



Technische gegevens

Voedingsspanning	5 volt DC
Stroomverbruik onbelast	100mA
Uitgangsspanning	Hoog 3,5 volt Laag 0,3 Volt
Ingangsspanning	Hoog 3,5 Volt Laag 0,3 Volt

De CX81-I/O is een ingangspoort of een uitgangspoort voor een ZX81 of Spectrum. Men spreekt van een 8-bit I/O-poort omdat de poort 8 signalen tegelijk kan verwerken. Om te kunnen werken is 16K geheugen nodig, dus bij gebruik van een ZX81 is een RAM-pack noodzakelijk. In de softwareservice van HOBBIT is een cassette beschikbaar met de programma's die in artikel afgedrukt staan. Als U een Spectrum bezit moet U de I/O-poort als laatste unit op de uitbreidingsconnector aansluiten. Het is mogelijk met de computer een machine te besturen, een diefstalalarm te bouwen, een modeltrein te regelen en veel, veel meer. Om bijvoorbeeld maar bij de modeltrein te blijven: de computer "ziet" de stand van een sein of van een wissel en kan, afhankelijk van de situatie, actie ondernemen!

Hardware

De I/O-poort bouwt U op de dubbelzijdige printplaat HB422. De verbindingen tussen boven- en onderzijde van de print komen tot stand door middel van doormetaliserings van de print. Daarom worden de onderdelen alleen aan de onderzijde gesoldeerd. U mag de gaatjes niet groter boren mocht een onderdeel niet passen. De print wordt dan ernstig beschadigd! Ook goed opletten bij het solderen want tussen sommige printsporen is de afstand soms maar 0,3 mm!

De bouw

Volg de componentenlijst en de tekening van de componentenopstelling (fig. 2). Let even op bij het plaatsen van R1. Zie hiervoor figuur 1. De IC's kunt U het beste op voetjes plaatsen want een defekt IC uit de print solderen is vragen om problemen. Het printje kan natuurlijk ingebouwd worden in een kastje. Zorg er echter wel voor, dat beide connectoren goed bereikbaar blijven. De CX81-I/O steekt U nu op de uitbreidingsconnector van de computer. Bij de ZX81 de geheugenuitbreiding niet vergeten.

De CX81-I/O poort

Een input en outputpoort voor de Sinclair ZX81 en Spectrum computer (en afgeleide apparaten, b.v. Timex 1000).

