

```

programm : sequencer          c480 c779
-----
c480 : 4c 39 c7 ff 00 00 00 c8 ec
c488 : 0c c8 e0 c8 fd c8 20 c9 84
c490 : 43 c9 18 18 18 00 00 43
c498 : 0d 0d 0d 00 00 00 20 30
c4a0 : 20 01 02 04 00 00 00 42
c4a8 : 00 00 01 1e 86 18 8e 8b 27
c4b0 : 7e 7e 9f fa a8 06 b3 ac af
c4b8 : bd f3 c8 e6 d4 8f e1 f8 c1
c4c0 : ee 2e fd 7d c6 7d c6 7d 63
c4c8 : c6 7d c6 7d c6 7d c6 7d 1c
c4d0 : c6 7d c6 fc c6 95 c5 31 38
c4d8 : ea ad 14 03 8d d7 c4 ad 04
c4e0 : 15 03 8d d8 c4 a9 1e 8d 22
c4e8 : 14 03 a9 c5 8d 15 03 a9 82
c4f0 : e0 8d 06 dc a9 2e 8d 07 04
c4f8 : dc a9 82 8d 0d dc a9 11 7c
c500 : 8d 0f dc 60 ae 84 c4 bd f6
c508 : 8d c4 85 ff bd 8c c4 85 b7
c510 : fe a0 00 b1 fe fe 8c c4 38
c518 : d0 03 fe 8d c4 60 ad 0d fb
c520 : dc 48 29 02 d0 03 4c a1 45
c528 : c5 a5 fe 48 a5 ff 48 a5 4f
c530 : 01 29 fe 85 01 ad 0d dc a2
c538 : a2 0e 8e 85 c4 a2 04 8e c4
c540 : 84 c4 a2 02 8e 83 c4 20 68
c548 : 4d c6 ce 83 c4 30 12 ce d0
c550 : 84 c4 ce 84 c4 ad 85 c4 d4
c558 : 38 e9 07 8d 85 c4 4c 47 37
c560 : c5 ad a7 c4 c9 07 d0 0c ae
c568 : ad a9 c4 d0 07 20 df c5 b1
c570 : 90 c6 b0 1e a2 00 bd a4 be
c578 : c4 f0 03 20 b3 c5 a2 01 6f

c580 : bd a4 c4 f0 03 20 b3 c5 6a
c588 : a2 02 bd a4 c4 f0 03 20 4f
c590 : b3 c5 6c d5 c4 a5 01 09 8b
c598 : 01 85 01 68 85 ff 68 85 ae
c5a0 : fe 68 0d 0d dc 29 01 f0 b4
c5a8 : 04 58 6c d7 c4 68 a8 68 f1
c5b0 : aa 68 40 8e 83 c4 8a 0a 0d
c5b8 : 8d 84 c4 0a 0a 38 ed 83 1b
c5c0 : c4 8d 85 c4 ad 8a c4 85 92
c5c8 : fe ad 8b c4 85 ff ac 84 2c
c5d0 : c4 b1 fe 99 8c c4 c8 b1 d5
c5d8 : fe 99 8c c4 4c 7d c6 ad 86
c5e0 : 88 c4 18 69 03 8d 88 c4 46
c5e8 : 90 03 ee 89 c4 ad 88 c4 4c
c5f0 : 85 fe ad 84 c4 85 ff a0 4b
c5f8 : 00 b1 fe 8d 8a c4 c8 b1 97
c600 : fe f0 31 8d 8c c4 85 ff 69
c608 : ad 8a c4 85 fe a0 00 b1 34
c610 : fe 99 8c c4 c8 06 d0 e3
c618 : fe a9 00 8d 98 c4 8d 99 ae
c620 : c4 8d 9a c4 8d a7 c4 a9 66
c628 : 01 8d 9b c4 8d 9c a8 5b
c630 : 9d c4 18 60 ad aa c4 f0 67
c638 : 0f ad 86 c4 8d 88 c4 ad e3
c640 : 87 c4 8d 89 c4 4c ed c5 b0
c648 : 20 16 c7 38 60 ae 83 c4 7f
c650 : bd 98 c4 f0 22 de 98 c4 ad
c658 : f0 01 60 ad a8 c4 d0 0b a1
c660 : bd 9e c4 29 fe ae 85 c4 c8
c668 : 9d 04 d4 ae 83 c4 bd 95 93
c670 : c4 f0 0a 9d 9b c4 60 de 02
c678 : 9b c4 f0 01 60 ae 83 c4 e5
c680 : a9 00 9d a4 c4 20 04 c5 0e
c688 : c9 00 d0 12 ae 83 c4 ad 3d
c690 : a7 c4 1d a1 c4 8d a7 c4 f6

c698 : a9 01 9d a4 c4 60 10 15 77
c6a0 : c9 f8 90 29 29 07 0a aa 77
c6a8 : bd c3 c4 85 fe bd c4 c4 a3
c6b0 : 85 ff 6c fe 00 c9 61 b0 65
c6b8 : 09 ae 83 c4 9d 92 c4 4c ac
c6c0 : 7d c6 e9 61 ae 83 c4 9d 9c
c6c8 : 95 c4 4c 7d c6 a8 ae 83 f5
c6d0 : c4 bd 92 c4 9d 98 c4 98 93
c6d8 : 29 0f c9 0c 90 01 60 0a 23
c6e0 : aa 98 4a 4a 4a 4a a8 bd c7
c6e8 : ac c4 85 fe bd ab c4 c0 05
c6f0 : 0e f0 06 46 fe 6a c8 d0 c9
c6f8 : f6 6c d3 c4 ae 85 c4 9d 17
c700 : 00 d4 a5 fe 9d 01 d4 ae 46
c708 : 83 c4 bd 9e c4 09 01 ae 26
c710 : 85 c4 9d 04 d4 60 a9 00 d6
c718 : 8d 0f dc a9 02 8d 0d dc 14
c720 : ad 9e c4 8d 00 d4 8d 07 ea
c728 : d4 8d 0e d4 ad d7 c4 8d a9
c730 : 14 03 ad 88 c4 8d 15 03 5f
c738 : 60 78 a5 fe 48 a5 ff 48 80
c740 : ad f0 6c 8d 88 c4 ad 87 88
c748 : c4 8d 89 c4 20 ed c5 90 77
c750 : 02 b0 1e a9 08 a2 49 8d 3d
c758 : 05 d4 8e 06 d4 8d 0c d4 bf
c760 : 8e 0d d4 8d 13 d4 8e 14 96
c768 : d4 a9 0f 8d 18 d4 20 d9 e3
c770 : c4 68 85 ff 68 85 fe 58 29
c778 : 60 ff 00 ff 00 ff 00 00 d8
    
```

