

User-Port-Display

Eine kleine elektronische Schaltung setzt den User-Port ins richtige Licht. Sie zeigt auch, wie der User-Port für Steuerungsaufgaben eingesetzt werden kann.

Die Computer VC 20 und C 64 haben als Verbindung zur Außenwelt an der Gehäuse-Rückseite den User-Port. Dieser 8 Bit breite parallele Port besteht aus acht programmierbaren Anschlüssen PB0 bis PB7, die als Eingabe- oder Ausgabeleitung geschaltet werden können. Weiterhin stehen noch die Eingabe-Handshakeleitung FLAG2 (CA1 beim VC 20) und die Ausgabe-Handshakeleitung PA2 (VC 20: CB2) für diese acht Leitungen zur Verfügung.

Diese Anschlüsse sind vom User-Port aus mit dem universellen Interface-Baustein (VIA) 6522 direkt verbunden (VIA = Versatile Interface Adapter). Sie erlauben den Anschluß von externen Schaltungen und Geräten. Das User-Port-Display, das noch als universeller Ein-/Ausgabe-Adapter verwendet werden kann, wird an dieser Platine-Stekkerleiste am besten über einen Platinenstecker der Bezeichnung 251-12-50-170 angeklemt. Die einzelnen Port-Leitungen PB0 bis PB7 haben eine bestimmte binäre Wertigkeit, wie in Tabelle 1 gezeigt. Die Belegung des User-Ports entnehmen Sie bitte dem Handbuch zum C 64. Die Handshakeleitung FLAG2 (CA1) kann bei der Eingabe, die Handshakeleitung PA2 (CB2) bei der Ausgabe von Datenwörtern als Quittierungssignal benutzt werden.

Die Schaltung des User-Port-Displays

Zur Überwachung der logischen Zustände bei der Programmierung der einzelnen Ports wurde die in Bild 1 gezeigte Schaltung entworfen. Dadurch werden die Eingabe- und Ausgabe-Datenwörter an den Ports PB0 bis PB7 durch Leuchtdioden (LEDs) sichtbar gemacht. Gleichermaßen werden die logischen Zustände der Handshakeleitungen an-

gezeigt. Eine Stückliste für den I/O-Adapter (Input/Output) finden Sie in Tabelle 2. High-Pegel, also logisch »Eins«, werden durch Leuchten, Low-Pegel (logisch »Null«) durch Verlöschen der entsprechenden LED signalisiert.

Die Schaltung besteht aus zehn Invertern (2 x 74LS04, Bild 2), an die über je einen Widerstand von 330 Ohm die LED an +5 V angeschlossen sind. Ein Inverter kehrt das Eingangssignal um. Aus +5 V werden 0 V, aus 0 V werden +5 V.

Spannungsversorgung und Aufbau

Zur Spannungsversorgung für die Anzeigeeinheit wird die an Pin 2 des User-Port anliegende Spannung von +5 V benutzt. Es ist darauf zu achten, daß der Stromverbrauch der gesamten Anordnung 100 mA nicht übersteigt. In Fällen größerer Stromentnahme wird ein separates Netzteil erforderlich.

Port	Wertigkeit	binär
PB0	1	2 ⁰
PB1	2	2 ¹
PB2	4	2 ²
PB3	8	2 ³
PB4	16	2 ⁴
PB5	32	2 ⁵
PB6	64	2 ⁶
PB7	128	2 ⁷

Tabelle 1. Die binären Wertigkeiten der Ein-/Ausgabeleitungen

Aufbau der Schaltung

Da es sich bei dieser Schaltung um eine relativ einfach aufzubauen-e Einheit handelt, wurde auf ein Platinenlayout verzichtet. Statt dessen kann die Schaltung auf eine Lochrasterplatte (Rastermaß: 2,51 mm) aufgebaut werden. Die entsprechenden Verbindungen werden dann mit Drahtverbindungen ausgeführt. Entweder durch Löten von Drahtstücken, oder mit Wire-Wrapping. Bei der letzteren Technik umwickelt man mit einer speziellen Maschine die Anschlußbeine mit Kupferlackdraht und stellt so die Verbindungen her. Die Anzeigeeinheit und die 13polige Buchsenleiste kann in einem kleinen Gehäuse untergebracht werden. Über ein 12adriges Kabel wird nun der User-Port mit dem Display verbunden. Zusätzlich ist es sinnvoll, die +5-V-Spannung und die Masse auf separate Buchse zu legen, um ein eigenes Netzteil anschließen zu können. Pin 2 des User-Ports muß dann natürlich abgeklemmt werden.