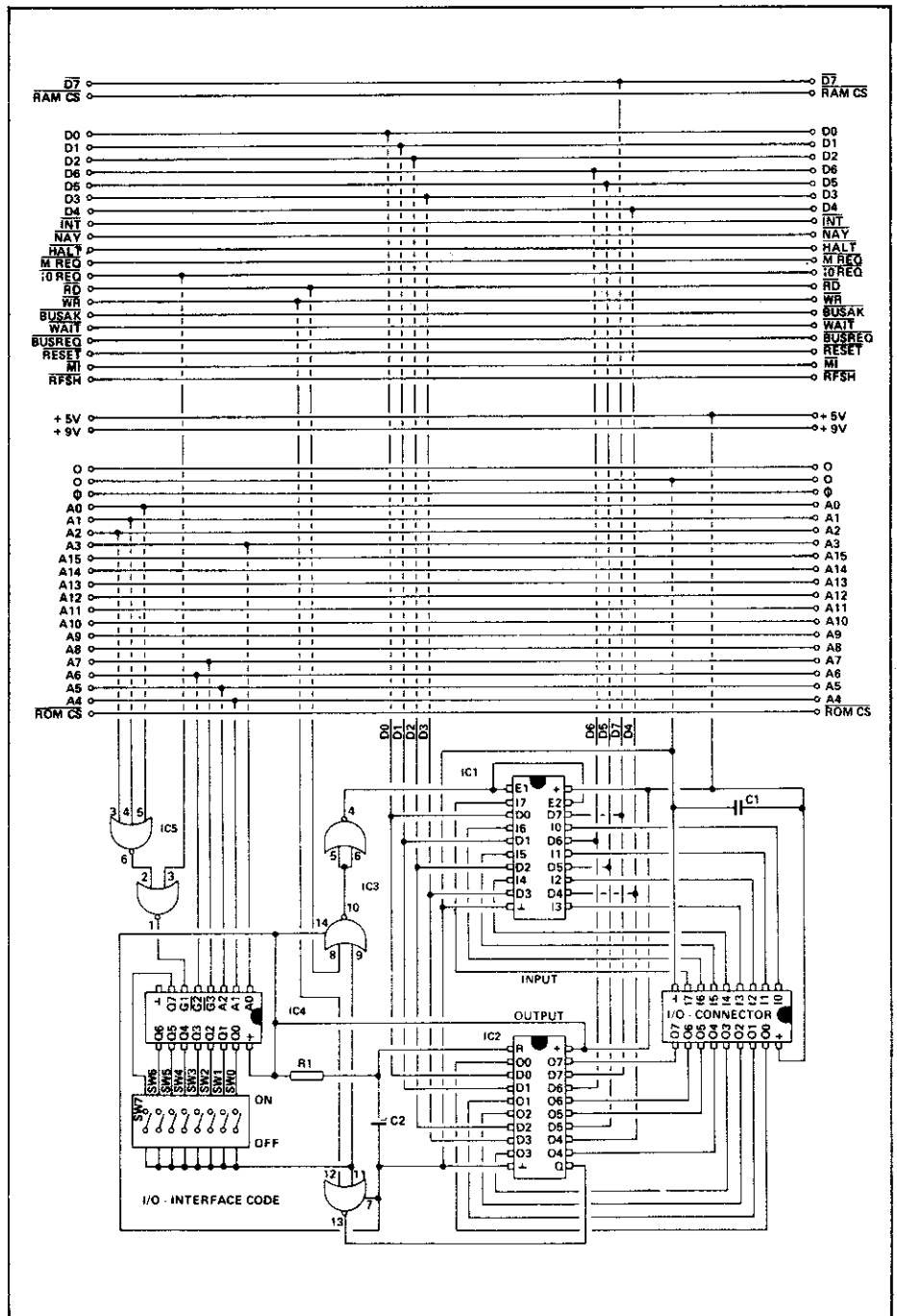


Fig. 3: Het schema van de poort.



Opmerking

Met de 8-voudige schakelaar SW wordt de print geadresseerd. Elke print krijgt zijn eigen adres zodat de computer hem kan "vinden". Wanneer schakelaar 1 van SW gesloten is kan de print met het hierna beschreven programma werken. Dat geldt ook voor het in het Hobbit software service opgenomen programma. U moet de software zelf aanpassen voor de andere mogelijkheden. Bij de Spectrum is dat heel eenvoudig; het wordt gedaan door het wijzigen van de Basic-commando's IN en OUT. Bij de ZX81 moeten in het in machinetaal geschreven programma veranderingen worden aangebracht. Maar laat U niet afschrikken, als U het hier beschreven basic programma gebruikt is het werken met de I/O-poort heel eenvoudig. "C0" staat voor de print waarop schakelaar 1 van SW ingeschakeld is. "C1" voor de print waarop schakelaar 2 van SW ingeschakeld is. "C2" voor schakelaar 3 enzovoort. C0 gebruikt de In/Out adressen van de Z80 die lopen van hex 07 tot en met hex 0E (dus 8 adresplaatsen of 8 bytes).

Waarschuwing

Wanneer U bij ingeschakelde computer een van de schakelaars omzet kunnen de IC's van de I/O-poort defekt raken. Dus eerst de computer uitschakelen als U iets wilt veranderen op de print!

Schema

Het schema toont de 8-bits in- en uitgangspoort. De poort wordt gestuurd door de data- en adresbus van de computer. Bij de ZX81 gebeurt dat met machinetaalinstructies. Bij de Spectrum kan het met machinetaalinstructies maar ook met de basic commando's IN en OUT. Zie bijvoorbeeld het programma voor de Spectrum regel 2020 en volgende.

I/O-request

De Z80 microprocessorchip die in de ZX81 en in de Spectrum toegepast wordt heeft 256 I/O adressen. Adressering hiervan gebeurt met de adresbus en wel met de lijnen A0 t/m A7. Met een I/O instructie kan een bepaald adres geselecteerd worden. Als dat adres overeenstemt met het adres dat op de I/O-print met SW is ingesteld zal betreffend I/O-print aangesproken

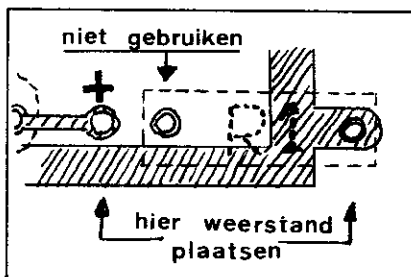


Fig. 1: Zó moet weerstand R1 geplaatst worden.

worden. Er moet natuurlijk op gelet worden dat andere op de bus aangesloten apparatuur niet wordt beïnvloed. De Sinclair printer en de microdrive bijvoorbeeld gebruiken ook enkele van de I/O adressen. De I/O-poort gebruikt adressen die vrij zijn en dat zijn de volgende:

Connectornummer (dus welke print wordt bedoeld)							
C0	C1	C2	C3	C4	C5	C6	C7
Dil schakelaar SW							
1	2	3	4	5	6	7	8
Decimaal adres							
7	15	23	31	39	47	55	63
Hexadecimaal adres							
07	0F	17	1F	27	2F	37	3F

