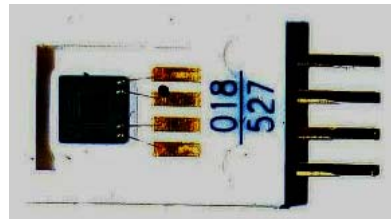


Agenda

- Zaterdag 6 oktober Bijeenkomst Hooglanderveen.
- Zaterdag 3 november Roborama wedstrijd Hooglanderveen.
- Zaterdag 1 december Bijeenkomst Hooglanderveen.

De bijeenkomsten te Hooglanderveen worden gehouden in Dorpshuis "de Dissel" Disselplein 6 3829 MD te Hooglanderveen. Route beschrijvingen op onze website of op die van onze voorzitter Bert Buiskool: www.robot.buiskool.net. PS het "tunneltje" in Hooglanderveen is weer open!

Voti
webshop
www.voti.nl

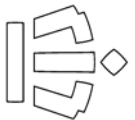


Een (lucht) druksensor voor relatieve druk. Helaas weinig gegevens bekend. Misschien iets voor duikboten? Normaal E 4.13, nu voor E 2.00.

<http://www.voti.nl/winkel/p/IC-PRES-01.html>
vermeldt "RoboBits aanbieding 9" - pas geld overmaken als u het verzoek tot betalen krijgt met het aangepaste bedrag - geldig tot de volgende RoboBits uitkomt

ROBO- BITS-38

Jaargang 10, nummer 3, september 2007



TNT Post
Port betaald
Port Payé
Pays-Bas

Afz. hcc Robotica gg, p.a. Henk de Gans, Anjerlaan 3, 3871 ev Hoevelaken.

Home Nieuws Links Contact Us Projecten search...

NEWSFLASH
HCCrobotica Roborama 2007 inschrijving -
Aanmelding Roborama 2007

WHO'S ONLINE
We have 1 guest online

MAIN MENU
Home
Nieuws
Links
Contact Us
Search
Projecten
FAQs
Administrator

LOGIN FORM
sername
password
Remember me
Login
Lost Password? > account yet? Register

LATEST NEWS
• RoboRama 4 nov 2006 uitslagen
• Sumobot Proto
• Sumobot Project 2007
• Welkom bij de HCCRobotica
• Route beschrijving naar PCGG Thales te Hengelo

POPULAR
• Sumobot Proto
• Sumobot Project 2007
• Welkom bij de HCCRobotica
• Route beschrijving naar PCGG Thales te Hengelo
• RoboRama 4 nov 2006 uitslagen

Home
Welkom bij de HCCRobotica
Written by Web Master
Saturday, 12 June 2004
Welkom op de Website van de HCCRobotica
Last Updated (Thursday, 20 September 2007)
[Read more...](#)

(C) 2007 HCCRobotica
Joomla! is Free Software released under the GNU/GPL License.

WWW.HCCROBOTICA.NL
WWW.HCCROBOTICA.NL

hcc[!]robotica

De Robobits is een uitgave van de hcc!robotica gebruikers groep, en wordt vier keer per jaar toegezonden aan de leden. De oplage is ongeveer 550 exemplaren. hcc!robotica is een onderdeel van de hcc! (hobby computer club), een vereniging van bijna 170.000 leden.

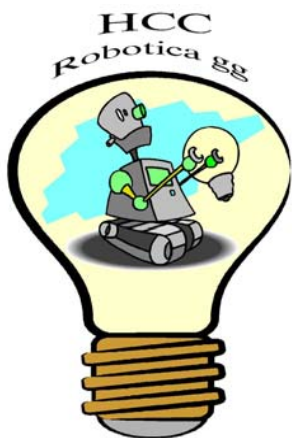
Redactie adres: H.J. de Gans, Anjerlaan 3, 3871EV Hoevelaken.
 henkdegans@kpnplanet.nl Tekst aanleveren in WORD of platte tekst in ASCII. Afbeeldingen los er bij in JPG, GIF of BMP formaat.

Dagelijks bestuur:

Voorzitter: B.T.J.A.Buiskool(Bert), robot@buiskool.net
 Technisch adviseur: Ing.H.M.A.van Bodegom(Henny) ing.h.m.a.van.bodegom@hccnet.nl
 Secretaris: A.J.Janssen (Lex) lex.janssen@hccnet.nl
 Penningmeester H.J. de Gans(Henk) henkdegans@kpnplanet.nl
 Lid: P.Smits(Paul) psmits.1@hccnet.nl
 Lid: W.C.de Boer (Wim) wim.deboer@nl.thalesgroup.com

inhouds opgave:

- Bladz. 3 Redactie.
- Bladz. 4 In de ban van de ring door Ed!
- Bladz. 7 Verslag open CNC dag, door Bas!
- Bladz. 10 Gelezen op de maillist!
- Bladz. 11 Multitasking door Joep
- Bladz. 22 Beeldverslag bijeenkomst Hengelo.
- Bladz. 23 Inschrijving Roborama November 2007!
- Bladz. 23 WWW.HCCROBOTICA.NL onze nieuwe website!
- Bladz. 24 Agenda.



Roborama november 2007

Zaterdag 3 november 2007 zullen wij voor de tweede keer een ROBORAMA wedstrijd organiseren. Ook dit keer zullen de opdrachten zijn:

- Heen en Weer!
- T-Tijd!
- Lijnvolgen!
- Blikken!

Ook zullen de mini sumo robots weer de strijd met elkaar aangaan, en gezien de grote belangstelling voor het mini sumo project, mogen we zeker weer veel deelnemers verwachten!

