

Muziek in machinetaal (1)

Hans Zeedijk

Met behulp van de nieuwe A/D- en D/A-prints is het mogelijk uitvoerig in Uniface te gaan experimenteren met geluid, omdat deze printen voldoende snel zijn om respectievelijk geluidsignalen te meten en te genereren. In het algemeen zullen de daarvoor benodigde programmadelen in machinetaal geschreven moeten worden en dit artikel geeft een voorbeeld hoe het mogelijk is met de DAC8-print een eenvoudig liedje te spelen op een normale radio.

Aansluiting op de radio.

De schrijver van dit stuk heeft van electronica weinig verstand en van radio's helemaal niet; wel heeft hij de ervaring dat in de electronica veel kan zolang er maar geen kortsluiting wordt gemaakt en daarom is gewoon geprobeerd het uitgangssignaal van de Uniface DAC8-print direct aan die ingang van de eindversterker van een radio te hangen waarbij 300 mV stond en ziedaar het werkte. Nu geeft de DAC8-print signaal tussen 0 en 2,5 Volt gelijkspanning en daarom had ik bedacht, dat het beter zou zijn het signaal van de gelijkspanningscomponent te ontdoen door het via een condensator van 1 microfarad (kleine weerstand voor wisselspanningen) aan de versterker toe te voeren. Nu dat werkt ook probleemloos.

Programmering in BASIC.

De trillingen die opgewekt moeten worden door de DAC8-print moeten

uiteraard geluidstrillingen zijn om iets te kunnen horen, dus de frekquenties moeten tussen 20 en 10000 Hertz liggen, dit zijn dus trillingen per seconde. De normale toonhoogtes voor muziek liggen in de buurt van 500 tot 1000 Hertz en om een behoorlijke vorm aan de trilling te geven, b.v. een sinus, zijn toch wel een 10 punten per cyclus nodig. De DAC8-print zou dan 5000 tot 10000 getallen per seconde moeten "vertalen" naar analoge gelijkspanningen en aangezien de betreffende IC per vertaling niet meer dan 1 microseconde nodig heeft, zou dat goed moeten kunnen. De bottle-neck is in dit geval echter de snelheid, waarmee een programma in staat is de DAC8 aan te sturen en de verwachting om dit in BASIC-interpret te doen lukken was niet al te groot. Het volgende programmaatje produceerde echter toch wel een hoorbare toon op mijn NMS9100:

```
10 OUT 784,0:OUT  
785,191:OUT 785,190  
20 OUT 784,128:OUT  
784,0:GOTO 20
```

Hoewel hiermee geen muziek is te produceren, is het als voorbeeld wel inzichtelijk. De DAC8 heeft twee print-adressen. In dit geval wordt bij 191 op de adresbus (785) een commando naar de D/A-IC gestuurd via de databus (784). Het commando nul betekent, dat de IC "transparant" wordt, d.w.z. alle toegevoerde data worden direct met de hoogste spoed vertaald in een uitgangsspanning. Bij 190 op de adresbus worden data en geen commando's naar de IC gestuurd.

De data worden gegenereerd in regel 20 en zijn niets anders dan een herhaling van de getallen 128 en 0. De DAC8-print gaat daardoor een blok golf opwekken, waarvan de frekventie inderdaad in het hoorbare gebied blijkt te liggen. De radio produceert dan ook een niet al te fraaie constante dreuntone, maar voor variatie van de frekventie laat de BASIC-interpret geen ruimte.

Ook programmeer-experimenten in Turbo-Basic, die ik u verder zal onthouden, leveren weinig op. Er zijn weliswaar een aantal tonen van verschillende hoogte realiseerbaar, maar met muziek heeft het niet veel te maken.

Programmering in machinetaal.

Het is duidelijk: voor voldoende snelheid is het nodig de geluidstrillingen te programmeren in machinetaal. In het nu te behandelen voorbeeld gaan we een sinusvormige trilling in het



PTC-Print, juli 1990 - Nummer 39

20

*Artikel uit PTC Print 39
Copyright PTC en de auteur
Gescand en omgezet naar PDF door HansO, 2002*

hoorbare gebied opwekken. Het steeds berekenen van de volgende waarde van de trilling zou ook in machinetaal veel te veel tijd vergen. Dat doen we dan ook vooraf en gewoon in BASIC:

```
10 X=6.283185
20 DEF SEG=&H4200:FOR
I=1 TO 10000
30 POKE I,INT(SIN(X*I/
10000)*60+120.5):NEXT:
DEF SEG
40 DEF SEG=$H4200:
BSAVE"SINUS",0,10000:
DEFSEG
50 END
```

10000 waarden van een sinustrilling worden uitgerekend in regel 30, waarbij de waarden als integer-getallen tussen 60 en 180 worden weggepoked naar een vrij stuk geheugen van mijn NMS9100 te beginnen bij segment &H4200. Zoals gebruikelijk moet de hoek voor de sinusberekening opgegeven worden in radialen. X heeft de waarde 2 * pi en komt dus overeen met een hoek van 360 graden. Na de berekening worden de getallen als binaire file onder de naam SINUS naar disk weggeschreven.