NB: er mogen nooit twee of meer schakelaars ingeschakeld staan. Dan ontstaat een foutief adres. De print met het adresnummer C0 gebruikt dus de adressen 07, 08, 09, 0A, 0B, 0C, 0D en 0E. De print met adresnummer C1 gebruikt 0F, 11, 12, 13, 14, 15, 16 en 17 enz. De in- en uitgangen van de computer kunnen niet veel stroom leveren. Daarom worden ze gebufferd met IC 74LS244 die stuursignalen ontvangt van de strobe-uitgang van de computer. De informatie komt zodoende op het juiste tijdstip op de bus. Omdat het niet altijd mogelijk is de aangeboden informatie meteen te verwerken wordt de informatie tijdelijk opgeslagen in een tussengeheugen (74LS273). Na elke uitlees- en stuurcyclus wordt dit geheugen gereset zodat er dan weer nieuwe informatie in geplaatst kan worden. De schakeling heeft 10 tot 100m Sec. nodig om een cyclus af te werken. Het kan sneller, maar dan moet U het programma helemaal in machinetaal zetten. Voor testdoeleinden kunt U de poortuitgang rechtstreeks verbinden met 8 LED's. LED's aansluiten tussen uitgang en massa.

Software

Als U de I/O-poort aangesloten hebt op de Spectrum kunt U de poort rechtstreeks besturen met de basic IN en OUT instructie. Voorbeeld:

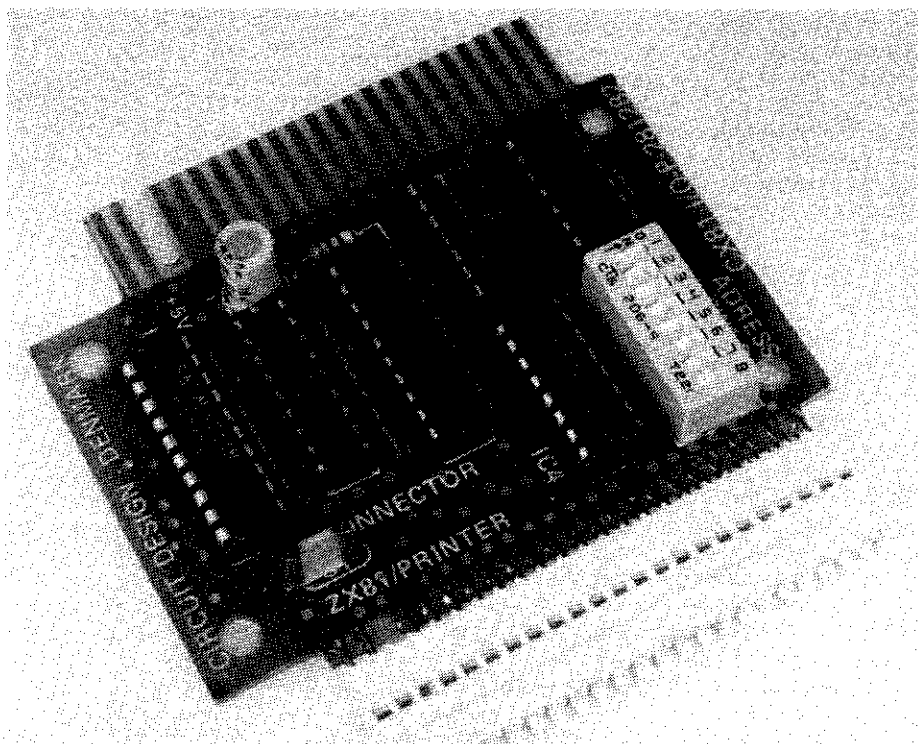


Foto 1: De complete I/O-print.

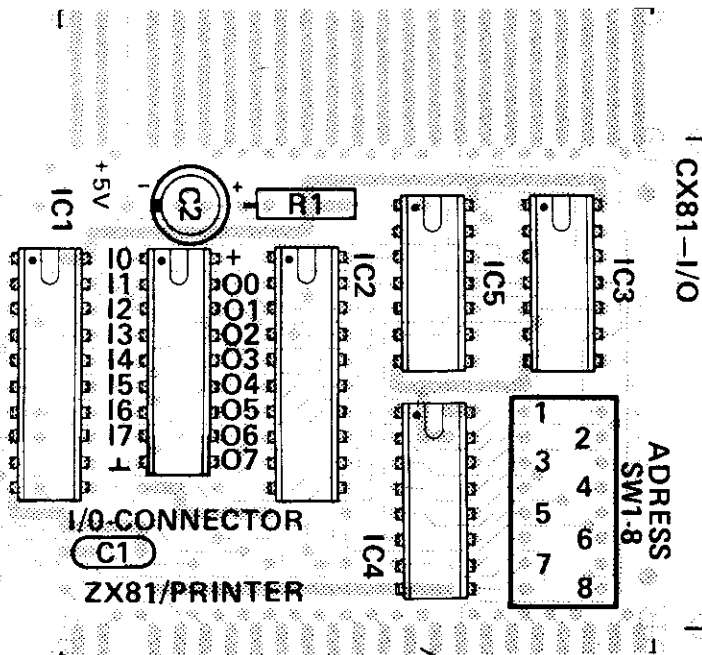


Fig. 2: De componentenopstelling.

