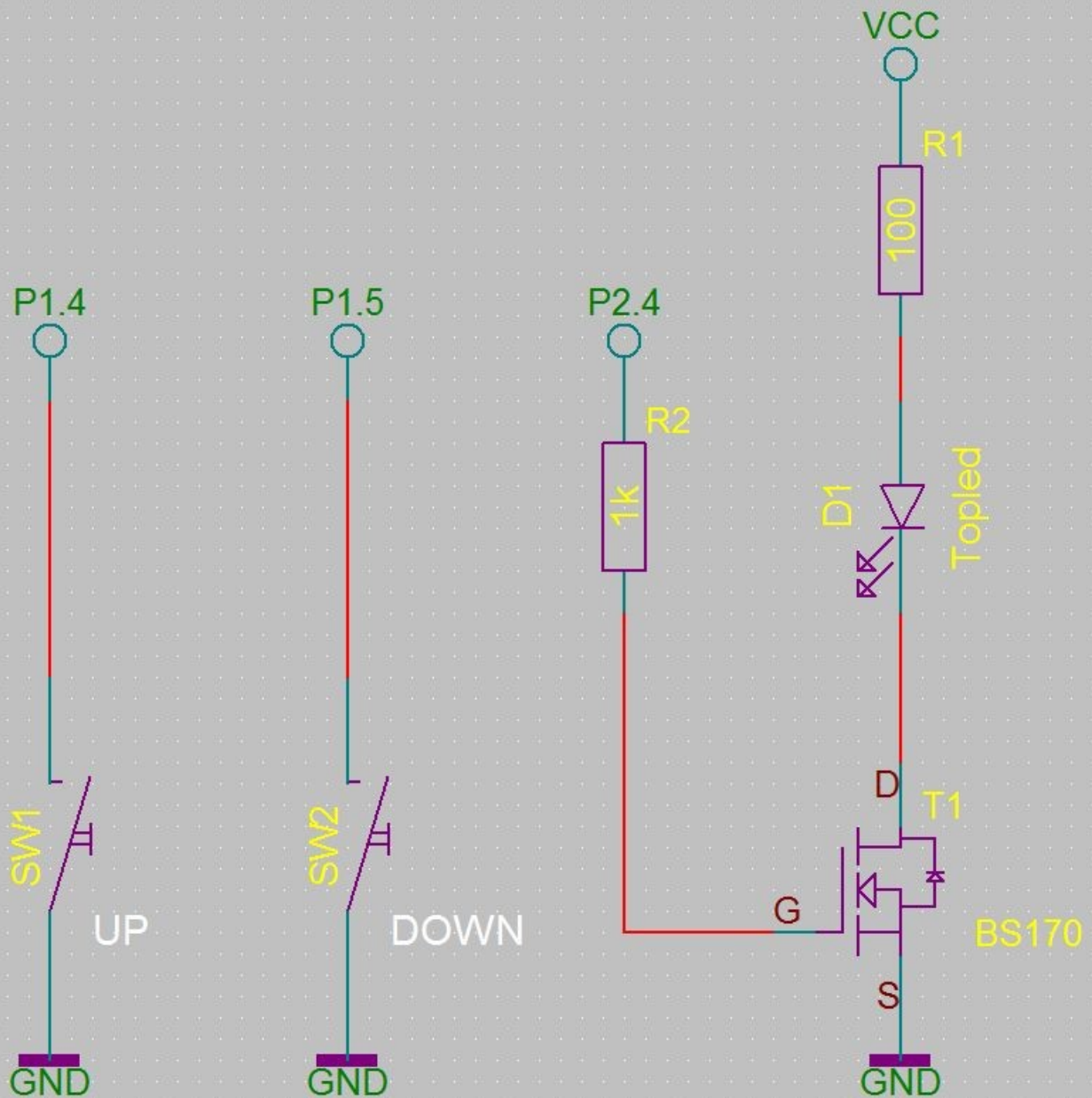
Op de bijeenkomst behandel ik twee afgeronde applicaties, een in ByteForth de MUS-3, de tweede in noForth een via Bluetooth bestuurd applicatie.