Listing 1. Der Sequencer. Bitte mit dem MSE eingeben.

```

programm : musik          c800 c950
-----
c800 : 15 c8 00 31 c8 00 31 c8 82
c808 : 00 a4 c8 00 a4 c8 00 e9 f1
c810 : c8 00 00 00 00 1b c8 21 17
c818 : c8 27 c8 60 61 ef 48 ef 49
c820 : 00 60 61 ef 48 ef 00 0c c3
c828 : 6d 93 9a 93 8a 93 9a 93 4f
c830 : 00 37 c8 69 c8 9d c8 18 f8
c838 : 61 ef 0c ba 24 ef 0c ba 53
c840 : 24 ef 0c 6d ba b6 b6 ba be
c848 : b9 b5 b5 b9 18 61 b9 48 84
c850 : ef 0c c3 24 ef 0c c3 24 71
c858 : ef 0c 6d c3 bb bb c3 c2 4f
c860 : ba ba c2 18 61 c2 30 ef f8
c868 : 00 0c 61 b3 b5 b6 b5 b3 8c

c870 : b5 b6 b5 b3 b5 6d b6 b3 6d
c878 : b3 b6 b5 b0 b0 b5 18 61 e6
c880 : b5 30 ef 0c b8 ba bb ba 90
c888 : b8 ba bb ba b8 ba 6d bb 72
c890 : b8 b8 bb ba b5 b5 ba 18 0f
c898 : 61 ba 30 ef 00 c8 6d 8a 8c
c8a0 : 93 9a 93 00 aa c8 b8 c8 cb
c8a8 : de c8 18 61 ef 79 d1 d1 d2
c8b0 : c8 c8 c6 c6 c1 61 c1 00 95
c8b8 : 0c 61 c6 c8 ca c8 c6 c8 df
c8c0 : 18 ca 0c c1 c3 c5 c3 c1 76
c8c8 : c3 18 c5 0c bb c1 c3 c1 e7
c8d0 : bb c1 18 c3 0c b6 b8 ba b9
c8d8 : b8 b6 b8 18 ba 18 61 eb
c8e0 : ef 48 79 b6 b1 ab 61 a6 74
c8e8 : 00 ef c8 08 c9 2c c9 18 68

c8f0 : 61 ef 60 ef ef ef ef 06 a9
c8f8 : 67 9a a5 aa aa b5 ba ba 24
c900 : c5 18 61 ca ef ef ef 00 c1
c908 : 0c 61 a5 a6 a8 a6 a5 ef 39
c910 : a6 a5 a3 ef a5 a3 a1 ef 4d
c918 : a3 a1 a0 ef 60 ef 0c a3 af
c920 : a1 9b 3c ef 60 95 18 b5 1a
c928 : ef ef ef 00 0c 61 95 96 5a
c930 : 98 96 95 ef 96 95 93 ef bb
c938 : 95 93 91 ef 93 91 90 ef e1
c940 : 60 ef 0c 93 91 8b 3c ef 54
c948 : 60 8a 18 aa ef ef ef 00 87
    
```