Die Programmierung des User-Ports

Wie bereits erwähnt, lassen sich die acht Port-Leitungen als Ein-/Ausgabeleitungen programmieren. Nach dem Einschalten des Computers werden alle Ports automatisch auf Eingabe geschaltet. Dies läßt sich leicht daran erkennen, daß alle acht LEDs des User-Port-Displays leuchten. Über das Datenrichtungsregister, Adresse DDRB = 56579 (VC 20 = 37138), lassen sich die einzelnen acht Ports so programmieren, daß sie entweder als Ein- oder Ausgabeleitung arbeiten. Mit dem Befehl POKE DDRB,X wird die Programmierung vorgenommen. Hierbei ist »X« eine Dezimalzahl zwischen 0 und 255. Die der Dezimalzahl entsprechende Binärzahl erscheint dann am User-Port an den Ausgängen PB0 bis PB7. In der Binärzahl stellt die logische »0« einen Eingabe-, die logische »1« einen Ausgabeport dar. Als logische Pegel liegen allerdings die invertierten Pegel an, da der C 64 und VC 20-Computer mit negativer Logik arbeitet; also High-Pegel bei logisch »0« und umgekehrt.

Mit POKE DDRB,255 werden alle acht Ports zu Ausgabeleitungen, da dezimal 255 in Binärdarstellung 1111111 ist. Die Programmierung mit POKE DDRB,0 ergibt entsprechend 00000000 und alle acht Ports sind Eingabeleitungen. POKEt man in die Speicherstelle DDRB die Dezimalzahl 240, so ergibt dies die Binärzahl 11110000. Dies bedeutet, daß die Ports PB0 bis PB3 Eingabeleitungen und die Ports PB4 bis PB7 Ausgabeleitungen sind. So lassen sich durch POKEn einer Zahl zwischen 0 und 255 in die Speicherstelle DDRB die jeweils gewünschten Ein-/Ausgabekombinationen einstellen.

Die Ausgabe von Daten

Nachdem das Datenrichtungsregister auf Ausgabe programmiert ist, können die Daten über die Speicherstelle PRB = 56577 (VC 20: PRB = 37136) ausgegeben werden. Der entsprechende Befehl hierzu lautet POKE PRB,X. Auch hier kann X eine Dezimalzahl zwischen 0 und 255 sein. Die entsprechende Binärzahl erscheint an den Ports und bringt die dazugehörigen LEDs zum Leuchten. High-Pegel bedeutet jetzt Aufleuchten der LED. Mit Listing 1 kann dies leicht überprüft werden. In Zeile 10 werden alle Ports PB0 bis PB7 auf Ausgabe geschaltet. In Zeile 20 erfolgt die Zahleneingabe (Zahl zwischen 0 und 255). In Zeile 30 wird


```

0 REM LISTING 1
3 REM VC 20: DDRB=37138 / C 64: DDRB=56579
5 REM VC 20: PRB=37136 / C 64: PRB=56577
10 POKE DDRB,255
20 INPUT "ZAHL";X
30 POKE PRB,X
40 GOTO 20

```

READY.

Listing 1. Ausgabe einer Binärzahl am User-Port

```

0 REM LISTING 2 C 64
5 DDRB=56579:PRB=56577
10 POKE DDRB,255
20 POKE PRB,127
30 POKE 56576,147
40 FOR I=1 TO 100:NEXT I
50 POKE 56576,151

```

READY.

Listing 2a. Demonstration der
Handshakeleitung beim C 64

```

0 REM LISTING 2 VC 20
5 DDRB=37138:PRB=37136
10 POKE DDRB,255
20 POKE PRB,127
30 POKE 37148,220
40 FOR I=1 TO 100:NEXT I
50 POKE 37148,225

```

READY.

Listing 2b. Demonstration der
Handshakeleitung beim VC 20

```

0 REM LISTING 3
3 INPUT "1) VC 20 2) C 64";COMP
5 ON COMP GOTO 7,9,3
7 DDRB=37138:PRB=37136:X=37149:Y=2:GOTO10
9 DDRB=56579:PRB=56577:X=56589:Y=16:GOTO10
10 POKE DDRB,0
20 PRINT"WARTEN AUF DATENFREIGABE"
30 WAIT X,Y
40 PRINT"AUSGABE DES DATENWORTES";PEEK(DRB)
50 GOTO20

```

READY.