Albert Nijhof heeft ondertussen de nieuwste update van noForth op de website geplaatst:

<http://home.hccnet.nl/anj/nof/noforth.html>



f06 - pwm using compare unit



s06-schema van pwm using compare unit

Programma (e06-using compare unit.f)

(For noForth 2553 lp.0, C&V version: Port input using switches at P1.4 & P1.5 . Using the onboard P1.0 led as output. Pulswidth power control with 4 KHz PWM at P2.4 or P2.5 with a resolution of 2000 steps)

(Port-1 must be wired to 2 switches, placed on the launchpad experimenters kit, Wire P1.4 & P1.5 to the switches on the breadboard. The pinlayout can be found in the hardware doc of the launchpad. Port-1 bit 3 holds a switch on the Launchpad board.)

(P2.4 must be wired to a logic power FET like the BUK101 using a suppression diode like 1N4001. Any 5 or 6 Volt DC motor or lamp may be used. Take care for the maximum USB current!)

HEX

```
( Address 020 = plin, port-1 input register
  Address 022 = pldir, port-1 direction register
  Address 029 = p2out, port-2 output with 8 leds
  Address 02A = p2dir, port-2 direction register
  Address 02E = p2sel, port-2 selection register
  Address 180 = talctl, timer a1 compare mode
  Address 186 = talcctl2, timer a1 output mode
  Address 192 = talccr0, timer a1 period timing
  Address 196 = talccr2, timer a1 Duty cycle )

( FEDCBA9876543210 bit-numbers
  0000000000010000 - Choose output bit4 or bit5
  0000000011000000 - toggle-set output
  0000001000110100 - TA clear, up/down, SMCLK, no presc. )

\ Period length is 2000 clock cycles ( 2x#CYCLUS )
dm 1000 constant #CYCLUS
value POWER

: STOP-TIMER1      0 180 ! ;
: SET-PWM          #cyclus umin 196 ! 010 029 *bic ;
                  ( 0 to #CYCLUS -- )
: PWM-OFF         stop-timer1 010 02E *bic ;

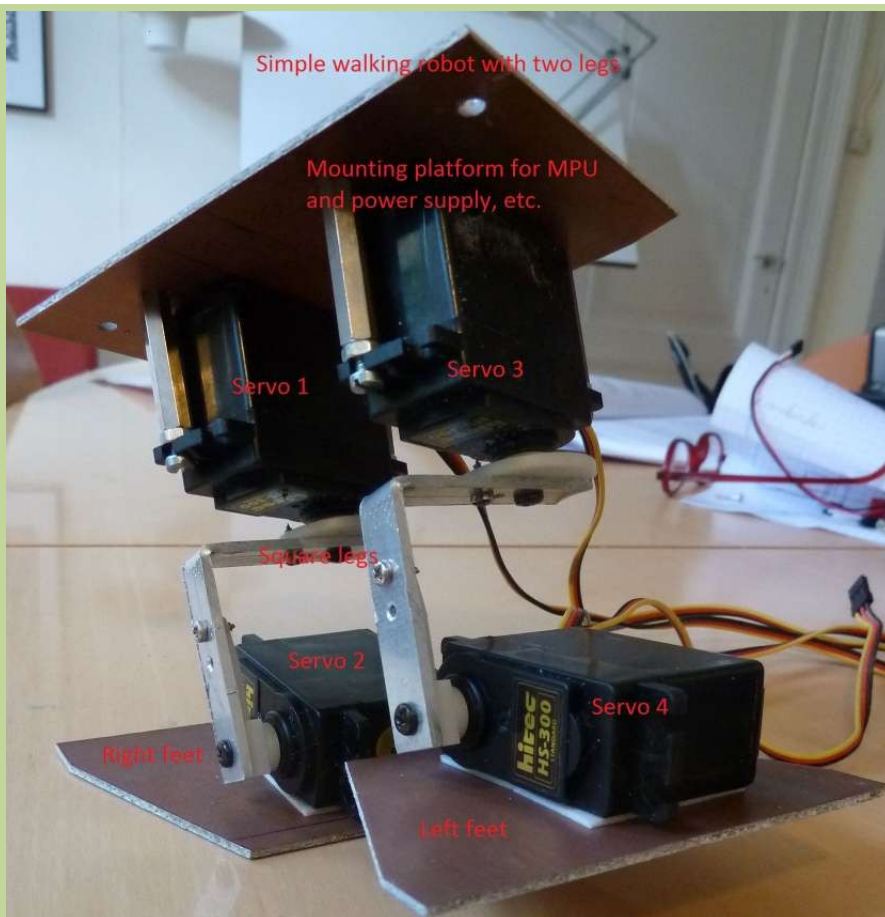
\ PWM AT 2.4 or P2.5
: SETUP-PWM       ( -- )
  038 022 *bic 038 020 *bis \ P1.3 to P1.5 are inputs