Listing 2. Ein Musik-Demo. Bitte mit dem MSE eingeben.

C 64 extern — Der Weg nach draußen (Teil 3)

Nachdem wir in der letzten Folge die Programmierung der Control-Ports abgeschlossen haben, wenden wir uns heute einer der vielseitigsten Schnittstellen des C 64/VC 20 zu — dem User-Port.

Der User-Port führte neben dem Expansion-Port lange Zeit ein Schattendasein, denn an ihm können keine Programm-Module verwendet werden. Mittlerweile hat der User-Port aber weit aufgeholt. Er wird für Steuerzwecke, zum Anschluss eines Druckers, für die RS232 und auch zum Programmieren von EPROMs mit einem Zusatzgerät

verwendet. Sehen wir uns diese interessante Schnittstelle etwas genauer an. Die Anschlußbelegung der User-Ports von C 64 und VC 20 zeigen die Bilder 1 und 2. Beachten Sie bitte, daß die Anschlüsse an der Ober- und Unterseite des User-Ports verschiedene Funktionen haben. Klemmen Sie deshalb dort niemals eine Krokodilklemme

1	2	3	4	5	6	7	8	9	10	11	12
A	B	C	D	E	F	H	J	K	L	M	N
PIN	BELEGUNG	PIN	BELEGUNG								
1	GND	A	GND								
2	+5V, max. 100 mA	B	FLAG2								
3	RESET	C	PB0								
4	CNT1	D	PB1								
5	SP 1	E	PB2								
6	CNT2	F	PB3								
7	SP 2	H	PB4								
8	PC2	J	PB5								
9	SER. ATN IN	K	PB6								
10	9V AC, max. 100 mA	L	PB7								
11	9V AC, max. 100mA	M	PA2								
12	GND	N	GND								

Bild 1. Anschlußbelegung des C 64 User-Ports. (Bei Aufsicht auf die Computerrückseite)

oder vergleichbares an! Ich möchte Ihnen nun zuerst zeigen, wie der 8-Bit-Parallelport am User-Port programmiert wird. Dieser Port bietet sehr vielseitige Anwendungsmöglichkeiten, obwohl er einfach zu programmieren ist. Angesprochen wird er über die acht Leitungen PB0 bis PB7 (Pin C bis L).

