

# De UNIFACE ADC-kaart

Een nieuwe loot aan de UNIFACE-stam

Hans Zeedijk, Dick Kroonenberg, Anton Bombeek

In PTC Print nummer 9 besloot Klaas Robers zijn artikel "Uniface, de buitenwereld aan de computer" met de oproep aan de PTC-leden ideeën te ontwikkelen voor nieuwe buitenwereld-delen. Eén idee ligt uiteraard voor de hand. Met de 8-bit IN en 8-bit UIT kaarten is het mogelijk vanuit de computer in digitale vorm met de buitenwereld te communiceren. In veel gevallen willen we echter analoge informatie behandelen. Een eenvoudig voorbeeld: als men via de computer informeert naar de temperatuur in de buitenwereld, is het onbevredigend te vernemen, dat het koud of warm is. In dat geval wenst men ook te weten hoe koud of hoe warm.

## Analoog/digitaal conversie en digitaal/analoog conversie

In het geval van analoog/digitaal conversie wordt een continue variabele grootheid omgezet in een elektrische spanning (bijvoorbeeld temperatuur met een thermokoppel) en vervolgens het elektrische signaal vertaald (geconverteerd) door een elektronische schakeling in een (binaire) getalswaarde. In het dagelijkse leven zijn we vertrouwd met een dergelijk omzettingproces. Wanneer de temperatuur van een thermometer wordt afgelezen, dan zetten we met behulp van de schaalverdeling de continue variabele lengte van de kwikkolom om in een getalswaarde en het meten van een afstand met een lineaal is eveneens een a/d conversie.

Digitaal/analoog conversie is het omgekeerde: een getalswaarde in de computer wordt naar buiten gebracht als een elektrische spanning overeenkomend via een "vertalings-schaal" met die getalswaarde.

## ADC-kaart

Op het moment dat U dit leest zal naar verwachting in de PTC-winkel de Uniface ADC-kaart verkrijgbaar zijn, zodat voortaan de mogelijkheid bestaat analoge gegevens uit de buitenwereld in te voeren in de computer via het Uniface-systeem. In dit geval kunnen we ook in meervoud spreken, omdat de

ADC-kaart vier ingangen heeft, die elk hun eigen signaal kunnen ontvangen en omzetten in getallen, uiteraard met behulp van een computerprogramma. Het omgezette spanningssignaal uit de buitenwereld wordt uitgedrukt in een getalswaarde tussen 0 en 4095, overeenkomend met spanningen tussen 0 en vijf Volt. De kleinste meetbare spanning komt dan overeen met  $5000 \text{ mV} / 4095 = 1,2 \text{ mV}$ . Het getal 4095 is niet toevallig, maar komt overeen een woordlengte van 12 bits. Uiteraard is ook de tijd die nodig is voor de conversie van de analoge waarde in het digitale getal van belang: de ADC-kaart heeft hiervoor 150 microseconden nodig.

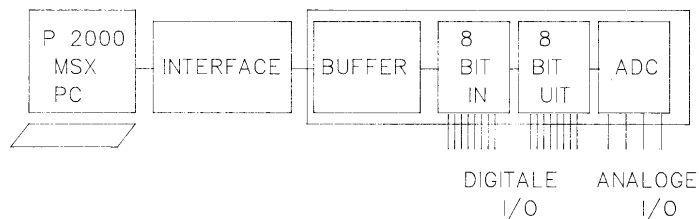
Evenals het geval is bij de 8-bit IN/UIT kaarten is het adres van de ADC-kaart in het Uniface-systeem vrij kiesbaar tussen 0 en 255 en instelbaar met schakelaars op de kaart zelf. De enige voorwaarde is uiteraard, dat er geen tweede kaart met hetzelfde adres bestaat. De ADC-kaart vraagt wel een aparte voeding die behalve +5 ook +15 en -15 Volt kan leveren. De tekening geeft een schematisch overzicht van de diverse bestanddelen van een volledig Uniface-systeem.

De analoge output-kaart is nog niet in de PTC-winkel verkrijgbaar, maar het prototype is gereed en het besluit is genomen een print voor deze kaart te ontwikkelen.

## Het programmeren van de ADC-kaart

Het hart van de ADC-kaart is de AD 7582 chip van Analog Devices. De adressering van de kaart en de aansturing van deze chip is zoals gebruikelijk in het Uniface-systeem, maar wel ingewikkelder dan voor de digitale I/O kaarten, omdat er meer besturingshandelingen nodig zijn. Dat spreekt ook voor zich, omdat de ADC-chip niet alleen moet weten, wanneer hij zijn werk moet beginnen, maar ook welk ingangskanaal afgelezen moet worden. De uitlezing van 12 bits dient ook in twee bytes te geschieden, dus de chip moet ook een commando kunnen ontvangen, welke byte het eerst geproduceerd moet worden. In dit verband zullen we spreken van het lower byte (met de 8 minst significante bits van het uitgelezen binaire getal) en het higher byte, waarvan alleen de 4 laatste bits een betekenis hebben voor de meetwaarde. Van dit higher byte heeft het meest significante bit wel een betekenis: een 1 op deze plaats is het teken dat de ADC-chip zijn conversie beëindigd had (busy low). Het commanderen van de ADC-chip geschiedt nu met de eerste 5 bits van het commando-byte:

- Bit 0 en 1 zijn nodig voor de formulering van het multiplexkanaal van de ADC-kaart (de 4 mogelijkheden zijn 00, 01, 10, en 11).
- Bit 2 is de byte select (BSL), waarmee we de chip aangeven



welk resultaat-byte uitgelezen gaat worden.

