



Programma HCC!Forth komende zaterdag

**Zaterdag 12 augustus 2017 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 Voortgang van de werkgroepen, door Albert van der Horst.

11:15 Tiny Basic in Forth, door Coos Haak

12:30 Pauze

13:00 Stemming van muziek en voorbeelden in noForth en
idee over compiler-only woorden, door Albert Nijhof

14:15 Plan voor nieuwe werkgroep 'toepassingen', door Willem Ouwerkerk

15:00 Sluiting.

Tot ziens



Zuiderkapel Boslaan 1 Bilthoven

Tiny Basic in Forth

Coos Haak

Coos zal Tiny Basic in Forth demonstreren.

Dit is een minimale basicversie die al in 1976 ontworpen is voor een 8080 processor en destijds kon draaien in 3 KB Ram (Wikipedia).

Voor Forth zijn er ook meerdere Tiny-Basic's gemaakt die vaak meer mogelijkheden hebben, zoals arrays en FOR..NEXT.

Omdat dit een eenvoudige taal is, is het gemakkelijker te begrijpen hoe dit onder Forth werkt.

Coos zal dit demonstreren met een oude Forth versie die draait onder een 8086 processor in een virtual box onder Windows of Linux.

De Backus-Naur grammatica van de eerste Tiny Basic staat hier onder (Wikipedia). Daarin valt bv. op dat er slechts 26 variabelen zijn genaamd 'A' t/m 'Z'.

De ϵ duidt een lege plek aan, string is een tekst tussen accentjes, CR is de carriage-return en het '*' achteraan de regel betekent nul of meer herhalingen.

```
line ::= number statement CR | statement CR
statement ::= PRINT expr-list
              IF expression relop expression THEN statement
              GOTO expression
              INPUT var-list
              LET var = expression
              GOSUB expression
              RETURN
              CLEAR
              LIST
              RUN
              END

expr-list ::= (string|expression) (, (string|expression) )*
var-list  ::= var (, var)*
expression ::= (+|-| $\epsilon$ ) term ((+|-) term)*
term       ::= factor ((*|/) factor)*
factor     ::= var | number | (expression)
var        ::= A | B | C ... | Y | Z
number     ::= digit digit*
digit      ::= 0 | 1 | 2 | 3 | ... | 8 | 9
relop      ::= < (>|=| $\epsilon$ ) | > (<|=| $\epsilon$ ) | =
```

Een simpel voorbeeldprogramma in deze Tiny Basic:

```
10 INPUT X
   IF X>99 THEN GOTO 60
   LET Y = 59 + (X * 43) / 7
   PRINT X , Y
   PRINT X , " TEST "
   GOTO 10
60 END
RUN
```

Stemming van muziek en voorbeelden in noForth

Albert Nijhof

--- Stemming, over toonhoogte en frekwentie

Als je een programma wilt maken waarmee je melodietjes kunt spelen, bijvoorbeeld met noForth op de Launchpad, moet je natuurlijk iets van toonfrekwenties weten. Die kun je vast wel ergens van internet plukken, maar je kunt ze eenvoudig zelf berekenen, als je weet "hoe het zit".

"Een zuivere toon op een contrabas is puur toeval" schijnt een gevleugelde uitspraak van Brahms z'n vader te zijn. Dat zegt iets over de contrabas maar meer over de speler. Tegenwoordig zijn er contrabasspelers waar dat beslist niet meer voor geldt. Een zuiver gestemde gitaar, viool of piano daarentegen is geen puur toeval, maar is gewoon onmogelijk en dat heeft niets met de speler te maken.

"Hoe dat zit" zal ik op de komende bijeenkomst uit de doeken trachten te doen.

--- LIT en LITERAL in één woord

Als tweede onderwerp (helemaal los van het vorige) heb ik een ideetje over LIT, DOCOL, 0BRANCH en soortgenoten. Een ideetje dat ik graag aan jullie wil voorleggen omdat ik vind dat het zó voor de hand ligt, dat ik mij erover verbaas dat ik daar niet dertig jaar eerder opgekomen ben. Het gaat om de woordjes die een programmeur nooit expliciet gebruikt, maar die uitsluitend impliciet via compilerwoorden en definiërende woorden in een programma terecht komen.