Über den Parallelport kann man Daten einlesen und Daten ausgeben. Das Praktische an dieser Tatsache ist, daß man die Richtung der Daten für jede der acht Leitungen einzeln bestimmen kann. Warum es aber gerade acht Leitungen gibt, hat einen einfachen Grund: Zur Steuerung des Ports gibt es im Computer zwei verschiedene Speicherstellen. Jeder Leitung am Port ist in jeder dieser Speicherstellen genau ein Bit zugeordnet. Daher also acht Leitungen, gleich acht Bit, gleich ein Byte.

Welche Bedeutungen haben nun diese beiden Speicherzellen? Die erste nennt sich »DDR«. Das heißt »Data Direction Register« (Datenrichtungsregister). Hier wird, wie der Name schon sagt, die Datenrichtung festgelegt. Aber wie? Dem Bit X im DDR ist die Datenleitung PBX zugeordnet. Ist zum Beispiel Bit 2 in diesem Register auf 1, dann ist die Leitung PB2 als Ausgang deklariert. Eine 0 macht die entsprechende Leitung zum Eingang. So einfach ist das!

Die zweite bereits genannte Speicherstelle enthält die eigentlichen Daten. Man bezeichnet sie als Portregister.

Nehmen wir einmal an, wir hätten in das Datenrichtungsregister die Zahl 255 eingeschrieben, also alle Leitungen als Ausgänge programmiert. Eine 255 im Portregister legt dann High-Pegel auf alle acht Datenleitungen. Eine 0 im Portregister bewirkt GND auf allen Datenleitungen. Man kann also wie im Datenrichtungsregister bitweise den Spannungszustand am Ausgang bestimmen (1=high, 0=low). Hierbei ist Bit 0 wieder PB0 zugeordnet. Die Spannungspegel, die wir am Port erhalten, können wir dann »extern« verarbeiten, doch davon später.

Im zweiten Beispiel benutzen wir den Parallelport als Eingang. Nehmen wir an, wir hätten in das Datenrichtungsregister eine 0 eingeschoben, also alle

1	2	3	4	5	6	7	8	9	10	11	12
A	B	C	D	E	F	H	J	K	L	M	N

PIN	BELEGUNG	PIN	BELEGUNG
1	GND	A	GND
2	+5V, max. 100 mA	B	CB1
3	RESET	C	PB0
4	JOY 0	D	PB1
5	JOY 1	E	PB2
6	JOY 2	F	PB3
7	LIGHT PEN	H	PB4
8	CASSETTE SWITCH	J	PB5
9	SERIAL ATN IN	K	PB6
10	9V AC, max. 100 mA	L	PB7
11	9V AC, max. 100 mA	M	CB2
12	GND	N	GND

Bild 2. Anschlußbelegung des VC 20 User-Ports. (Bei Aufsicht auf die Computerrückseite)

Datenleitungen als Eingänge definiert. Jetzt müssen wir »externe« Daten an den Port anlegen. Das geschieht durch Verbinden der Datenleitungen entweder mit +5 V oder GND der

Computerversorgungsspannung. Dabei entspricht einer »0« GND und eine »1« +5V. Eine »1« entsteht auch, wenn eine Leitung nicht angeschlossen, also weder mit +5 V noch mit GND verbunden wird. Das spart oft viel Schaltungsaufwand. Die an den Port angelegte Bitkombination kann jederzeit durch Abfragen des Portregisters ausgelesen werden. Der neue Zustand der Datenleitungen wird, wenn sich etwas geändert hat, jeweils sofort und automatisch in das Portregister übernommen.