OUT 7,8 laat de LED aan uitgang 3 oplichten. (7 is het adres van de poort: 8 is de decimale waarde van de geheugeninhoud, dus 00001000. OUT 7,255 laat alle LED's oplichten. OUT 7, bin 00110011 laat de LED's 0,1,4 en 5 oplichten. Bij de instructie IN 7 leest de computer de stand van de I/O-poort en schrijft deze op adres 07. Voor de ZX81 is het noodzakelijk dat U een gedeelte van het programma (of het hele programma) in machinetaal invoert. Hobbit heeft hiervoor een hulpprogramma opgenomen (zie listing in figuur 4). Na de eerste REM instructie moeten 387 punten volgen. Dit zijn geheugenplaatsen die we reserveren voor het machinetaalprogramma. Als het programma ingetypt is start U met RUN. De computer wacht op het invoeren van de 387 kodes (zie listing in figuur 5). U voert de kodes in van links naar rechts. Na elke code (2 cijfers) drukt U op ENTER. Als U alle 387 kodes hebt ingevoerd geeft de computer aan of dit goed (Input OK) of fout (Input fejl) is gebeurd. Als U het niet goed gedaan hebt zult U weer helemaal opnieuw moeten beginnen. Als het wel goed is gegaan kunt U het programma van figuur 8 intypen.

Let op

Het machinetaalgedeelte dat U zojuist weggeschreven hebt staat in het REM statement van regel 1. U mag regel 1 dus niet verwijderen met CLEAR of

NEW. U begint gewoon met het intypen van het programma vanaf regel 2. Het oude programma wordt dan overschreven. Hoewel regel 2 t/m 26 niet van belang zijn voor de werking en U deze mag overslaan is het toch beter ze wel in te typen. Dan gaat er zeker niets mis. Zorg er voor dat, als U de eerste 26 regels toch over wilt slaan, in ieder geval regel 1 blijft staan en regel 26 eruit gaat (26 intypen en laten volgen door ENTER).

PX81-I/O software op cassette

Heeft U een hekel aan het intypen van het programma dan kunt U het programma ook op cassette kopen. De cassette bevat het programma voor beide computers. Op kant A vindt U de software voor de ZX81 en op kant B voor de Spectrum. Het programma kunt U laden met de naam "PX81-I/O". Dit duurt met de ZX81 ongeveer 2 minuten en met de Spectrum 30 seconden. Als ze goed geladen zijn volgt een autostart en verschijnt het volgende beeld (zie figuur 6). U kunt nu werken met de poort en hebt daarbij de volgende mogelijkheden:

P = Page select: (Alleen ZX-81). Met behulp van dit commando kiest U het paginanummer. Een pagina is in dit geval een blok geheugen waar Uw eigen programma in staat. Voorbeeld: U kiest P3. U roept hiermee een programma aan dat de ruimte beslaat vanaf regel 3010 tot maximaal 3989 en dat U zelf

De Componentenlijst

Weerstanden:

R1 = 1k 1/4W

Kondensatoren:

C1 = 100nF Sibatit steek 5 mm

C2 = 47µF/16 volt elko radiaal

Halfgeleiders:

IC1 = 74LS244

IC2 = 74LS273

IC3 = 74LS02

IC4 = 74LS138

IC5 = 74LS10

Diversen:

Print HB422

IC voeten 2 x 14 polig

1 x 16 polig

1 x 18 polig

2 x 20 polig

SW DIL schakelaar 8-voudig

B1 kaartconnector 2 x 23 polig 0,1 inch penafstand

Eventueel:

18 polige Dip-connector voor uitgang.



hebt ingebracht (zie ook listing in figuur 8). P0 is het demonstratieprogramma en geeft de gebruiker informatie over het gebruik van de I/O-poort. P1 tot en met P8 kunt U zelf definiëren. P9 is de save-routine. Hiermee kunt U het hele programma (ook het door U zelf ingevulde gedeelte) op een cassette vastleggen.

CS Connector select:

Hiermee kunt U de adresplaats van de I/O-poort kiezen. U wijst dus als het ware aan om welke I/O-print het gaat. De printen heeft U een nummer gegeven (C0, C1, C2, etc.). Als U bv. kiest voor C0 zullen alle In/Out handelingen naar de print gaan die als C0 geselecteerd is (de print waar schakelaar 1 van SW "aan" staat).

SB of RB (Set bit of Reset bit):

Met deze instructie kunt U elk bit afzonderlijk hoog of laag maken.