Voor inschrijven kunt u terecht op onze geheel vernieuwde website:

www.hccrobotica.nl

Voor de reglementen en meer informatie, kunt u terecht op de website van RobotMC (in de toekomst ook op onze eigen site)

<http://wiki.robotmc.org>

Meer informatie kunt u ook krijgen bij Wim de Boer wim.deboer@nl.thalesgroup.com die ook deze keer de organisatie weer op zich wou nemen(waarvoor onze dank!). Bedenk dat ook deze keer weer, meedoen is veel belangrijker als winnen! De wedstrijd zal plaatsvinden op onze locatie in Hooglanderveen, en vangt aan om 10 uur.

nieuwe website!

We hebben er lang op gewacht, maar dank zij Lex en Wim, hebben we inmiddels de eerste aanzet voor een totaal nieuwe website, online staan!



Joomla!

...because open source matters

U ziet van de homepage een screendump op de omslag van deze robobits. De opmaak is gemaakt met JOOMLA, een open course content management systeem. Hiermee zal het mogelijk zijn dat meerdere personen de site zullen kunnen bijwerken en actueel houden! Een belangrijk gegeven waar onze oude site te kort in schoot de

laatste tijd. Vele handen maken ook hierin "licht werk" Wilt u meer weten of wilt u meehelpen de site uit te bouwen tot een "levend" stuk verenigingsleven, waar een ieder kennis kan vergaren maar ook kan delen, dan kunt u contact opnemen met Wim de Boer wim.deboer@nl.thalesgroup.com



REDACTIE

Op de voorpagina deze keer eens niet een robot, maar een screendump van de nieuwe website! We hebben lange tijd geen actuele en goede website gehad, maar nu ligt daar toch dankzij Lex en Wim een opzet voor onze nieuwe website op een eigen domein. Ook in deze robobits een "inside" beleving van een deelnemer aan het sumo project, een verslag van de succesvolle CNC dag en een mooi artikel van Joep over multitasking. Al met al een goed gevulde robobits lijkt me! Ik wens u allen veel leesplezier!

Henk de Gans

maillist!

We krijgen altijd veel vragen over HOE aan en af te melden bij de hccrobotica maillist.

Aanmelden:

Stuur een mail naar : hccrobotica-subscribe@buiskool.net

Afmelden:

Stuur een mail naar: hccrobotica-unsubscribe@buiskool.net

Je ontvangt na aan- of afmelden een mailtje ter bevestiging.

E-mail sturen naar de mailinglijst:

Stuur uw mail naar: hccrobotica@buiskool.net en alle deelnemers ontvangen uw mail !!!!

deadline deadline deadline deadline deadline deadline deadline deadline

De deadline voor kopie voor Robobits 39 die eind december 2007 zal uitkomen, is gesteld op dinsdag avond 11 december 2007!!!! Maar u hoeft natuurlijk niet zo lang te wachten, nu al insturen kan natuurlijk ook! Iedereen wil heel graag eens wat lezen over dat waar u mee bezig bent, en natuurlijk met onze hobby te maken heeft! Vooral van u, die niet in de gelegenheid bent onze bijeenkomsten te bezoeken! Zoals al eerder gezegd, hoeven het geen hoogdravende journalistieke meester werken te zijn(mag natuurlijk wel;-)). Gewoon een leuk stukje over uw creatie of iets dergelijks, of wat informatie over iets waarvan u gehoord of gelezen hebt enz. Ook zoek ik nog een boek bespreking(of tijdschrift).

deadline deadline deadline deadline deadline deadline deadline deadline

In de ban van de ring!

Het was inmiddels een kleine 20 jaar geleden dat de elektronica en computer hobby regelmatig werd beoefend. De soldeerbout werd hooguit nog een keer opgestookt om een kleine reparatie uit te voeren maar van enige opwinding bij het vooruitzicht weer te gaan solderen was al lang geen sprake meer. Kortom, de eens zo met plezier beoefende hobby was inmiddels weggezakt tot een historisch dieptepunt. Dat hopen we tenminste.

Ja, wel al die tijd HCC lid gebleven. In 1998 zelfs moedig opnieuw aangemeld bij Elektuur, in 2003 opgegeven voor de HCC Robotica groep en omstreeks dezelfde tijd een BoeBot van Parallax gekocht. Het mocht allemaal niet baten, het oude vuur was blijkbaar gedoofd en niet meer aan te wakkeren totdat ...ik in Robobits-34 een verhaal las van ene Hinnie van Bodegom over een Sumo voor zelfbouw. Wel tien keer geleden, met mijn zoon besproken en toch nog heel even getwijfeld of ik zou happen maar je raadt het al, het was nu of nooit en dus heb ik mij aangemeld om mee te doen en een Sumo te bouwen.

Ik wou de trein niet missen en zag mijn kans schoon om de draad die lang geleden uit de vingers was geglipt weer op te pakken. Het werken met anderen in een groep tijdens opkomsten maar ook via mail contact houden en overleggen was net het duwtje in de rug dat ik nodig had om weer van start te gaan. Thuis was mijn knutselhoek weer snel ingericht en het was en is nu net alsof ik nooit gestopt ben behalve dan dat er uiteraard nieuwe hardware gebruikt wordt. Had ik vroeger te doen met een 6800 en Z80, de Sumo werkt met een ATmega32 AVR processor en dat is net wat anders.