Plan voor nieuwe werkgroep 'toepassingen'

Willem Ouwerkerk

Ik zou graag een nieuwe werkgroep oprichten, die als doel heeft toepassingen met Forth te ontwikkelen. We hebben erg veel kennis binnen de groep, maar doen daar weinig mee, dat was in het verleden niet altijd zo! We hebben met onze club meerdere malen prijzen gewonnen bij de HCC.

In mijn eentje ontwerp ik wel wat, maar het zou zo veel levendiger worden als we als groep de ideeën en kennis zouden combineren. Daar komen vanzelf weer leuke lezingen van. Let wel, ik wil niet mijn eigen ideeën opdringen!!

We hebben Teun Liedorp en Jan van Kleef voor de specialistische technische kennis, mijzelf voor printjes en drivers.

Daarnaast is er voldoende programmeerkennis met Forth bij verschillende mensen.

Er zijn al wat voorbeelden opgekomen:

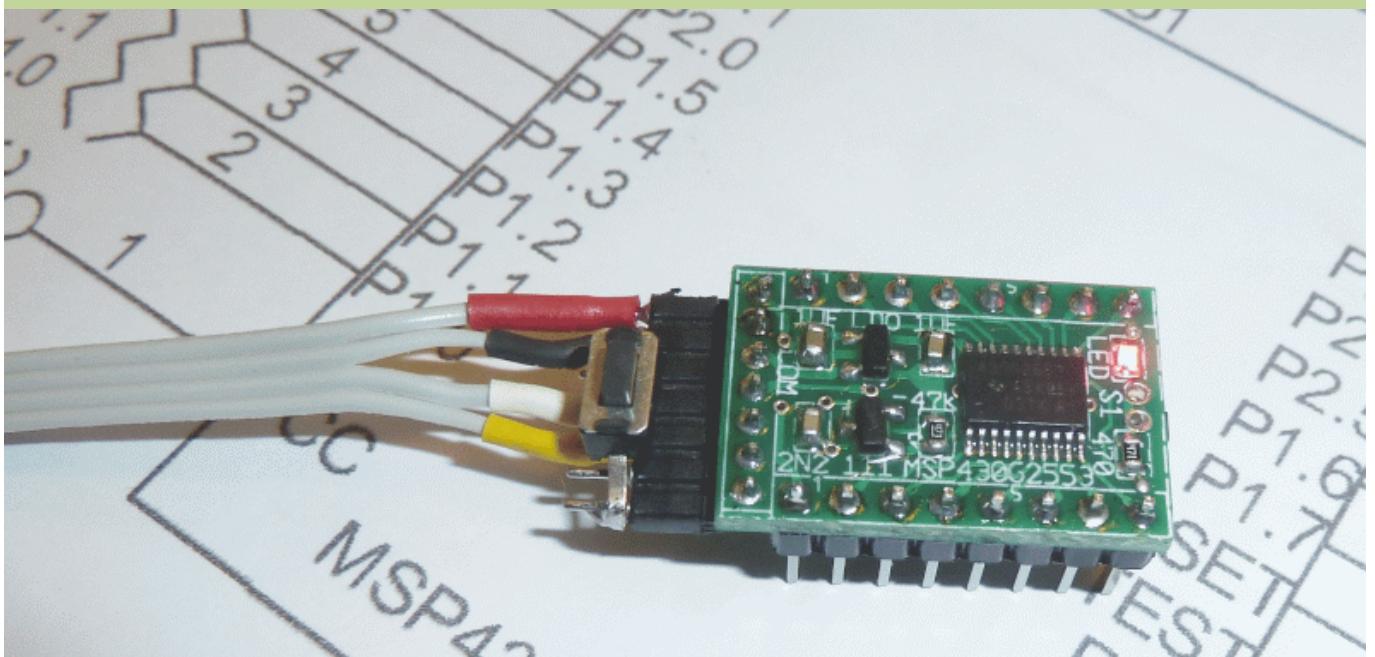
- * Applicaties met "energy harvesting" ofwel omgevings- energie (Leon)
- * CO2 besparen bv. universele standby
- * Digitale servo ombouw set voor analoge servo's
- * Ideeën en/of wensen uit onze groep zelf

Dat zou mogelijke nieuwe Egel voorbeelden op kunnen leveren of/en gebruik maken van de daar opgeslagen kennis. Maar het draait natuurlijk niet alleen om noForth en Egel.