Voorbeeld: SB2 maakt bit 2 van de gekozen I/O-print hoog.

DS = Decimal:

U voert achter DS een decimaal getal tussen 0 en 255 in. De computer berekent het binaire getal en maakt van de I/O connector de aansluitingen hoog die de waarde 1 hebben in het binaire getal. Voorbeeld: DS47. De computer rekent het getal om. Binair wordt het 00101111 en dus worden de uitgangen 0,1,2,3 en 5 hoog gemaakt.

RI of RO = Read bit:

U kunt per bit (0 tot 7) de waarde aflezen op het scherm. R1 geeft de waarde van het betreffendeingangssignaal en R0 van het uitgangssignaal.

B1 of B0 = Binary read:

Zet op het scherm bij de betreffende aansluiting de waarde van alle bits (binair). B1 doet dat voor de ingangspoort en B0 voor de uitgangspoort.

D1 of D0 = Decimal read:

Geeft de decimale in- en uitgangswaarde achter het woordje INPUT en OUTPUT op het scherm (rechts van de afbeelding van de I/O-Connector).

Opmerking

Bij RO, BO en DO is voor het verkrijgen van een zo betrouwbaar mogelijk resultaat noodzakelijk de oude waarde eerst te verwijderen met SB, RB of DS. Het snelst gaat dit met het commando DS0 omdat dit commando alle uitgangen op 0 brengt en omdat U dit anders per uitgang moet doen.

Als Page 1 (en alle volgende) op het scherm verschijnt (zie figuur 7) verandert de scherminhoud een beetje. Er staat een nieuw commando AI (Read all inputs). De commando's RI, RO, BI, BO, DI en DO zijn niet meer aanwezig. U mag meerdere commando's tegelijk intoetsen. Op het scherm blijven de gegevens "input" en "output" én bij de connectoraansluitingen staan totdat een nieuw commando wordt ingevoerd of door de computer uitgevoerd.

Toepassing van de I/O-commando's in uw eigen programma

Er is plaats voor 8 gebruikersprogramma's tussen de regels 1020 en 8920. Deze programma's kunt U oproepen met het P-commando. Het programma dat U zelf schrijft sluit U af met: Goto 10.

Advies

Als U uw programma wilt lijsten (LIST) gebruik dan LIST 10 en dus niet alleen LIST. Als U het programma wilt "saven" gebruik dan eerst RUN en kies vervolgens Page 9 (P9). Het programma wordt dan op de band weggeschreven. Bij de besturing van de I/O-poort gebruiken we machinetaalroutines. U kunt de instructies opgeven in de variabele C\$ (zie demonstratieprogramma ZX81 regel 2080 tot en met regel 2100). U mag natuurlijk ook zelf het machinetaalprogramma schrijven. Hier volgen de instructies met hun werking in ons programma:

RI, RO, DI, DO	Let variabel = USR I/O IF USR IO = Then ...	De 'variabel' bevat de C\$ waar in de commando's opgenomen worden
(Voorbeeld: C\$ = "RIO" = leest BIT 0 van IN-poort)		
RI, RO, DI, DO, BI, BO	Print USR IO	Drukt de opdracht op het scherm
CS, SB, RB, DS	RAND USR	Voert de op- dracht via I/O connector uit.

Op Page 2 (P2) staat een eenvoudige routine die de waarde van bit 0 van de IN-poort leest en overbrengt op de bits 0 en 1 van de UIT-poort.

IO is een variabele en heeft de waarde 16684.

NB: het commando "AI" kunt U niet gebruiken in uw programma.

Foutmelding

S met regelnummer:

De variabele C\$ is niet gedefinieerd.

W met regelnummer:

In de variabele C\$ komt een onbekend commando voor.

K met regelnummer:

Bij de commando's CS, SB, RB, RI en RO wordt een waarde gebruikt die niet tussen 0 en 7 ligt.

Basic

Nu volgen de listings van het basic programma voor de IO-poort. Voor de ZX81 is dat de listing van figuur 8 en voor de Spectrum die van figuur 9. Het is voor de werking niet nodig het hele

demonstratieprogramma in te voeren. Het verdient echter wel aanbeveling want de gang van zaken wordt zichtbaar gemaakt op het scherm en dat is veel duidelijker dan nog een paar pagina's tekst en uitleg.

Hardware

Als U de I/O-poort in combinatie met uw computer gebruikt kunt U softwarematig externe apparatuur sturen. De ZX computers zijn pas in samenwerking met uitbreidingen als de I/O-poort échte besturingseenheden. We zullen verschillende toepassingen als zelfbouwprojecten publiceren in de komende HOBBIT'S.