  038 027 *bis           \ with pullup
  010 02E *bis 010 02A *bis \ Set PWM to output pin P2.4
  stop-timer1 #cyclus 192 ! \ Set period time
\ 0C0 186 !           \ Set output mode negative pulse
  040 186 !           \ Set output mode positive pulse
  234 180 !           \ Activate timer
  0 to power ;

: LOWER           power if -1 +to power then ;
: HIGHER          power #cyclus u< if 1 +to power then ;

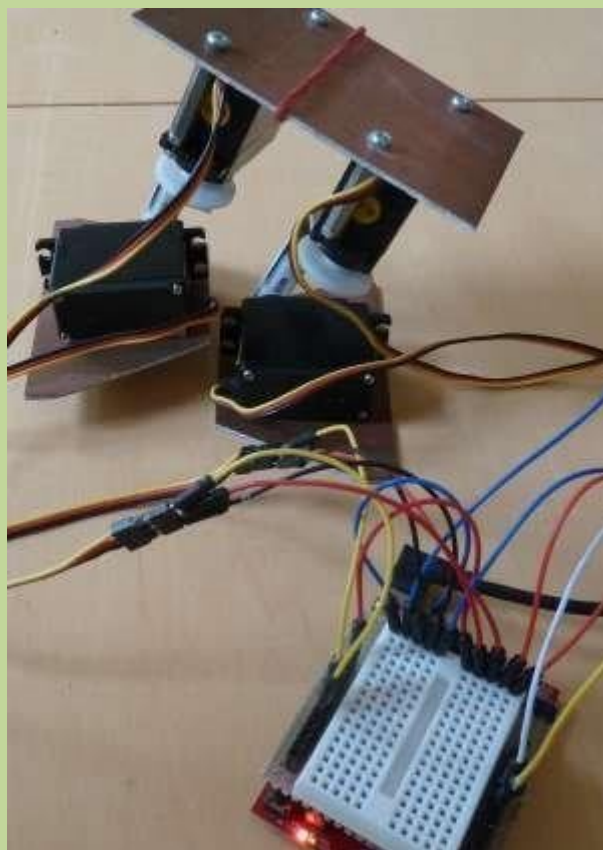
\ The switch on P1.4 is the increase speed key
\ The switch on P1.5 is the decrease speed key
: SET-POWER       ( -- )
  010 020 bit* 0= if higher then
  020 020 bit* 0= if lower then
  power set-pwm 2 ms power .
  ;

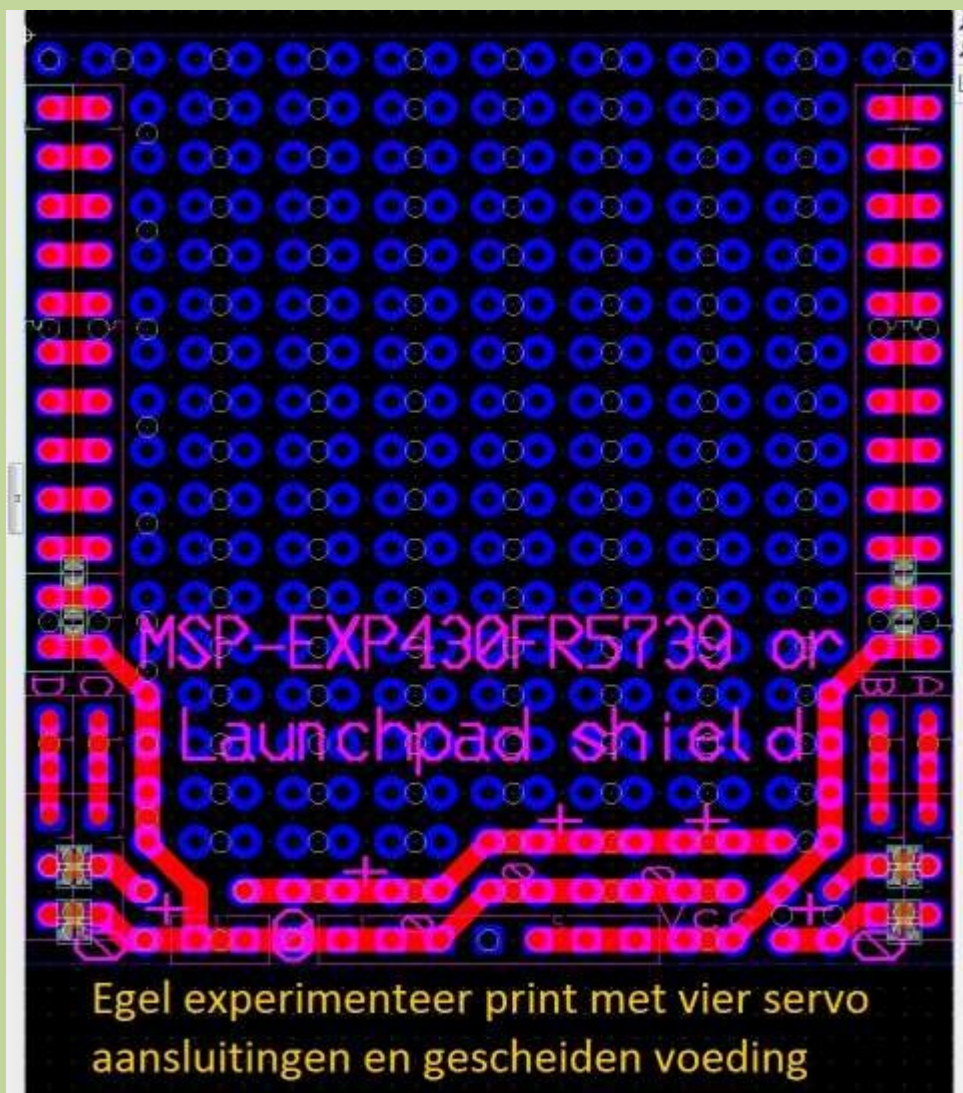
\ Two demonstrations of pulsewidth modulation power control
: CYCLUS          setup-pwm #cyclus 1+ 0 ?do i set-pwm 02 ms loop
                  pwm-off ;
: POWERCONTROL    setup-pwm begin set-power key? until pwm-off ;

pwm-off freeze
```