Sinds eind vorig jaar bezoek ik iedere maand de HCC Robotica bijeenkomst in Hooglanderveen. Een beetje uit de buurt maar het loont wel de moeite. Er wordt sinds februari door 16 deelnemers druk gewerkt om ieder een robot in elkaar te krijgen. Dankzij de goede voorbereidingen van het project is er voor alle deelnemers de gelegenheid om het project met succes af te ronden. De organisatie heeft een schema opgesteld en levert de printen, behuizing en moeilijk verkrijgbare onderdelen aan. Tijdens de bijeenkomsten wordt er gebouwd, met elkaar overlegt en ook een beetje "afgekeken". De een is goed is het knutselen met de hardware, anderen ontpoppen zich als

En de twee routines:

```
//-----  
// SetDelay - SetDelay in microseconds, up to 104.8ms  
//-----  
// Returns a unsigned int, use CheckDelay to check if delay is reached  
//-----  
unsigned int SetDelay(long Delay)  
{ unsigned int Timer0;  
  
    Timer0 = (TMR0L + (((long) TMR0H) << 8));  
  
    Delay = (Delay * 10) >> 5; // from microseconds to n*3.2 uS  
  
    if (Delay > 32767) Delay = 32767; // clip to max  
  
    Timer0 += Delay;  
    return Timer0;  
}  
  
//-----  
// CheckDelay - Check if delay is reached  
//-----  
// Use SetDelay to calculate 'End'.  
// returns 1 if delay is reached, otherwise 0.  
//-----  
int CheckDelay(unsigned int End)  
{ unsigned int Timer0;  
  
    Timer0 = (TMR0L + (((long) TMR0H) << 8));  
    if (Timer0 < End) return 0; // Delay not reached  
  
    if (End > 32767) return 1; // Delay reached  
  
    if (Timer0 < (End + 32767)) return 1; // Delay reached  
  
    return 0; // Delay not reached (or a long time ago...)  
}
```

Bijeenkomst Hengelo

Op zaterdag 15 september was er voor de derde keer al weer, een bijeenkomst in Hengelo. Net als vorige keren wederom georganiseerd door Wim de Boer op uitnodiging van de PCGG van Thales. Ook deze keer was de opkomst weer uitstekend, en was er vooral veel jeugd! En evenals de vorige jaren was de entourage en catering weer perfect, mede door de inzet van Wim zijn vrouw! Onze dank gaat dan ook uit naar de organisatie en de PCGG van Thales voor hun gastvrijheid!

```

    Taak1state = 2
End If
Case 2
If Nieuwestate1 = 1 Then
    Led1 = 0
    Wachtijd1 = 500
End If
If Wachtijd1 < 1 Then
    Taak1state = 1
End If

Case else
    Taak1state = 1

End Select

```

End Sub

Bijlage 2:

Deze bijlage bevat een voorbeeld van een wachttijd die gerealiseerd wordt zonder interrupts. Om dit te realiseren wordt een timer gebruikt, die met een vaste frequentie oploopt. Als een bepaalde wachttijd gewenst is, wordt door de functie SetDelay() het eindtijd bepaald op basis van de wachttijd en de huidige waarde van de teller. Met de functie CheckDelay() wordt gecontroleerd of de eindtijd bereikt is. De C-code is gebruikt op een 18F482 op 40 MHz. De prescaler is zo ingesteld dat de tijd-stappen 3.2 microseconden zijn. De maximale vertraging die wordt ondersteund is 32768 stappen, ofwel $32768 * 3.2 = 104.8$ miliseconden.

De initialisatie van de timer gaat als volgt:

```

// init TMR0 for polled timing.
TOCON = 0x84; //set up timer0 - 16 bits & prescaler 1:32 (of osc/4)
//increment at 312.5 kHz (@40 MHz), 3.2uS, 32k = 0.1S

```

Gebruik van de wachttijd-routines:

```

unsigned int EndTime;
EndTime = SetDelay(10000); // wachttijd 10.000 microseconden

while (CheckDelay(EndTime) == 0) {
    // doe hier iets totdat de tijd voorbij is...
}

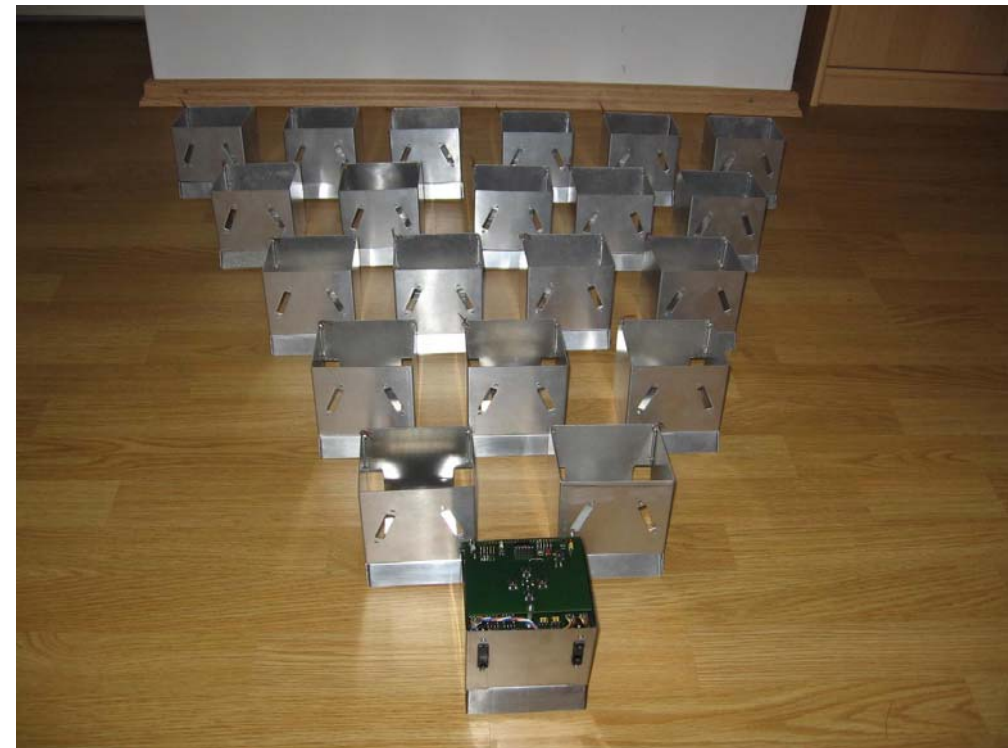
```