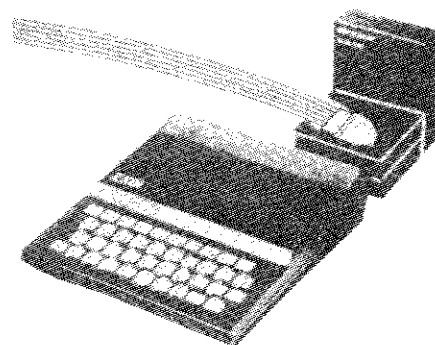


Foto 3: De ZX81 met I/O-poort en RAM-pack.



```

ZX-SPECTRUM I/O-MODULE
DEMONSTRATION PROGRAM

PX81-I/O

MADE IN DENMARK
© CIRCUIT DESIGN
Jan Soelberg
Carsten Claudinger

DIM U(8): LET i0=7: DIM d$(
FOR i=1 TO 8
  U(i)="0 0 0 0 0 0 0 0"
NEXT i
PRINT "PX81-I/O"
INK 3: PRINT AT 0,18;"CIRCUIT
DESIGN"
INK 0: PLOT 0,167: DRAW 255
PRINT AT 2,22;"INPUT"
INK 5: PRINT " "
INK 0: PRINT AT 3,4;"0 0 0 0"
PRINT " "
PRINT " I/O CONNECTOR 0 "
PRINT " "
PRINT " 0 0 0 0 0 0 0 0 0 0 "
INK 2: PRINT AT 7,18;" "
INK 0: PRINT TAB 22;"OUTPUT"
PRINT " "
PRINT "COMMANDS:"
PRINT TAB 10;"CONNECTOR S
ELECT C50"
PRINT TAB 10;"SET BIT";TAB
22;"OUTPUT SB"
PRINT TAB 10;"RESET BIT 0
UTPUT FB"
PRINT TAB 10;"DECIMAL SET 0
UTPUT DS"
PRINT TAB 10;"READ BIT
INPUT FI"
PRINT TAB 22;"OUTPUT FO"
PRINT TAB 10;"READ
INPUT IP"
PRINT TAB 22;"OUTPUT OP"
INPUT "COMMANDS:";CS
IF LEN CS>32 THEN LET CS=CS
(TO 32)
LET d$=CS
PRINT AT 21,0; d$
FOR P=1 TO LEN CS STEP 2
  LET t=p+1
  IF t>32 THEN GO TO 250
  IF d$(p TO t)="op" THEN OUT
  IO U(i): PRINT AT 8,2;U(i);AT
  8,29;U(i);
  IF d$(p TO t)="ip" THEN LET
  s=IN i0: GO SUB 400: PRINT AT 2
  ,4; s$;AT 2,29;V;
  FOR Z=t+1 TO LEN CS
  305 IF d$(Z)<"0" OR d$(Z)>"9" T
  HEN GO TO 315
  310 NEXT Z
  320 LET z=z-1
  330 IF z<t<1 THEN NEXT P: GO TO
  250
  340 IF d$(p TO t)="ds" THEN LET
  s=VAL d$(t+1 TO z): GO SUB 400:
  OUT IO,V: LET u$(i)=s$: LET U(i
  )=v: PRINT AT 8,2;U(i);AT 8,29;
  U(i);
  350 IF d$(p TO t)="sb" THEN LET
  o$="1": GO TO 600
  360 IF d$(p TO t)="fb" THEN LET
  o$="0": GO TO 600
  370 IF d$(p TO t)="fo" THEN LET
  v=VAL d$(z): LET a$=U(i): GO S
  UB 500: PRINT AT 8,2;B$;AT 8,29;
  S;
  375 IF d$(p TO t)="fi" THEN LET
  s=IN i0: LET v=VAL d$(z): GO SU
  B 415: GO SUB 500: PRINT AT 2,4;
  B$;AT 2,29;S;
  380 IF d$(p TO t)<>"cs" THEN GO
  TO 395
  385 LET i=VAL d$(z)+1: LET i0=i
  *8-1: IF i>8 THEN LET i=8: LET I
  0=63
  387 PRINT AT 5,17;I-1
  390 LET d$(p TO t)="op": GO TO
  280
  395 LET p=z-1: NEXT P: GO TO 25
  0
  400 REM dec to bin
  405 IF s>255 THEN LET s=255
  410 LET v=s
  415 LET a$="0 0 0 0 0 0 0 0"
  420 FOR n=s TO 1 STEP -1
  430 IF s/2<>INT (s/2) THEN LET
  a$(n*2-1)="1"
  435 LET s=INT (s/2)
  440 IF s>0 THEN NEXT n
  450 RETURN
  500 REM bit i/o
  510 LET B$=""
  520 IF v>7 THEN RETURN
  530 LET v=15-2*v: LET B$(v)=a$(
  v)
  540 LET s=0: LET p=p+1
  550 IF B$(v)="1" THEN LET s=2*(
  (15-v)/2)
  560 RETURN
  600 REM bin to dec
  610 FOR n=1 TO z
  620 IF d$(n)>"7" THEN NEXT n: G
  O TO 840
  630 LET U$(i,2*(8-VAL d$(n))-1)
  =0$
  640 NEXT n
  650 LET U(i)=0
  700 REM bin to dec/output
  720 FOR n=1 TO 8
  730 IF U$(i,2*n-1)="1" THEN LET
  U(i)=U(i)+2*(8-n)
  740 NEXT n
  750 LET d$(p TO t)="op": GO TO
  280
  800 REM
  8010 REM Demonstration of input/
  output via I/O-module
  connectornumber: 0
  8020 LET data=BIN 00000000
  8040 LET bit0=2*(IN 7/2-INT (IN
  7/2))
  8050 IF bit0 THEN LET data=BIN 0
  000011
  8060 OUT 7,data
  8070 GO TO 8030
  8080 SAVE "PX81-I/O" LINE 1
  8010 GO TO 1
  