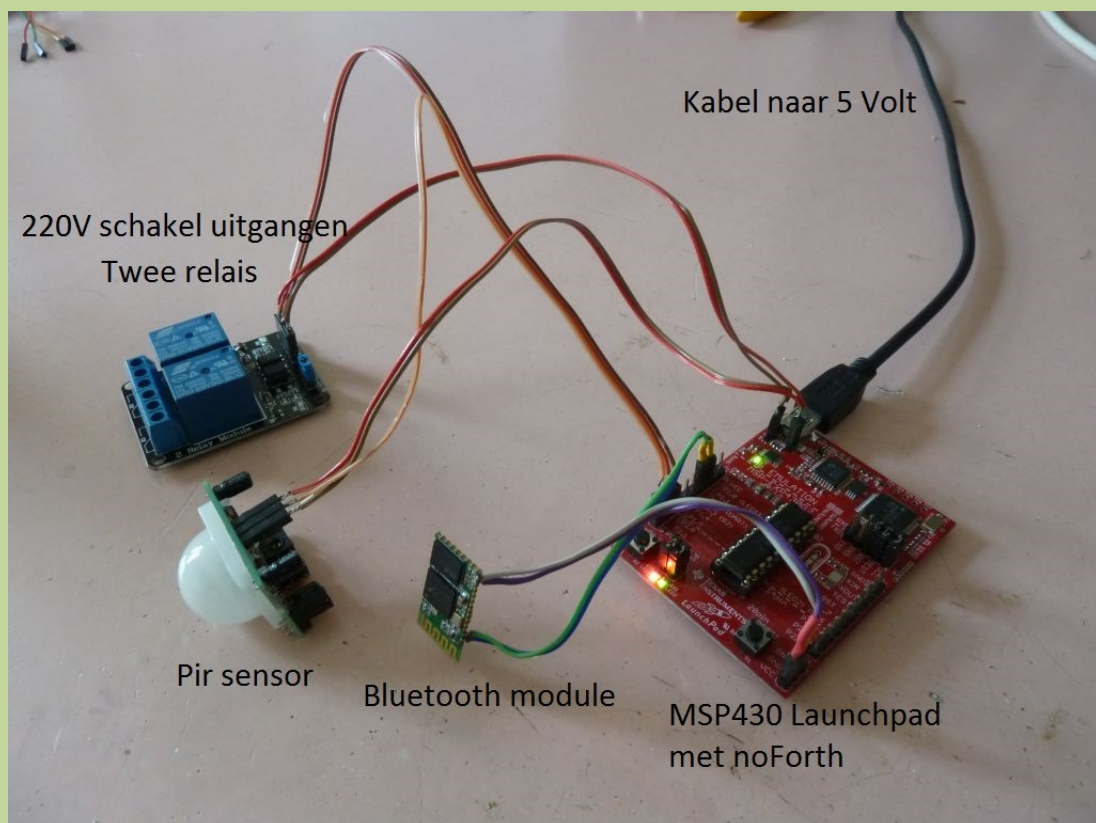


p09 - walking biped robot





Launchpad experimenteer print voor Egel



Finished app

Programma voor het uitlezen van de MUS-3 sensor (ultrasoon)

```
\ GP2D02 like sensor interface for MUS-3
\ P2.5 GP2-CONTROL          \ Vin - 203
\ P2.6 GP2-DATA            \ Vout - 201
\                            \ Direction 205, Resistance 206

HEX
: !LEDS          ( u -- )          dup F0 and 222 c!  0F and 322 c! ;
: GP2-SETUP      ( -- )           20 205 c!  40 207 c! ;

\ About 40 us clockcycle width about 25 KHz freq.
: NEXT-BIT      ( x1 -- x2 )      \ Read one databit
    20 203 *bis ( 1 wait ) 20 203 *bic
    2* 40 201 bit* 2* 2* >< or ;

\ Give distance to an object in centimeter, lower is nearby.
\ One measurement takes about 14 milliseconds.
\ The range of the sensor is from 1 cm to 200 cm
: DISTANCE      ( -- cm )        \ Read distance from MUS-3 sensor
    20 203 *bic 0D ms
    0 8 0 do next-bit loop
    20 203 *bis ;

\ Show distance on onboard leds
: GP2DEMO      ( -- )
    gp2-setup begin distance !leds key? until 0 !leds ;

DECIMAL
' gp2demo to app freeze

\ End
```

Project-ideeën

Albert van der Horst

De lezer.