Dit databestand van 10000 getallen wordt de basis van het opwekken van geluidstrillingen, die we opwekken door steeds opvolgende waardes uit te kiezen en naar de DAC8 te sturen, waarbij een variabel aantal cijfers wordt overgeslagen. Bijvoorbeeld het overslaan van steeds 99 getallen produceert een trilling met 100 waarden per sinus. Slaan we er 199 over dan zijn er 50 waarden per sinus en is de frequentie twee keer zo hoog als in het eerste geval. Het volgende machine-taalprogramma gemaakt in de DE-BUGGER, die iedereen die onder MS-DOS werkt bezit, doet dit:

Mnemonische code	Decimale waarde	Kommentaar
CLI	250	;zet interrupt af
PUSH DS	30	;data segment naar stack
MOV AX,4200	184 0 66	;begin van de file SINUS
MOV DS,AX	142 216	
MOV CX,xxxx	185 xxx xxx	;teller voor lengte toon
MOV SI,0000	190 0 0	;waarde voor SI register
** MOV BX,0000	187 0 0	;waarde voor BX register
* MOV AL,[BX+SI]	138 0	;breng eerste waarde in de processor
MOV DX,310	186 16 3	;interface op 784
OUT DX,AL	238	;data naar interface
ADD BX,xxxx	129 195 xxx xxx	;aantal waardes overslaan
CMP BX,4E25	129 251 16 39	;einde tabel ?
JL *	114 240	;nee, ga terug naar *
DEC CX	73	;verlaag teller toon
JNZ **	117 234	;doorgaan bij niet nul
POP DS	31	;restore data segment
STI	251	;set interrupt
RETF	203	;far return

```
5 REM *****
10 REM * Naam programma: MUZE1.BAS *
15 REM * Aanmaakdatum : 10-5-1990 *
20 REM * Computer : PHILLIPS NMS 9100 *
25 REM * Hardware : Komset-systeem met DAC8 *
30 REM * Beschrijving : Het programma genereert een geluidssignaal *
35 REM * uitgaande van een sinusvormige trilling en *
40 REM * speelt hiermee een liedje af. De hoogte van de *
45 REM * tonen en de lengte van de noten staan ingesteld *
50 REM * in een tabel. *
55 REM * *
85 REM * Opmerkingen : DAC8-adres 190/191 *
90 REM *****
100 DATA 250,30,184,0,66,142,216,185,0,1,190,0,0,187,0,0,138,0,186,16,3,238,
129,195,128,0,129,251,16,39,114,240,73,117,234,31,251,203
110 FOR I=0 TO 37:READ X:POKE 20000+I,X:NEXT
150 DEF SEG=&H4200:BLOAD"sinus",0:DEF SEG
160 DIM A(100,3)
170 FOR I=0 TO 25:READ A(I,0):READ A(I,1):READ A(I,2):READ A(I,3):NEXT
180 DATA 128,0,104,0,144,0,117,0,68,1,131,0,162,0,131,0,192,0,156,0,86,1,139,
0,171,0,139,0,216,0,175,0,192,0,156,0,216,0,175,0,192,0,156,0,171,0,139,0,136,2,
131,0
181 DATA 192,0,156,0,216,0,175,0,192,0,156,0,171,0,139,0,68,1,131,0,162,0,131,
0,192,0,156,0,86,1,139,0,171,0,139,0,162,0,131,0,32,1,117,0,128,1,156,0,16,5,131,
0
200 OUT 785,191:OUT 784,0:OUT 785,190
210 A=20000:P=A+8:Q=A+9:R=A+24:S=A+25
220 FOR I=0 TO 25
230 POKE P,A(I,0):POKE Q,A(I,1):POKE R,A(I,2):POKE S,A(I,3)
240 CALL A
250 NEXT
260 END
```

Deze machinetaalprocedure genereert een toon van een hoogte die nog ingevuld moet worden en van een lengte die ook nog ingevuld moet worden (plaats van de kruisjes). Dit invullen doen we gewoon in BASIC.

De listing van het gehele muziekprogramma MUZE1 is bijgevoegd. In de regels 100 en 110 wordt bovenstaande machinetaal subroutine ingelezen in een stukje van het BASIC-geheugen vanaf 20000. Vervolgens wordt in regel 150 de SINUS-file opgehaald van disk, die we eerder gemaakt hadden. Het stukje muziek staat in de tabel A(100,3), steeds twee getallen voor de toonhoogte en twee getallen voor de lengte van de toon. Deze tabel wordt ingelezen in de regels 160 t/m 181. Regel 200 maakt de DAC8 transparant. Regel 210 geeft aan waar het machine-taalprogramma zich bevindt en op welke plaatsen de toonhoogte en de toonduur in dit programma nog gepoked moeten worden. De regels 220 t/m

250 spelen een liedje van 26 noten af. Uiteraard is dit een begin van een serie mogelijkheden. Door SINUS te vervangen door andere golfpatronen kunnen klankkleuren gemaakt worden en instrumenten nagebootst; een stuk programma kan toegevoegd worden dat zelf de codes genereert voor de toonhoogte en de toonduur; het volume van de toon kan toegevoegd worden; verschillende tonen kunnen gemengd worden tot accorden en dan afgespeeld worden; enz.

Een interessant aspect is, dat de tonen zeer zuiver aangemaakt kunnen worden door de nauwkeurige programmering van de frequenties. Wellicht biedt dit mogelijkheden voor het stemmen van muziekinstrumenten. Omdat de DAC8 ook in dubbele uitvoering is te krijgen, moet het in principe ook mogelijk zijn "stereogeluid" te produceren en af te spelen.

Hoewel ik het nog niet heb gedaan, ben ik ervan overtuigd, dat dit programma uiteraard met een aangepaste machinetaal subroutine ook moet kunnen werken op een P2000 en een MSX.

Prijsvraag

Tot slot een prijsvraag. In het programma MUZE1 wordt een populair liedje afgespeeld op de DAC8. Onder diegenen die de juiste titel van dit liedje opsturen naar de redactie van de PRINT binnen drie weken na het verschijnen van dit artikel wordt een fraaie prijs verloot. Denk niet dat u voor de oplossing gedwongen wordt de DAC8-print aan te schaffen; dat moet ook kunnen door slimme lieden door middel van analyse van het programma.