```

Fig. 9: De basic listing van het Spectrum programma.

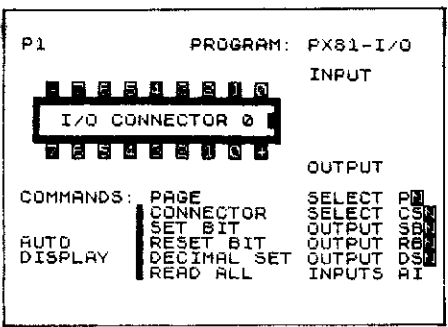


Fig. 7: Het "volgende" scherm.

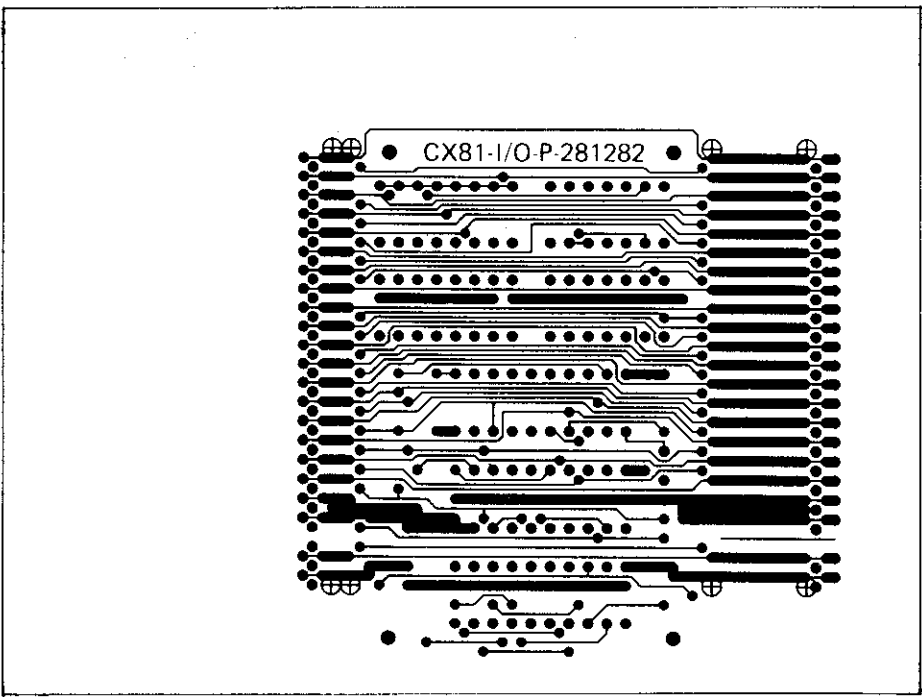
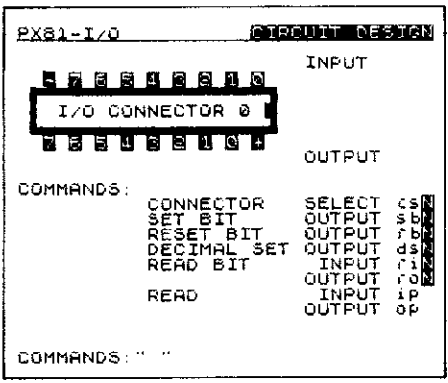
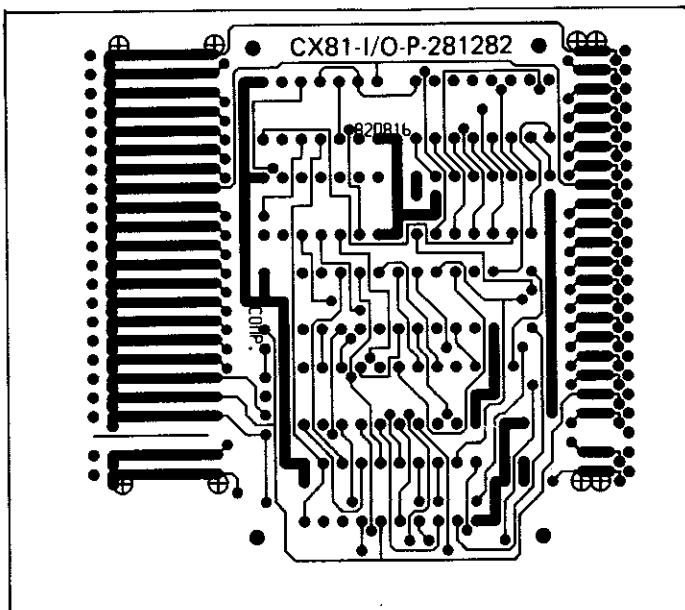