Listing 3. So einfach ist das Einlesen mit
Handshake von Daten am User-Port

```

0 REM LISTING 4
3 INPUT "1) VC 20 2) C 64";COMP
5 ON COMP GOTO 7,9,3
7 DDRB=37138:PRB=37136:X=37149:Y=2:GOTO10
9 DDRB=56579:PRB=56577:X=56589:Y=16:GOTO10
10 POKE PRB,0
20 POKE DDRB,240
30 PRINT"WARTEN AUF DATENFREIGABE"
40 WAIT X,Y
50 IF PEEK(PRIB)=11 THEN POKE PRB,16:END
60 GOTO 50

```

READY.

Listing 4. Der User-Port kann auch gleichzeitig
als Ein- und Ausgabe-Schnittstelle benutzt werden

```

0 REM LAUFLICHT-DEMO
3 INPUT "1) VC 20 2) C 64";COMP
5 ON COMP GOTO 6,8,3
6 DDRB=37138:PRB=37136:YY=37148:
7 ZZ=255:Z1=220:GOTO10
8 DDRB=56579:PRB=56577:YY=56576:
9 ZZ=151:Z1=147:GOTO10
10 POKE DDRB,255
20 FORX=0TO7
30 Y=2^X
40 POKE PRB,Y
50 FORI=1TO100:NEXTI
60 NEXTX
70 POKE PRB,0
80 POKE YY,ZZ
90 FORI=1TO100:NEXTI
100 POKE YY,Z1
110 GOTO 20

```

READY.

Listing 5. Laufflicht am User-Port

den. Bei einem Flankenwechsel von High nach Low werden die anliegenden Daten eingelesen. Mit Hilfe von Listing 3 läßt sich dieser Vorgang bewerkstelligen.

In Zeile 10 werden die Ports auf Eingabe programmiert. In Zeile 30 wird das Basic-Programm solange angehalten, bis der Flankenwechsel am Port FLAG2 (CA1) erfolgt. Danach wird das Programm wieder fortgesetzt, und in Zeile 40 das angelegte Datenwort auf dem Bildschirm angezeigt.

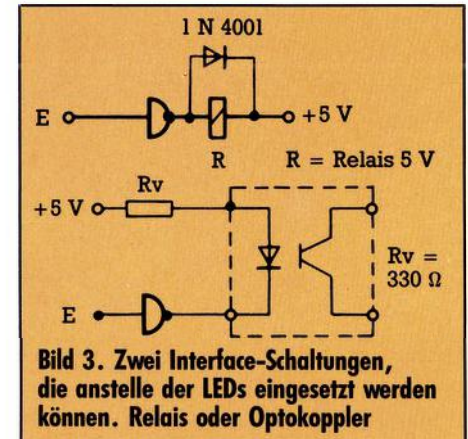


Bild 3. Zwei Interface-Schaltungen,
die anstelle der LEDs eingesetzt werden
können. Relais oder Optokoppler

Daten-Ein- und -Ausgabe kombiniert

Mit der Kombination von Daten-Ein- und Ausgabe können vielfältige Steuer- und Kontrollaufgaben gelöst werden. Hierbei werden Datenwörter eingelesen, im Computer verarbeitet und zur Steuerung wieder ausgegeben. Listing 4 zeigt dies beispielhaft. In Zeile 10 wird eine eventuell bestehende Kombination gelöscht. In 20 werden die Ports PB0 bis PB3 auf Eingabe und die Ports PB4 bis PB7 auf Ausgabe programmiert. In Zeile 40 wird das Programm solange angehalten, bis ein Flankenwechsel von High nach Low an Port FLAG2 (VC 20: CA1) erfolgt. In Zeile 50 wird die eingegebene Binärkombination mit der vorgegebenen Dezimalzahl 11 verglichen. Bei Erfüllung dieser Bedingung wird ein Steuerbefehl zum Port PB4 gesendet, der dann irgendeine Funktion auslösen kann. Bei Nichterfüllung der vorgegebenen Bedingung wird in Zeile 60 eine Rücksprunganweisung zur Zeile 50 gegeben. Mit Listing 5 können Sie den C 64 in eine Lichtorgel umfunktionieren.

Ausblick

In Bild 3 sind zwei Interface-Schaltungen gezeigt, die den C 64 oder VC 20 mit der Außenwelt verbinden. Hüten Sie sich bei Ihren Experimenten vor Netzspannung. Nicht nur der Computer kann damit ins Jenseits befördert werden.

(Dipl.-Ing. Richard Kurzhals/hm)