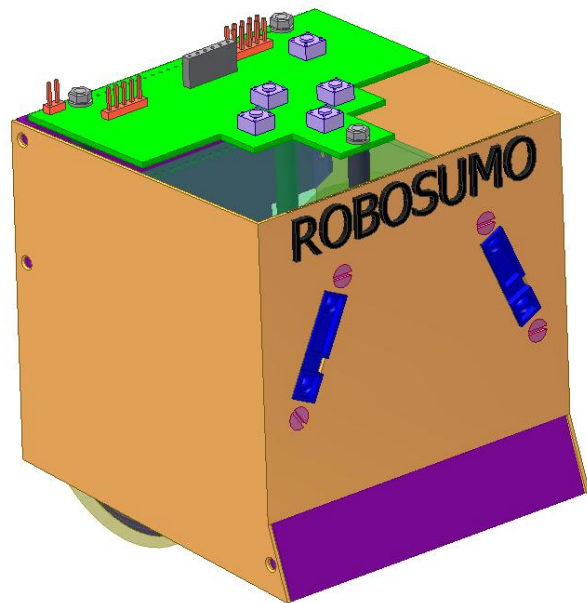
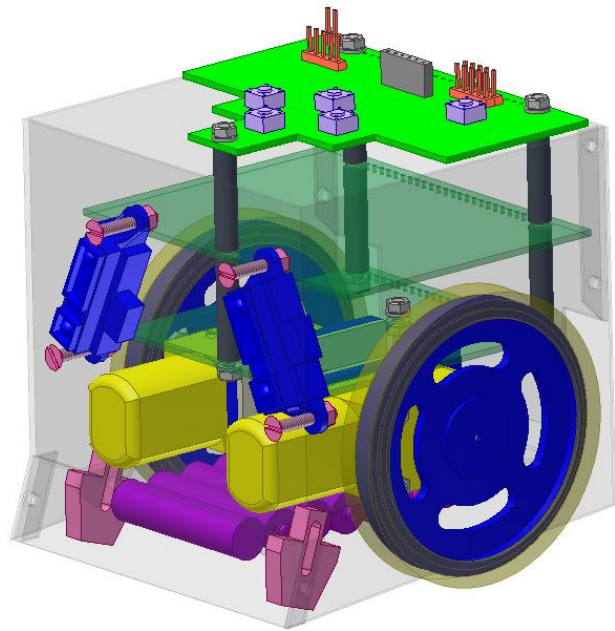
software meesters en daar gaan wij met z'n allen uiteraard ons voordeel mee doen.

Zoals het inmiddels wel duidelijk mag zijn, ik ben een enthousiast lid (geworden) en verheug mij om samen met deze club nog veel leuke dingen te leren en te doen. Een echte aanrader voor iedereen die de hobby een warm hart toedraagt en op zoek is naar een nieuwe uitdaging. De ultieme doelstelling is om tijdens Roborama wedstrijd van 3 november a.s. goed "uitgerust" in de ring te verschijnen. Daar gaan we voor. Hopelijk zijn jullie van de partij als deelnemer of om inspiratie op te doen. Tot ziens en .. wordt vervolgd.

Ed Buzzi

Red. na dit enthousiaste verhaal van Ed, een beeldverslag van Aloys! Aloys heeft ondermeer de body's voor iedereen geproduceerd, en zich over nog meer hardware ontfermt!





```
'multitasking
'quartzfrequentie= 80000000/64= 125000Hz
'overloopfrequentie = 31250/(256-x) = 1000Hz
'x = 193
```

```
$regfile = "m88def.dat"
$crystal = 8000000
Declare Sub Doetaak1()
```

```
Config Pinb.6 = Output
Led1 Alias Portb.6
```

```
Config Timer0 = Timer , Prescale = 64
On Timer0 Timer_irq
Enable Timer0
Enable Interrupts
```

```
Dim Wachtijd1 As Integer
Wachtijd1 = 500
Dim Taak1state As Byte
Taak1state = 1
Dim Taak1vorigestate As Byte
Taak1vorigestate = 0
Dim Nieuwestate1 As Bit
Nieuwestate1 = 0
```

```
Do
  Call Doetaak1()
Loop
```

```
Timer_irq:
  Timer0 = 193
  If Wachtijd1 > 0 Then
    Wachtijd1 = Wachtijd1 - 1
  End If
Return
```

```
Sub Doetaak1()
```

```
  Nieuwestate1 = 0
  If Taak1state <> Taak1vorigestate Then
    Taak1vorigestate = Taak1state
    Nieuwestate1 = 1
  End If
```

```
  Select Case Taak1state
  Case 1
    If Nieuwestate1 = 1 Then
      Led1 = 1
      Wachtijd1 = 500
    End If
    If Wachtijd1 < 1 Then
```

In dit voorbeeld is er geen vaste volgorde van states meer, maar de volgorde wordt door de omgeving beïnvloed. Zo kan de state na 'volg wand' zowel 'draai links' als 'draai rechts' zijn, afhankelijk van de sensorwaarde. En of na 'draai links' de robot stopt of doorgaat (resp. 'rust' of 'volg wand'), is afhankelijk van de waarde van de variabele hoekteller.

De daadwerkelijke implementatie wordt overgelaten aan de lezer. En mogelijk blijkt bij het testen, dat de draai naar rechts niet helemaal goed gaat. Als alternatief kan een bocht naar rechts worden geprobeerd (aanpassen van de state-actie bij 'draai rechts'). En als dat niet werkt, kan na 'SensorR > 100' een bepaalde tijd rechtdoor worden gereden (state toevoegen), voordat de draai naar rechts wordt gemaakt.

Natuurlijk kan ook Roborama heen-en-weer uitgewerkt worden op basis van een state machine. De states die je hiervoor kunt gebruiken zijn: 'rust', 'volg wand links', 'draai 180 graden' en 'volg wand rechts'.