Die beiden Funktionen Eingänge/Ausgänge dürfen, wie schon gesagt, beliebig gemischt werden. So ist beispielsweise möglich, die Leitungen 2, 5 und 7 als Ausgänge und alle restlichen Leitungen als Eingänge zu definieren. Dazu müßte man im Datenrichtungsregister die Zahl 164 ablegen (164 = Bits 2, 5 und 7 gesetzt). Die Information über die Spannungszustände an den Ausgängen müßte in die Bits 2, 5 und 7 des Portregisters gePOKET und die Information über die Eingänge aus den Bits 0, 1, 3, 4

und 6 des Portregisters ausgelesen werden.

Nach soviel grauer Theorie wollen wir endlich wieder etwas programmieren. Bevor es losgeht, müssen Sie noch wissen, wie die Adressen der eben besprochenen Speicherstellen lauten. Sie sind in Tabelle 1 aufgelistet. Jetzt haben wir alles nötige Rüstzeug, um voll in eine sinnvolle Anwendung des Computers einzusteigen: Programmieren wir eine Alarmanlage!

Alarmanlage per User-Port

Tippen Sie Listing 1 ab und starten Sie das Programm mit RUN. Auf dem Bildschirm erscheint der Kontaktplan des Parallelports. Hoffentlich haben Sie beim Einkauf für die letzte Folge an den Stecker für den User-Port gedacht, den brauchen wir nämlich jetzt.

Bringen Sie an einem der vier Eckkontakte (=GND, siehe Bild 1) des User-Port-Steckers einen Draht an, und stecken Sie den Stecker in den User-Port. Zum Anbringen des Drahtes ist Löten wohl das Beste. Keinesfalls dürfen Sie einen anderen Anschluß versehentlich berühren, bezie-

hungsweise mit GND in Verbindung bringen.

Nehmen Sie sich nun das andere Ende des Drahtes zur Hand, und berühren Sie damit eine der Datenleitungen PB1 bis PB7. Bitte nicht die Datenleitung PB0, da diese vom Programm als Ausgang (Alarmanzeiger) benutzt wird (was wir mit diesem machen können, sehen wir in der nächsten Folge). Welche Pins die richtigen sind, können Sie Bild 1 und 2 entnehmen.

Beim Berühren erklingt ein Alarmton, der Bildschirmrahmen ändert die Farbe, und der Alarmanzeiger (PB0) geht auf high (+5 V). Zusätzlich wird auf dem Bildschirm angezeigt, welche Kontakte den Alarm ausgelöst haben. Wenn Sie keine der Datenleitung mehr mit dem Draht berühren, die Eingänge also alle High-Signal erhalten, werden alle Alarmsignale sofort gelöscht.

Werfen wir einen Blick auf das Listing. Nach der Initialisierung der Variablen, der Portregister, des SID-Chips und des Bildschirms wird in Zeile 390 das Portregister (Variable R) so lange ausgelesen, bis ein anderer Wert als 254 auftaucht. Warum 254? 254 ist die Summe der Wertigkeiten der Bits 1 bis 7, also aller Bits des Portregisters ohne den Alarmanzeiger. Das ist der Ruhezustand der Alarmanzeige; alle Kontakte sind geöffnet (=High-Pegel).

Wird ein Alarm ausgelöst, dann werden ab Zeile 430 sämtliche Alarmanzeiger eingeschaltet. Ab Zeile 470 wird ein neuer Kontaktplan ausgegeben. Dabei wird jeder Kontakt auf Alarm überprüft und das Ergebnis angezeigt.

Anschließend wartet der Computer auf das Alarmende. Wenn noch Alarm besteht, wird wieder ein neuer Kontaktplan ausgegeben, weil sich am Kontaktzustand etwas geändert haben kann. Andernfalls werden die Alarmanzeiger ausgeschaltet und es wird zur Bildschirminitialisierung zurückgesprungen.