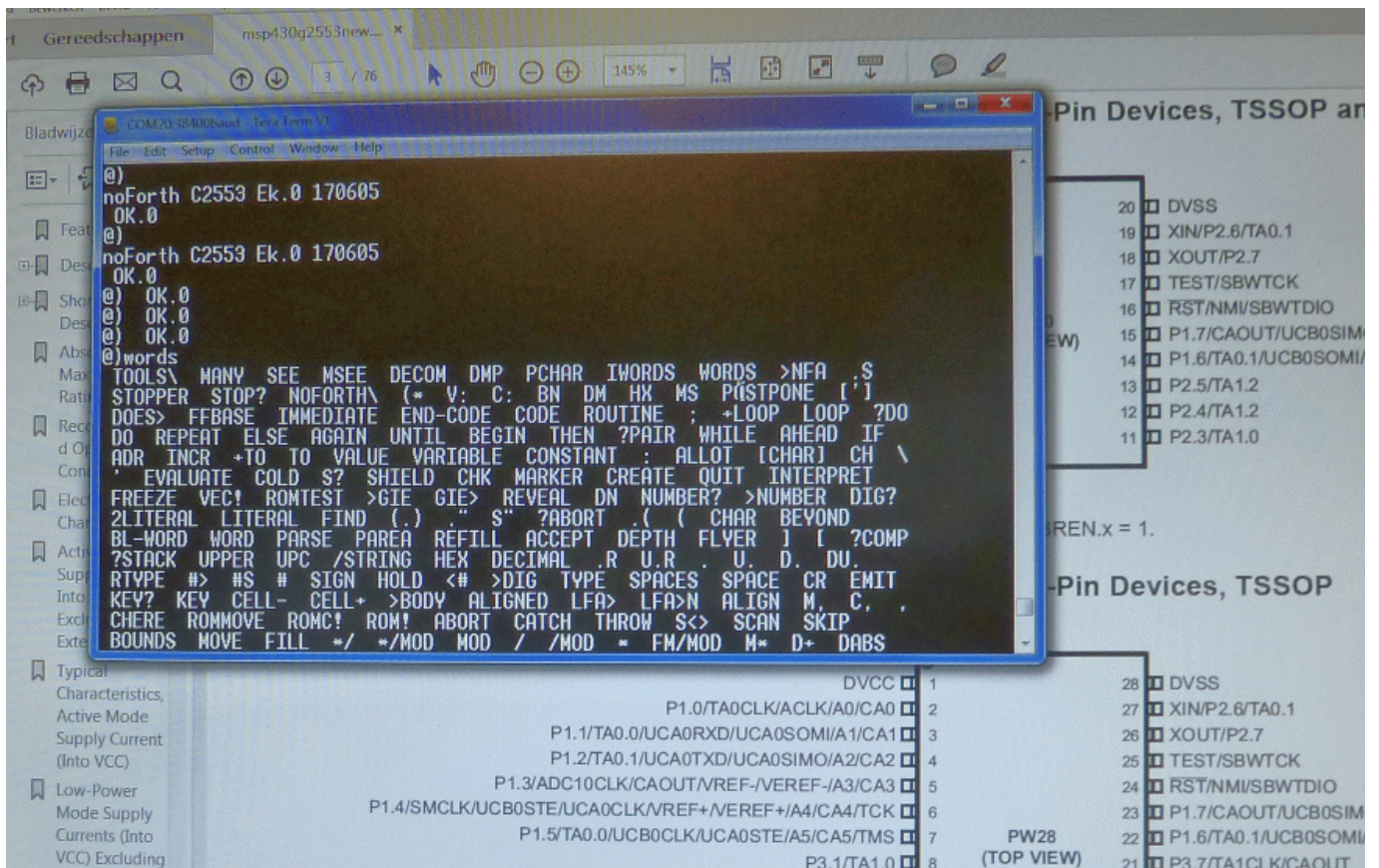
Als voorbeeld mijn digitale servo ontwerp als PDF bij deze mailing (Digital servo vsn 003a.pdf).