Om de lezer op weg te helpen is in bijlage 1 het voorbeeld van de knipperende led op meer gestructureerde wijze uitgewerkt in bascom (met dank aan Rien van Harmelen voor het programma).

Bijlage:1

De belangrijkste verschillen met de voorbeelden uit het artikel:

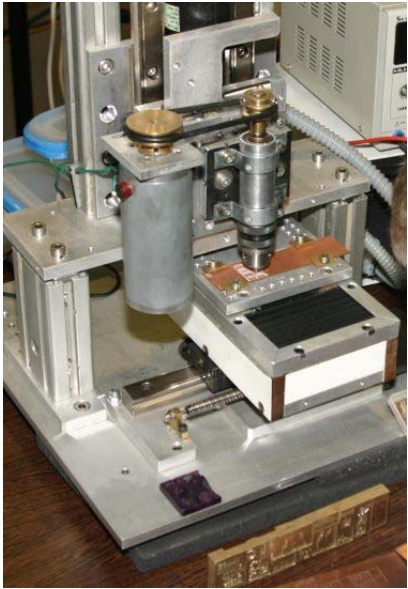
- o Het is geen code-fragment maar een compleet werkend programma.
- o De timing wordt bepaald door tellers die in de achtergrond worden verlaagd in de interrupt service routine.
- o Vooraf in de state machine wordt bepaald of de state entry action uitgevoerd moet worden. In dit geval wordt NieuweState1 op 1 gezet. In de state's zelf hoeft alleen maar gecontroleerd te worden of deze variabele 1 is. En de vlag kan tevens worden gebruikt om de state-overgangen te printen voor debug!
- o Ongeldige states worden afgevangen door 'case else'. In dit geval wordt de FSM weer 'op het goede spoor' gezet en het is verstandig om hier ook een melding toe te voegen, waarin de (ongeldige) state-waarde is opgenomen.

open CNC dag groot succes!



Op zaterdag 1 september hebben Lex en Bas i.s.m. met CNCzone.nl (Jan en Bas) een CNC dag georganiseerd. Toen Lex en ik s'morgens uit Utrecht vertrokken, richting Hooglanderveen, hadden we geen flauw idee wat ons deze dag allemaal zou overkomen, laat staan dat we een inschatting konden maken over hoeveel mensen er zouden komen en hoeveel machines er aanwezig zouden zijn. Binnen de HCC!Robotica is de belangstelling niet erg groot, zo dachten wij en zou de meeste respons uit de CNCzone hoek moeten komen. Degenen die rond een uur of 12 uur in het buurthuis de Dissel aanwezig waren troffen een volle zaal. Onze oproepen via de website, Robobits 37 en het forum van CNCzone hebben naast 15 mensen, die hun machine hadden meegenomen, ook een 100 tal belangstellenden aangetrokken. Een uitstekend resultaat dus! Het was alleen even schrikken dat we feitelijk maar net met het aantal tafels in de zaal uitkwamen. We moesten zelfs nog een plekje zoeken voor Roger Jonkers, SMD printjes, een tafeltje zoeken. Gelukkig werkte iedereen mee en hadden snel een goede oplossing gevonden.

Een korte impressie?



Het is haast onmogelijk om hier een beschrijving te geven van wat er allemaal te zien was. Maar als je een rondje liep, daar ben ik zelf geen eens aan toe gekomen, kon je bij iedere machinegebruiker wel wat opsteken. Zoals bij Robotica zijn CNC gebruikers ook eenlingen en maken zij (soms) verrassend leuke dingen op (super) eenvoudige manieren. Zo stond er een man, Robert, die hele kleine SMD printjes maakte door met zijn CNC machientje het koper weg te krassen. Hij gebruikte hiervoor de stift van een afgebroken freesje of boor, waaraan hij een scherpe punt had geslepen. De motor diende alleen als spankop, maar draaide niet. De printjes, maar icoontjes zagen er zeer goed uit.

Een tweede zaak die we hier toch niet onvermeld kunnen laten is de MDF machine van onze vriend Martin de Roode. Ook hij liet zien dat je met een klein budget een CNC machine kan maken. Ik leg hier wel de nadruk op KAN. Je moet toch een aantal zaken zeer goed uitvoeren om er bruikbare dingen mee te kunnen maken. Dat Martin hiertoe instaat is zien wij telkens weer. De MDF machine freest het Feyenoord logo. Een waarschuwing? Dat je moet weten waar je aan begint geldt voor een heleboel mensen of misschien wel voor ons allemaal. Zo spraken we een van de kopers van een Mark2, die de machine al ruim twee jaar ongebruikt in een hoekje had staan, omdat hij de machine niet durft aan te sluiten!! Maar hij was niet de enige die dit soort problemen heeft bij aanvang. Velen schaffen een machine (compleet met software) aan, maar hebben geen idee hoe ze een vierkantje moeten frezen. Het "compleet met software" betekent dat

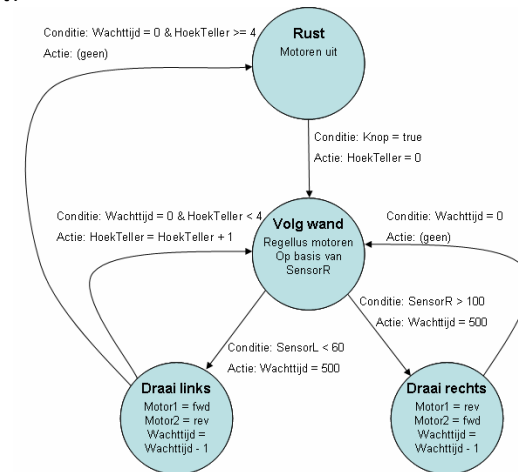


Om deze opgave tot een goed einde te brengen, hanteren we de volgende aanpak:

1. Bij het opstarten komt de robot in de rust-stand.
2. Zodra op de startknop wordt gedrukt, gaat de robot de rechter wand volgen met de sensor rechts-voor. Op deze manier rijdt de robot van A naar B (pijl 1).
3. De robot 'ziet' dat hij in B is gearriveerd doordat de linker sensor de achterwand ziet. Op dat punt draait de robot ongeveer 90 graden naar links, door de motoren een vastgestelde tijd aan te sturen.
4. Stappen 2 en 3 herhalen zich op pijl 2.
5. Stap 2 (volgen van de wand) wordt opnieuw gebruikt bij pijl 3.
6. Aan het einde van pijl 3 ziet de sensor rechts-voor ineens geen wand meer. Dit is het signaal om 90 graden naar rechts te draaien.
7. Stappen 2 en 3 worden gebruikt voor pijlen 4 en 5
8. Pijl 6 wordt afgehandeld met stappen 3 en 6.
9. Tot slot wordt pijl 7 afgehandeld met stappen 2 en 3.
10. Het einde van pijl 7 was de 5^e draai naar links. Op basis hiervan weten we dat de opgave is afgerond en we moeten stoppen.

Uit bovenstaande aanpak kunnen we de volgende states afleiden: 'rust', 'volg wand', 'draai links' en 'draai rechts'.

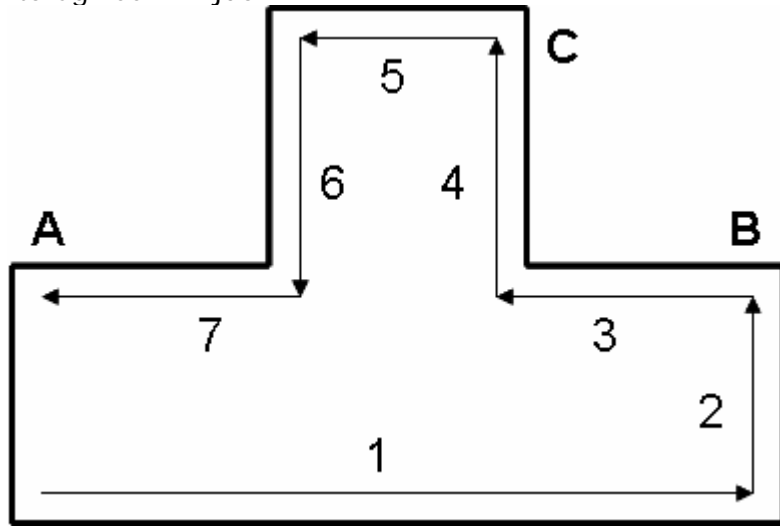
Daarnaast worden de belangrijkste condities voor overgang genoemd: druk op de knop, sensor links geeft signaal (output < 60 cm), sensor rechts ziet geen wand (output > 100 cm). In een state diagram ziet dit er als volgt uit:



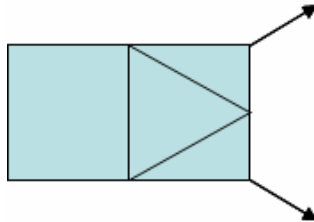
Transition action

Overgangactie - deze actie wordt eenmalig uitgevoerd als we van de ene state overgaan naar de andere state. Het instellen van de wachttijd is hiervan een voorbeeld. Transition actions zijn eenvoudig toe te voegen aan het if-then statement waarin de state wordt gewijzigd.

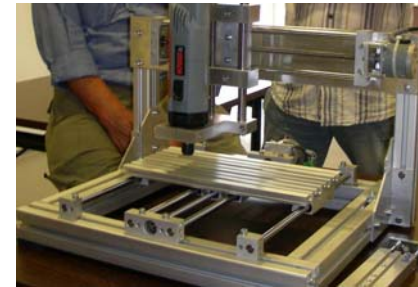
Voor het volgend voorbeeld nemen we de roborama opgave T-tijd als uitgangspunt. Bij deze opgave moet de robot van gebied A (links-onder in onderstaande tekening) naar gebieden B (rechts onder), C (bovenaan) en weer terug naar A rijden.



De robot die we gaan gebruiken in dit voorbeeld heeft twee sensoren, die onder 45 graden naar resp. links-voor en rechts-voor de afstand meten. Als aandrijving wordt een 'differential drive' gebruikt (een motor per wiel) en natuurlijk heeft de robot een start-knop.



de leverancier daar een Lite versie van zijn besturingssysteem onder MSDOS bij doet, maar dat je meestal zelf nog even opzoek moet naar een goed tekenpakket, een programma dat je tekening om kan zetten in G-Code (geen HPGL).



De toekomst? Als vereniging kan je de komende jaren een aantal CNC cursussen geven over hoe teken ik een vierkantje, rondje, servo-wiel etc. en hoe maak ik daar bruikbare G-Code van, zodat ik deze kan frezen. Je ziet het gebruik van deze machines onder de hobbyisten steeds meer toenemen, maar dat de kennis (zeker in het begin)

minimaal is. Mogelijk kunnen we als gebruikersgroep hier iets meedoen.



Tot slot.

We hebben, i.s.m. cncone.nl, een prachtige open CNC dag georganiseerd, met daarin twee interessante lezingen. Maar ook alle mensen die hun machine hebben meegenomen en leveranciers, zoals Fa. TEBI, DamenCNC en USB CNC, hebben hun bijdrage geleverd om van deze dag een groot succes te maken. Hopelijk kunnen we volgend jaar weer zo'n dag organiseren.

Bas Boetekees

van de maillist!