Wer sich das Programm genau ansieht, wird schnell feststellen, daß es so, wie es hier abgedruckt ist, zum realen Einsatz als Alarmanlage weniger geeignet ist. Dazu hat es verschiedene Schwächen. Zum einen wird

	DATEN- RICHTUNGS- REGISTER	PORTREGISTER
VC 20	37 138	37 136
C 64	56 579	56 577

Tabelle 1. Zusammenstellung der User-Port-Register

	DATEN- RICHTUNGS- REGISTER	PORTREGISTER
C 64, PORT 1	56 323	56 321
C 64, PORT 2	56 322	56 320
VC 20	37 139 / 37 154	37 137 / 37 152

Tabelle 2. Zusammenstellung der Control-Port-Register

	AN-SCHLUSS	PIN	BIT	DATEN-RICHTUNGS-REGISTER	PORTREGISTER
RECHTS	JOY 3	4	3	56323	56321
LINKS	JOY 2	3	2	56323	56321

Tabelle 3. Daten zur Abfrage der Paddle-Knöpfe beim Port 1 des C 64

nicht mit Sicherheit jeder Alarm erkannt: Wenn der Alarmimpuls sehr kurz ist, kann es passieren, daß der C 64 gerade etwas anderes tut, als das Portregister auszulesen, zum Beispiel den Rücksprung nach 390 ausführen. Außerdem ist der eigentlich alarmauslösende Zustand am Portregister nicht der, der ausgewertet wird, weil der auswertende Zustand erst nach dem Alarmauslösen in Zeile 420 ermittelt wird. In der Zwischenzeit kann sich schon etwas verändert haben. Das genannte trifft aber nur auf sehr kurze Impulse zu.

Zum anderen ist die Auswertung des Alarms viel zu einfach gehalten, eben nur als Beispiel. Wenn hier ein Alarm von einer Sekunde auftritt, wird auch nur eine Sekunde lang Alarm gegeben, was natürlich viel zu wenig ist.

Codeschloß

Kommen wir zu einem weiteren Programmbeispiel, dem Codeschloß in Listing 2. Wir haben es wieder mit Raumsicherung zu tun. Wieder wird die Datenleitung PB0 als Ausgang und die restlichen Datenleitungen als Eingänge benutzt.

Was leistet das Programm? Sie müssen durch Verbinden der Datenleitungen PB2 bis PB7 einen sechsstelligen Code eingeben. Wenn Sie keinen Fehler gemacht haben, wird die Datenleitung PB0 auf high gelegt, also beispielsweise eine Tür geöffnet oder eine geheime Anlage eingeschaltet. Wie Sie solche Dinge ansteuern können, erfahren Sie im nächsten Teil dieses Kurses. Über die Leitung PB1 wird das Codeschloß zurückgesetzt (sowohl nach dem Auslösen des Schlosses als auch nach einer Fehlereingabe). Jede Berührung an den Eingängen wird durch einen Signalton quittiert.

In den Zeilen 250/260 werden zwei Variablen deklariert, deren Bedeutung ich kurz erläutern möchte. CL enthält die Länge des einzugebenden Codes, C1\$ den Code selbst. Den Inhalt dieser beiden Variablen können Sie nach Belieben verändern. Der von mir gewählte Code ist bewußt sehr primitiv.

In C2\$ wird der Code aufgebaut, den Sie durch Berühren

1	AN-SCHLUSS	PIN	BIT	DATEN-RICHTUNGS-REGISTER	PORTREGISTER
RECHTS	JOY 3	4	7	37154	37152
LINKS	JOY 2	3	4	37139	37137

Tabelle 5. Daten zur Abfrage der Paddle-Knöpfe beim VC 20

der »externen« Kontakte eingeben. Der große Vorteil hierbei ist, daß die Eingabeeinheit für den Code an einer anderen Stelle (zum Beispiel im Nebenraum) als der Computer sein kann. Man wird also von der Computertastatur unabhängig. Mit einem langen Code wird ein Knacken theoretisch unmöglich. Da sich das Programm durch REM-Anweisungen selbst dokumentiert, bleibt nur noch zu sagen, daß es der Einfachheit halber bei jedem »RESET« an Kontakt PB1 einfach neu mit RUN gestartet wird.