Een boek rust op twee verende plateaus. Een vork-arm drukt het boek neer, totdat de bovenkant gelijk ligt met de houder en dan vergrendelt.

De bladzijde is nu in een exacte positie en kan gelezen worden.

Met een druk op de knop genereer je een flits en gaat de verlichting voor een moment aan. Een memopad of een notepad reageert op een flits en maakt een foto of een prefoto.

Het omslaan van de bladzijden wordt vergemakkelijkt door een zwart plaatje van 1.5 mm dat poreus is. Dit is tegenwoordig gemakkelijk te maken met een 3-D printer. Het rust op de bladzijden en zuigt zich vast. De bedoeling is dat het delicaat genoeg is om met minder kostbare boeken te kunnen rauzen.

De weerstandmeter.

De weerstandmeter heeft een rode, oranje, gele, groene en blauwe led. Kleur combinaties voor bruin, zwart en goud moeten we nog verzinnen (bijv. rood+groen is bruin; rood, geel en blauw is donkergrijs zwart; geel en oranje is goud).

Het is voor een belangrijk deel een mechanisch project om 0204, 0306 en 0408 weerstanden ergens in te leggen zodat ze gemeten kunnen worden. Denk aan een uitsparing in nylon en een dubbele wasknijper.

De gebruiksaanwijzing:

stel de meter in op bv. Oranje.

We krijgen een lage toon voor $<10\text{ Kohm}$ en een hoge toon voor $>100\text{ kOhm}$ en "we are the champions, of een wisselende toon" voor een oranje weerstand.

De universele componenten tester.

Ook hier staat weer mechanisch prutswerk centraal.

We willen een ZIF-socket met minstens 4 poten.

We willen ook een display, zeg qvga (12 regels van 40 karakters) met grafische mode. Elektor heeft er een.

Nu gaan we aan het meten. Denk aan de transistors om de gedachten te bepalen (r,c,l is een beetje flauw : Elektor heeft een precisie LCR-meter, maar het is in de praktijk veel belangrijker meer dingen te weten met 10% precisie zoals een versterkingsfactor).

Het vervelendste is altijd om uit te vinden wat de basis is. Dit wordt met automatisch schakelen geregeld (bij voorkeur met relais).

Dan komt er iets uit als:

- aansluitplaatje
- transistor Ge
- versterking 56
- lekstroom $70\text{ }\mu\text{A}$
- doorslag 31 V
- warmtegeleiding 150 K/W
- verzadiging 143 mV
- V_{be} 180 mV

of

Diac Doorslag 8 V

De bedoeling is dat dit oneindig uitbreidbaar is door opzetstukjes en een interne spanningsbron die eventueel ook honderden volts kan leveren.

Dan kan men bijv ook een onbekende buis testen:

- buis EF80 slechte kwaliteit
- gloeidraad tussen 4 en 5, 300mA 12 V
- anode 9
- kathode 12/4
- rooster 3 8 9

OCR

Bij de boekenlezer hoort een OCR. De bestaande OCR's zijn niet goed.

Ik wil een OCR op basis van het herkennen van glyphen op een aanzienlijk hoeveelheid gegevens (boeken van meer dan tien bladzijden).

Het begint met het herkennen van regels, dan verdelen van de regels in blokjes. De blokjes worden vergeleken met de glyphen in vector vorm, zodat een tilt verrekend kan worden. Dan begint de KI. Aan de hand van woordenboeken en waarschijnlijkheden wordt de tekst gereconstrueerd, waarmee ook de glyphen bevestigd worden. Uiteindelijk wil ik een systeem dat zo zeker van zichzelf is, dat het typefouten kan identificeren eerder dan signaleren. Een betere job dan wat Google nu doet moet mogelijk zijn.

Voor Albert Nijhof: de glyph herkenner moet het ook doen op bladmuziek, althans het onderste niveau.

Het paranoia toetsenbord.