Eind 2007 wordt het JAL boek "Kunstmatige Intelligentie, breng uw microcontroller tot leven" verwacht. Dit boek bevat ruim 20 bijzondere en spannende projecten over kunstmatige intelligentie en lerende machines, voor PIC microcontroller en PC. Leer hoe u een neurale netwerk in een microcontroller opzet, en hoe u het netwerk zelflerend kunt maken. Ontdek hoe u robots kunt kweken, en hoe u door verandering van een fitness functie een totaal ander gedrag krijgt. Merk hoe een PC programma uw zwakke punten vindt en deze in een spelletje meedogenloos uitbuit. Bouw een robot met een eigen wil, of laat een robot uw parket stofvrij houden.

Een paar voorbeeldprojecten uit het boek: een microcontroller die leert wat uw lievelingskleur is, een robot die door het huis zwerft op zoek naar iemand om mee te spelen, een gekweekt robot programma dat niet over een zwarte lijn kan, een microcontroller die een spel leert te spelen totdat u niet meer kunt winnen, een PC die zelf een microcontroller programmeert.

Tallose technieken uit de kunstmatige intelligentie komen in de projecten aan de orde zoals expert systeem, neurale netwerk, subsumptie, emergent gedrag, genetisch algoritme, cellulaire automaat en roulette hersenen. Ieder project bevat duidelijke instructies met schema's en foto's zodat u meteen aan de slag kunt. Overal vindt u ideeën en veel literatuur verwijzingen om nog veel meer projecten te kunnen maken. Zelfs als u alle projecten uit dit boek gemaakt hebt, zal het zodoende nog lang als ideeënboek en naslagwerk naast uw PC blijven staan.

Een uniek boek voor iedereen met interesse in kunstmatige intelligentie en lerende machines. Proeflezers: Joep Suijs en Stef Mientki.

Schrijf je in op de mailing list en maak kans op een gratis exemplaar van dit mooie boek, inclusief verzendkosten. Ga naar <http://members.home.nl/b.vandam/boek/index.html>, kies het juiste boek en volg de instructies.

Veel plezier,
Bert

Voordat we een iets complexer voorbeeld uitwerken gaan we eerst nog iets dieper in op het principe van de FSM. De letterlijke vertaling van Finite State Machine is Eindige Toestand Machine. Het gaat dus over een 'machine' die zich in verschillende (maar vooraf te bepalen, dus eindig aantal) toestanden kan bevinden. In ons voorbeeld zijn die toestanden 'led aan' en 'led uit' en in het diagram worden de toestanden weergegeven als bolletjes. Het aantal toestanden is eindig – bij het ontwerp bepaal je welke toestanden je nodig hebt en de machine zal zich altijd in één van die toestanden bevinden. De verandering van state (toestand) heet transition of overgang. Normaal gesproken hoort een transition bij een specifieke state en bestaat die uit een voorwaarde en een nieuwe state. Zo hoort de bovenste transition (1) in ons diagram bij de state 'led aan', is de voorwaarde 'wachtijd == 0' en de nieuwe state 'led uit'. In speciale gevallen is een transition los gekoppeld van een state. Zo kan bijvoorbeeld het indrukken van een stopknop of een fout bij de ontvangst van de seriële poort de state een bepaalde waarde geven, ongeacht de huidige waarde van de state.

Het hebben van een state machine is natuurlijk aardig, maar het gaat er om dat we hiermee de juiste acties kunnen ondernemen. We onderscheiden de volgende acties, op basis van het moment waarop deze plaatsvinden:

Entry action

Binnenkomstactie - deze actie wordt eenmalig wordt uitgevoerd als we de state binnenkomen. In het voorgaande programma zijn deze niet opgenomen, maar in het volgende voorbeeld wel.

State action

Toestandsactie - dit is een actie die wordt uitgevoerd zolang we in een bepaalde state zitten, eventueel afhankelijk van inputs (IO lijnen, andere variabelen etc). In ons voorbeeld is het aansturen van de led een 'state action'.

Exit action

Vertrekactie - deze actie wordt eenmalig uitgevoerd als een state verlaten, ongeacht naar welke state we toe gaan. In de praktijk wordt zo'n actie vaak als de hieronder beschreven 'transition action' opgenomen bij iedere transition.

```

    If (Wachttijd1 == 0) then
        WachtTijd1 = 500
        Taak1State = 2
    End if
End block

2 : block
    Led = false
    If (Wachttijd1 == 0) then
        WachtTijd1 = 500
        Taak1State = 1
    End if
End block
End procedure

```

En de aanroep vanuit de hoofdflus:

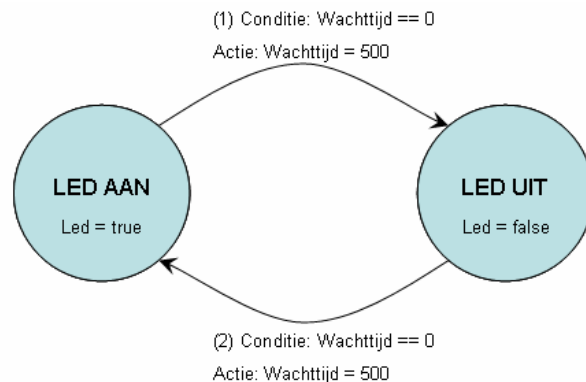
```

Forever loop
    DoeTaak1()
End loop

```

En hiermee hebben we ons doel bereikt: DoeTaak1() onthoudt zelf waar deze mee bezig, doordat de state wordt bewaard in een globale of static variabele. Daarnaast bepaalt de taak zelf de timing, in dit geval ondersteund door de timer ISR. Het enige wat we moeten doen is de taak regelmatig aanroepen uit de hoofdflus. In het bovenstaande voorbeeld zal dit duizenden keren per seconde gebeuren. Het is nu eenvoudig om een tweede procedure(te maken die een andere led bijvoorbeeld 600 ms inschakelt en vervolgens 350 ms uitschakelt.

Het knipperen van een led is eenvoudig, maar laat goed zien hoe state machines werken. Schematisch ziet het state diagram er als volgt uit:



Multitasking!