De Micro-Launchpad werkt!

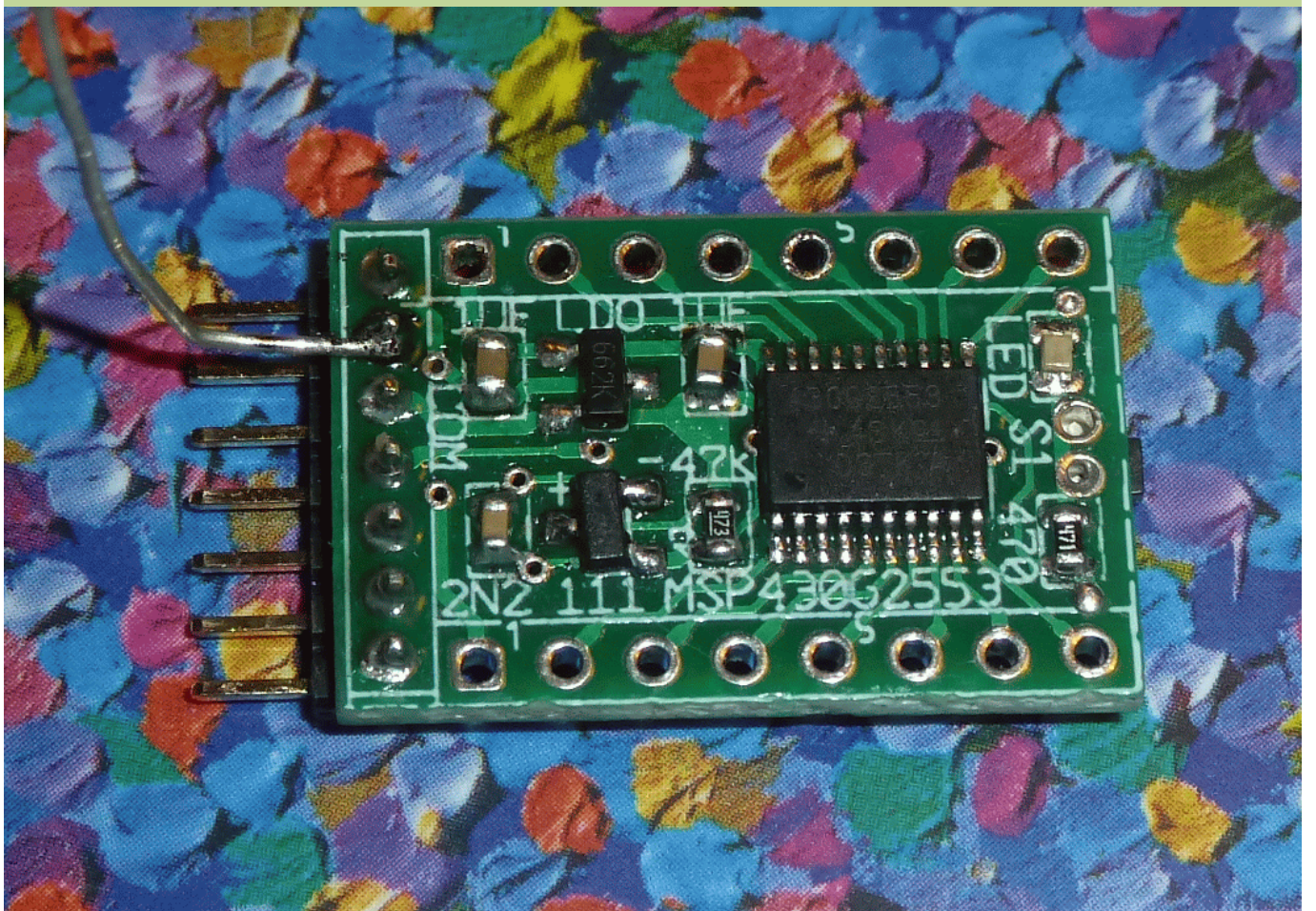
Willem Ouwerkerk mailde enkele plaatjes van een werkende Micro-Launchpad, een mooi succes!