- Bit 3 geeft een write opdracht (WR).
- Bit 4 heeft als betekenis chip select (CS), waarmee de ADC-chip wordt aangesproken.
- Bit 5 geeft een read opdracht (RD).

De andere drie bits van het commando-byte hebben geen betekenis. De formulering van het commando-byte is negatief, dat wil zeggen een betreffend bit in het byte is geset wanneer het de waarde 0 bezit.

#### Het programma

Het volgende programma met uitleg geeft het gebruik van de ADC-kaart in BASIC voor een MSX-computer, waarbij de output-poorten &H30 en &H31 gebruikt worden voor resp. datatransport van/naar en adressering van het Uniface-systeem (zie de eerdere artikelen van Klaas Robers).

```

10 REM Aansturing ADC-
   kaart in het UNIFACE-
   systeem.
20 OUT &H30,229:
   REM Het MPX-adres is
   01 ofwel kanaal 1; CS
   en WR laag (schrijven
   naar de chip)
30 OUT &H31,5:
   REM 5 is het kaart-
   adres; hiermee wordt
   het getal 229 inge-
   klokt
40 OUT &H30,255:REM Reset
   van alle databits
50 OUT &H30,207:
   REM CS en RD laag;
   voorbereiding uitlezen
   van het meest signifi-
   cante byte
60 A=INPUT(&H30):
   REM Het higher byte
   wordt gelezen door de
   computer
70 OUT &H30,203:
   REM CS, RD en BSL zijn
   laag
80 B=INPUT(&H30):
   REM Het lower byte
   wordt gelezen door de
   computer
90 OUT &H30,255:
   REM Alle besturings-
   bits terugzetten
100 OUT &H31,0:
   REM Reset adressering
   ADC-kaart
110 C=(A AND 15)*256+B:
   REM Uitrekenen van de
   meetwaarde. De AND-
   operatie is nodig om
   de vier higher bits
   zeker 0 te maken

```

De variabele C heeft na uitvoering van dit programma een waarde, die evenredig is met de elektrische spanning op de ingang 1 van de ADC-kaart. Spitsse lezers zal het zijn opgevallen, dat in het programma een test ontbreekt op het BUSY-sigitaal van de ADC-chip (bit 7 van de higher byte van de meetwaarde). In BASIC is deze test niet nodig, omdat de ADC-chip toch veel minder tijd nodig heeft voor een

conversie dan de computer voor het lezen en uitvoeren van een programmaregel. Voor een gecompileerd programma of een machinetaal meetprogramma gaat dit niet op en moet de BUSY-lijn van de ADC-chip wel onderzocht worden voor het detecteren van het einde van de conversie. Een volgende keer meer over de ADC-kaart.

---

## UNIFACE-TIP

### Besturing via tijdsregeling

Naar aanleiding van het artikel over de LEGO-besturing met behulp van UNIFACE in PTC Print nr. 8 en 9 hier een eenvoudig P2000-programma om de werking van UNIFACE te illustreren. Ga als volgt te werk: Sluit controlelampjes aan op de bits 0 en 3 en aarde en het motortje op de bits 1 en 2 van de uitgangskaat. Sluit de schakelaars (gewone enkelpolige) 1, 2 en 3 aan op de bits 0, 1, 2 en aarde van de ingangskaat.

Op de uitgangskaat moet ook nog een externe voedingsspanning worden aangesloten. In het geval van LEGO moet dit 5V zijn, maar het mag tot 24V gaan. Gebruik hier eventueel de batterijstaaf van LEGO voor. Met het programma krijgt u nu een indruk van de mogelijkheden van een automatische besturing met tijdsinstelling, die te stoppen is via toetsenbord of schakelaar.

P. Fransen

```

10 LPRINT CHR$(12)
20 PRINT "Besturing motor via tijdsregeling met
   controlelamp voor richting"
30 INPUT"Tijd vooruit (sec)";V:REM gewenste tijd
   vooruit
40 V=V*888:REM lus van 888 is 1 sec.
50 INPUT"Tijd achteruit (sec)";A:REM gewenste tijd
   achteruit
60 A=A*888:REM 888 = 1 sec.
70 INPUT"Stoptijd (sec)";S:REM gewenste stoptijd
80 S=S*888:REM lus van 888 is 1 sec.
90 OUT 97,8:REM adres van in/uitgang 8
100 OUT 96,0:REM alle uitgangen van kaart 8 nul
110 FOR Q=1 TO S:NEXT
120 IF INP(96)=1 THEN 230:REM stopt met schakelaar
130 ON ERROR GOTO 230:REM stopt met stoptoets
140 OUT 96,12:REM lamp v aan en motor vooruit
150 FOR Z=1 TO Y:NEXT
160 OUT 96,0
170 FOR R=1 TO S:NEXT
180 OUT 96,3:REM lamp a aan en motor achteruit
190 FOR Y=1 TO A:NEXT 200 OUT 96,0:REM lamp uit en motor stop
210 FOR P=1 TO S:NEXT
220 GOTO 120
230 OUT 97,8:OUT 96,0:REM uitgangen nul
240 OUT 97,0:REM adres nul
250 END

```

---