Door Joep Suijs

Een computer kan maar één ding tegelijk. Toch?

Op zich is dit waar, maar de meeste dingen die wij in een robotje van de controller vragen zijn zo gebeurd en de rest van de tijd staat de controller te wachten. Wat zou het mooi zijn als we die wachttijd kunnen benutten om iets anders te doen! In "grote" computers wordt dit geregeld door het besturingssysteem. Terwijl ik dit artikel zit te typen, wordt ook af en toe mijn mail gelezen en zo meteen krijg ik een melding van een afspraak die in de agenda staat. Voor iedere toepassing draait een programma en het besturingssysteem zorgt ervoor dat die om de beurt processortijd krijgen toegewezen. Deze vorm van multitasking door het operating system is een mooie oplossing voor het probleem, omdat je er als applicatieontwikkelaar niets voor hoeft te doen. Helaas maken de meeste robotjes gebruik van een microcontroller die te weinig geheugen heeft en ook wat voorzieningen mist om een besturings systeem te draaien waarmee we multitasking krijgen. We kunnen maar één programma draaien, eventueel ondersteund met een interrupt routine. Hoe gaan we dan meerdere dingen tegelijk doen? Om daarop een antwoord te geven, gaan we eerst eens kijken hoe we nu dingen doen. Stel dat we een ledje hebben met een frequentie van 1 Hz laten knipperen. In JAL(Zie de JALv2 homepage op <http://www.casadeyork.com/jalv2/> en klik zeker ook even op de link naar de site van Stef) kan dat voor een PICMicro 16F877 met het volgende programma:

```

Forever loop
    Led = true
    Delay_100ms(5)

    Led = false
    Delay_100ms(5)
End loop

```

Dit voorbeeldje geeft duidelijk aan dat het echte werk niet veel tijd kost en we vooral staan te wachten. In dit geval verwaarloos ik zelfs de tijd van het werk – de loop en het inschakelen van de led en het uitschakelen van de led 0 – en vertaal de frequentie van 1 Hz zelfs naar 1 seconde wachttijd. Maar hoe kunnen we die wachttijd benutten voor iets anders? De eerste stap is ervoor zorgen dat we de wachttijd gaan vervangen door

een controle op de eindtijd. Het is dan niet meer relevant hoelang we wachten, maar of het al tijd is om de volgende actie uit te voeren. Dit kan als volgt:

```
Forever loop

    Led = true
    WachtTijd = 500
    While (WachtTijd > 0) loop
    End loop

    Led = false
    WachtTijd = 500
    While (WachtTijd > 0) loop
    End loop

End loop
```

We bepalen nu de eindtijd door de variabele WachtTijd het gewenste aantal milliseconden wachttijd te geven en vervolgens te wachten tot deze 0 is. De variabele wordt natuurlijk niet vanzelf 0, maar door een interrupt routine die er als volgt uitziet (Het is ook mogelijk zonder interrupts te werken, door uit te gaan van timers. In bijlage 2 is hiervan een voorbeeld opgenomen.

http://en.wikipedia.org/wiki/Finite_state_machine):

```
procedure timer_isr is
    pragma interrupt

    ;-----
    ; f = 20 MHz / 2 / 4 / (65536 - x) = 2.5 MHz / (65536 - x)
    ; voor f = 1 kHz, x = 63036
    ;-----

    tmr1h = 0xF6
    tmr1l = 0x3C

    If (WachtTijd > 0) then
        WachtTijd = WachtTijd - 1
    End if
    tmr1if = off

end procedure
```

En de interrupt voor timer1 te activeren wordt bij het opstarten de volgende code aangeroepen:

```
tmr1if = off
tmr1ie = on
gie = on
```

```
peie = on
tmr1on = on
```

We hebben nu de eerste stap gezet: in plaats een halve seconde te besteden aan een delay-routine kunnen we andere dingen doen, zoals in dit geval in een loopje hangen. De volgende stap is om ervoor te zorgen dat we verder kunnen gaan met ons programma, nadat we iets anders gedaan hebben. Het is het belangrijk om te realiseren dat we gewend zijn om de programcounter van de processor – die aangeeft waar we met de uitvoering van het programma zijn gebleven – te gebruiken om te onthouden waar we mee bezig zijn. Met andere woorden: als we weten welke regel code de processor uitvoert weten we hoe ver we met onze taak zijn. Als we bijvoorbeeld in de eerste lus zitten, is de led aan en wachten we totdat de 500ms is verstreken. Deze manier van programmeren sluit het meest aan bij onze manier van denken en voldoet in de meeste gevallen. Een prima methode dus. Maar... omdat een microcontroller maar één programcounter heeft, kunnen we op deze manier maar 1 ding tegelijk doen. En omdat we meerdere dingen tegelijk willen doen, moeten we per taak een variabele gaan gebruiken om bij te houden waar we gebleven zijn. Dit kunnen we doen door de taak onder te verdelen in stappen. Het knipperen van een led valt uiteen in twee stappen:

- Op moment dat we in stap 1 komen zetten we de led aan en stellen de WachtTijd op 500 (entry action).
- Vervolgens blijven we in deze state wachten totdat de WachtTijd 0 is geworden (transition condition) en gaan dan over naar stap 2 (transition).
- Op moment dat we in stap 2 komen zetten we de led uit en stellen de WachtTijd op 500 (entry action).
- Vervolgens blijven we in deze state wachten totdat de WachtTijd 0 is geworden (transition condition) en gaan dan over naar stap 1 (transition).

De hierboven beschreven stappenbenadering heet state machine en is eenvoudig in een procedure (functie, subroutine) te stoppen. Het knipperen van de led kunnen we als volgt in een procedure stoppen:

```
Var byte Taak1State = 1
Var word WachtTijd1 = 500
```

```
Procedure DoeTaak1 is
```

```
Case (Taak1State) of
1 : block
    Led = true
```