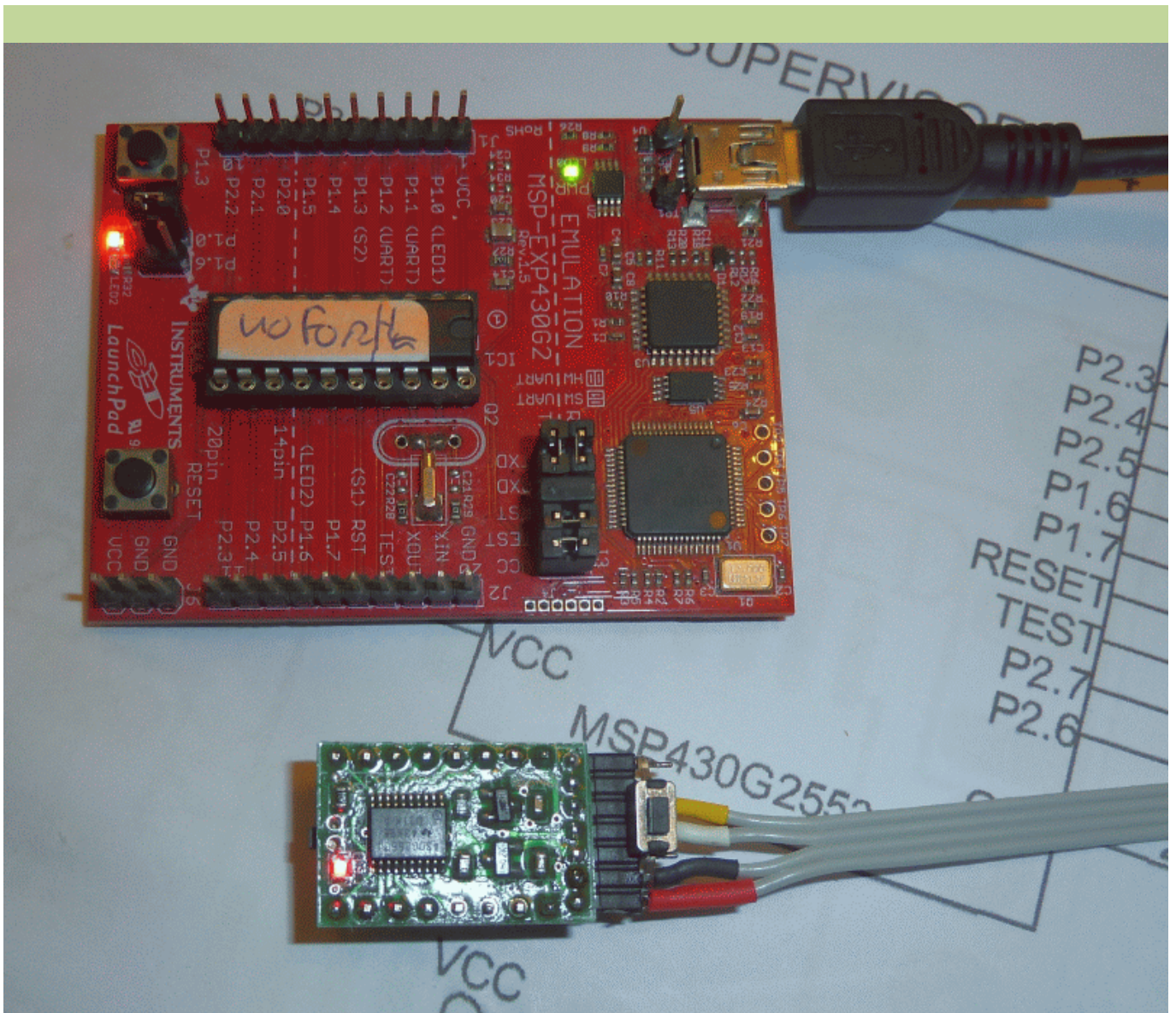
Alive and kicking



First boot of noForth C2553 Ek.0 170605



First prototype



Launchpad versus Micro-Launchpad

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 1 week voor de bijeenkomst, naar f.i.van.der.markt@kader.hcc.nl



HCC!Forth

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