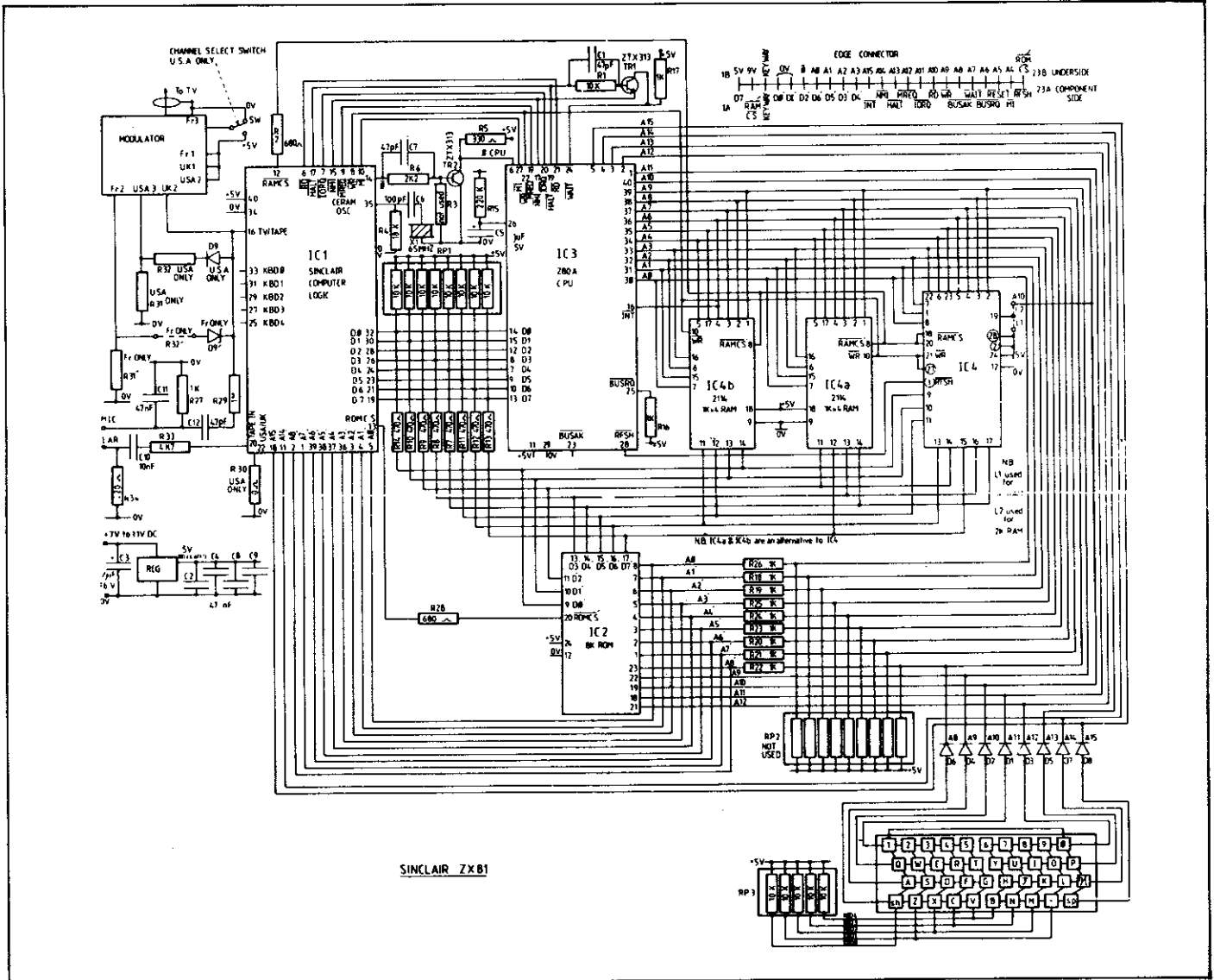


Fig. 10: De printlayout van de poort. De print is dubbelzijdig en doorgemetaliseerd, dus namaken is een hele klus. Hobbit levert een print in de printservice.



De andere zijde van de print.