Auch bei diesem Programm ist nur eine Minimallösung angegeben. Man kann es um viele Raffinessen erweitern, so daß zum Beispiel ein Einbruchversuch als solcher gedeutet wird und so weiter.

Nachdem wir nun wissen was sich am User-Port abspielt, wollen wir noch einmal einen Schritt zurückgehen. Im ersten Teil dieses Kurses sind ein paar Fragen offen geblieben, die ich jetzt beantworten möchte.

Da war als erstes die Sache mit der Umschaltung des Computers auf Joystickeingabe. Vielleicht haben Sie schon eine Ähnlichkeit zwischen Joystick- und Parallelport-Programmierung entdeckt. Bei beiden mußte etwas umgeschaltet werden. Dann gab es zwei mögliche Zustände: 0 oder 1! Die Umschaltung des Computers auf Joystickeingabe ist nichts anderes als die Programmierung des zugehörigen Datenrichtungsregisters. Bitte erinnern Sie sich: Wir haben bei der Umschaltung die zugehörigen Bits gelöscht, also die Datenleitungen der Control-Ports zu Eingängen gemacht. Wer darüber nachdenkt, wird schnell zu der Frage kommen, was denn passiert, wenn wir in die DDRs einfach Einsen hineinschreiben. Die Antwort ist tri-

	AN-SCHLUSS	PIN	BIT	DATEN-RICHTUNGS-REGISTER	PORTREGISTER
RECHTS	JOY 3	4	3	56322	56320
LINKS	JOY 2	3	2	56322	56320

Tabelle 4. Daten zur Abfrage der Paddle-Knöpfe beim Port 2 des C 64

sie nicht mehr ordnungsgemäß decodiert wird. Die Werte, die in bestimmte Speicherzellen immer wieder hineingeschrieben werden (siehe Folge 1) stammen auch aus der Tastaturdecodierungs-Routine.

Die Störung der Tastatur kommt nun daher, daß es dem Computer völlig gleichgültig ist, ob Sie den Joystickhebel bewegen oder auf die Tasten hämmern. Er prüft artig seine Spalten nach und kümmert sich nicht darum, woher die festgestellte Verbindung denn nun kam. Die Störung durch den Lichtgriffel hat dieselbe Ursache.

Jetzt müssen wir nur noch einen Rückstand aus der letzten Folge aufklären. Dabei handelt es sich um die Feuerknöpfe der Paddles. Ich kann Ihnen erst jetzt zeigen wie sie abgefragt werden, weil wir auch hier die Kenntnisse über die Programmierung des Parallelports benötigen. Es funktioniert nämlich schon wieder genauso: Man setze im zugehörigen DDR die entsprechenden Bits auf Null (Leitung auf Eingang schalten) und frage dann die entsprechenden Bits im zugehörigen Portregister ab. Natürlich zeigt auch hier wieder eine Null im Portregister an, daß der Feuerknopf am Paddle gedrückt wurde. In Tabelle 3 sind die nötigen Informationen für Port 1 des C 64, in Tabelle 4 die für den Port 2 und in Tabelle 5 die für den VC 20 zusammengefaßt. Dabei sind die Paddles gemäß der Position ihres Verbindungskabels am Control-Port mit links und rechts bezeichnet. Das rechte Paddle ist also an POT X, das linke an POT Y angeschlossen. Da die Feuerknöpfe der Paddles dieselben Leitungen wie der Joystick benutzen, gilt hier ebenfalls das oben Beschriebene über die gleichzeitige Benutzung von Paddles und Tastatur.

Damit wollen wir es für heute bewenden lassen. Das nächste mal sehen wir dann, wie versprochen, wo sich den Control-Ports Signale entnehmen lassen. Außerdem wollen wir uns mit Anschlußmöglichkeiten verschiedener Adapter an den User-Port beschäftigen und unter anderem einen 220 Volt-Adapter für unseren Computer bauen!