Niemand hoort zijn wachtwoorden te moeten toevertrouwen aan zijn *Ipod of aan Google* in de cloud. Daarom wil ik de wachtwoorden in mijn toetsenbord dat met een PS2-connector verbonden is met mijn computer.

Het idee is dat het wachtwoord het laatst wordt ingevuld zodat het toetsenbord er onmiddellijk een carriage return achter aan kan sturen. Het startpunt is zo'n onverwoestbaar IBM toetsenbord.

De toetsenbord processor word vervangen en een LCD schermpje toegevoegd in plaats van de status leds. Met een echte mechanische schakelaar kiezen we tussen intern en extern. Extern is normaal gebruik. Intern stuurt 1 van 12 wachtwoorden met de 12 functie-toetsen.

Intern is de communicatie. Bij opstarten moet je eenmaal het master password intikken en dan overgaan naar externe mode. De passwords worden natuurlijk geëncrypt opgeslagen. Een sd-kaartje kan gebruikt worden voor een backup. Als je het master password vergeet ben je verloren, maar dat is inherent aan zulke systemen.

Alle floppies op een kaart.

Dit is een hardware project. We maken een kaart die op een floppy drive connector kan worden aangesloten.

Hierin een sd-kaart van 32 Gbyte met een kopie van alle floppies die je voor de laatste keer met veel moeite gelezen hebt. Dit is onderverdeeld in gebieden van 2 Mbyte, genoeg voor floppies tot 1.44 Mbyte. Als je meer dan 16000 floppies wil archiveren dan heb je een extra sd-kaart nodig.

Het protocol is uiterst simpel (alles 64 bit).

Elke 2 Mbyte bestaat uit:

- sd-sector 0 : tekst, gevolgd door x keer:
- sd-sector 1: fp-sector grootte ($1 = 2 \dots 7 = 128 \ 10 = 1024 \dots$)
 - 2: sector #
 - 3: cyl #
 - 4: track #

sd-sectors zoveel als nodig voor de in sd-sector 0 aangeven fp-sectors.
(Dus elke fp-sector kost minimaal een sd-sector alleen al aan header)

De sectors staan in willekeurige volgorde. Sectors mogen ontbreken.
Het is niet verplicht dat alle sectoren van een floppy even groot zijn, maar
beveiligingen waar 2 sectoren hetzelfde sect/cyl/track nummer hebben kunnen niet.
Een sd-sector van 0 betekent: niets.

Het gebruik is als volgt:

Stel met het duimwiel of de roterende schakelaar de juiste floppy-soort in. De naam
verschijnt in het display. Schakel om.

Je computer kan nu de betreffende floppy lezen.

P.S. dit is archiveren. Het ding kan niet schrijven op de pseudo floppies.

P.P.S.S. We kunnen ook nog een knopje toevoegen: emuleer floppy-langzaamheid.

FIETS

En dan kunnen we altijd nog het volgende:

In de tachtiger jaren hebben we de FIETS processor verzonnen, tot en met een
emulatie in Forth.

Project: bouw dit in FPGA hardware.

FIETS 2

De GA144 van Chuck Moore is geschikt voor de ideale fiets computer. Hermetisch
gesloten roestvrij staal met een venster voor lcd en een lichtcel. Een processor kijkt
of de lichtcel genoeg energie heeft verzameld om de lading te transporteren naar
de goldcap. Een processor heeft een oscilator die beïnvloed wordt door een spaak.
Hiermee kunnen we spaken tellen en de rest is software. (O ja en we moeten een
horloge kristal aan kunnen sluiten, dat is ook bekend.)

Ook iets te melden?

Stuur uw ideeën, programma's of projecten naar de redactie,
zodat anderen daar ook kennis van kunnen nemen.

Bijdragen liefst per E-mail, Uiterlijk 2 weken voor de bijeenkomst, naar
f.l.van.der.markt@kader.hcc.nl



HCC!Forth

website van HCC!Forth: <http://www.forth.hcc.nl>