(Tobias Nicol/aw)

```

100 REM *****
110 REM *
120 REM *   A L A R M A N L A G E   *
130 REM *   -----   *
140 REM *   *
150 REM *   BY TOBIAS NICOL   *
160 REM *   *
170 REM *   NEUWIESENSTRASSE 20   *
180 REM *   *
190 REM *   6000 FRANKFURT 71   *
200 REM *   *
210 REM *****
220 :
230 REM ***** INITIALISIERUNG *****
240 R = 56577 : POKE 56579,1 : POKE R,0
250 :
260 REM ***** SID-CHIP EINSTELLEN *****
270 T = 54296 : SI = 54272
280 POKE SI+ 7,207 : POKE SI+ 8,200
290 POKE SI+13,240 : POKE SI+11,17
300 :
310 REM ***** BILDSCHIRM AUFBAUEN *****
320 PRINT "{CLR,2DOWN,5RIGHT}ALARMANLAGE"
330 PRINT "{5RIGHT}=====3DOWN}"
340 FOR A = 1 TO 7
350 PRINT"KONTAKT"A": O. K.{DOWN}"
360 NEXT A
370 :
380 REM ***** ALARM? *****
390 IF ((PEEK(R))AND 254) = 254 THEN 390
400 :
410 REM ***** ALARM! *****
420 IR = PEEK (R)
430 POKE 53280,2
440 POKE R , PEEK ( R ) OR 1
450 POKE T,15
460 :
470 REM **** KONTAKTPLAN AUSGEBEN ****

```

Listing 1. Die User-Port-Alarmanlage. Beachten Sie bitte die Eingabehinweise auf Seite 54.

```

100 REM *****
110 REM *
120 REM *   C O D E S C H L O S S   *
130 REM *   -----   *
140 REM *   *
150 REM *   BY TOBIAS NICOL   *
160 REM *   *
170 REM *   NEUWIESENSTRASSE 20   *
180 REM *   *
190 REM *   6000 FRANKFURT 71   *
200 REM *   *
210 REM *****
220 REM
230 REM ***** INITIALISIERUNG *****
240 R = 56577 : POKE 56579,1 : POKE R,0
250 T = 54296 : I = 1 : CL = 6
260 C1$ = "123456" : C2$ = ""
270 REM
280 REM ***** SID-CHIP EINSTELLEN *****
290 SI = 54272
300 POKE SI+ 7,207 : POKE SI+ 8,200
310 POKE SI+13,240 : POKE SI+11,17
320 REM
330 REM ***** BILDSCHIRM AUFBAUEN *****
340 PRINT "{CLR,DOWN,5RIGHT,BLACK}CODESCHL
OSS"
350 PRINT "{5RIGHT}-----{CYAN}"
360 PRINT "{2DOWN}GEBEN SIE BITTE DEN{DOWN
}"
370 PRINT "CODE EIN. JEDER KON-{DOWN}"
380 PRINT "TAKT WIRD MIT EINEM{DOWN}"
390 PRINT "TON BESTAETIGT."
400 PRINT "{2DOWN}{RESET: KONTAKT 2}"
410 REM
420 REM ***** KONTAKT BETAETIGT? *****
430 PR = (( PEEK (R) AND 254 ))
440 IF ( PR AND 254 ) = 254 THEN 420
450 REM
460 REM *** KONTAKTNUMMER ERRECHNEN ***
470 X = LOG ( 254 - PR ) / LOG (2) - 1
480 REM
490 REM ***** WERTE KONTROLLIEREN *****
500 IF ( X < 0 ) OR ( X > 6 ) THEN 420
510 IF X <> INT ( X ) THEN 420
520 REM
530 REM ***** RESET? *****
540 IF X = 0 THEN RUN
550 REM
560 REM * EINZUGEBENDEN CODE AUFBAUEN *

```

Listing 2. Das Codeschloß