

Der Volkscomputer und der große Bruder:

Adressenvergleich VC 20 – C 64

Leider wird man derartige Programme nur sehr selten finden, da beide Computer Grundlegendes gemeinsam haben: Die Hardwareeigenschaften werden von der Software — sprich vom Basic — kaum unterstützt. Immer dann, wenn die grafischen oder musikalischen Fähigkeiten von C 64 und VC 20 angesprochen werden, geschieht dies durch POKE-Befehle, auch wenn es nur darum geht, zum Beispiel die Farbe des Bildschirmrahmens einzustellen.

Das allein wäre ja noch nicht so schlimm. Nun ist aber die entsprechende Hardware, also Video- und Soundchip, bei beiden Computern grundlegend verschieden aufgebaut und belegt zudem noch völlig unterschiedliche Adressen im Speicher. So ist es kein Wunder, wenn das Umschreiben von C 64-Programmen auf den VC 20 (und umgekehrt) in der Regel mit erheblichen Schwierigkeiten verbunden ist. Wir wollen im folgenden versuchen, eine ganze Reihe dieser Schwierigkeiten aus dem Weg zu räumen. Als erste grobe Übersicht soll dabei die Tabelle 1 dienen. Auf weitere Einzelheiten wird im folgenden näher eingegangen.

Bevor man daran gehen kann, für den jeweils anderen Computer geschriebene Software an den eigenen C 64 oder VC 20 anzupassen, müssen die Programme erst einmal in den Speicher gebracht werden. Ein Blick auf die Modulschächte beider Computer verrät sofort, daß es mit dem Austausch von Steckmodulen nicht allzu weit her sein kann: Das Modulformat ist völlig unterschiedlich.

Leider ist auch der Softwareaustausch per Programmkassette nur über einen Umweg zu realisieren. Zwar ist das Aufzeichnungsformat auf Cassette bei beiden Computern gleich, dennoch kann der Computer die Aufzeichnungen des anderen in den meisten Fällen nicht lesen. Der Grund hierfür liegt darin, daß der VC 20 eine höhere Taktfrequenz als der C 64 hat (VC 20-Programme sind um einiges schneller als entsprechende C 64-Program-

Der Commodore 64 wird oft als der »große Bruder« des VC 20 bezeichnet. Tatsächlich sind zum Beispiel Betriebssystem, Basicinterpreter und Schnittstellen weitgehend identisch. Daher können Basicprogramme, in denen keine Befehle und Funktionen wie POKE, PEEK, SYS und USR vorkommen, praktisch unverändert übertragen werden.

	C 64	VC20 (Grundversion)	VC 20 (ab 8 K-Erw.)	VC 20
Video RAM	1024	7680	4096	Video-RAM
freier RAM-Bereich	2048	4096	4608	freier RAM-Bereich
Zeichengenerator-ROM	53248	32768	32768	Zeichengenerator-ROM
Video-Chip (VIC)	53248	36864	36864	Video & Sound Chip
Synthesizer Chip (SID)	54272	—	—	
I/O 1 (CIA #1)	56320	37136	37136	I/O 1 (VIA #1)
I/O 2 (CIA #2)	56576	37152	37152	I/O 2 (VIA #2)
FARB-RAM	55296	38400	37888	Farb-RAM
Basic-ROM	40960	49152	49152	Basic-ROM

Tabelle 1. Die Basisadressen wichtiger Speicher- und Registerbereiche

me). Dadurch gerät beim Lesen einer fremden Kassette die Synchronisation völlig aus den Fugen. Den einzigen Ausweg in dieser Situation bietet die Verwendung eines »großen« CBM der Serien 30xx oder 40xx als Vermittler. Die Taktfrequenz dieser CBM-Rechner liegt zwischen der des C 64 und der des VC 20, wodurch es zum Beispiel möglich ist, VC 20 Programme zunächst mit dem CBM zu laden, dann wieder abzuspeichern und nun wiederum mit dem C 64 zu laden.

Beim Programmaustausch per Floppy-Disk treten solche Probleme nicht auf. Hier kann man nach Herzenslust VC 20-Software in den C 64 laden oder auch umgekehrt. Hat man allerdings keine Floppy oder liegt das interessierende Programm nicht auf Diskette vor, dann bleibt in der Regel nur noch eins zu tun: Die Zähne zusammenbeißen und das Programm vom Listing abtippen. Dabei kann man dann auch gleich alle nötigen Programmänderungen vornehmen.

Leider gibt es viele Programme sowohl für den VC 20 als auch für den C 64, die man nicht durch Ändern einiger POKE-Adressen und kleinen Korrekturen am Bildschirmlayout an den jeweils anderen Computer anpassen kann.

Dazu gehören generell alle Programme, die hochauflösende Grafik verwenden. Die Prinzipien, nach dem die hochauflösende Grafik auf den beiden Computern realisiert ist, sind völlig unterschiedlich. Außerdem ist es natürlich von vorne herein völlig aussichtslos, ein C 64-Programm, das Sprites und Synthesizer-effekte einsetzt, für den VC 20 umschreiben zu wollen. Umgekehrt gibt es eine solche Einschränkung allerdings nicht — der VC 20 kennt keine Sprites und sein Tongenerator läßt sich mit dem Synthesizer des C 64 allemal simulieren.

Sehr viel Vorsicht ist geboten, wenn das Programm längere Abschnitte in Maschinensprache enthält. Oftmals müssen diese Maschinenspracheroutinen in andere Spei-

cherbereiche verschoben werden, um gemeinsam mit dem Basic-Programm laufen zu können. Wir wollen uns an dieser Stelle aber nur mit den Anpassungen bei Basic-Programmen beschäftigen.

POKE-Befehle für Farbe und Bildschirm

Selbst die einfachsten Programme enthalten in der Regel Befehle, um Rahmen- und Hintergrundfarbe des Bildschirms einzustellen. Beim VC 20 werden beide Einstellungen gleichzeitig mit einem einzigen POKE-Befehl in Register 36879 durchgeführt (Tabelle 2). Der C 64 verwendet zwei getrennte Register, nämlich 53280 für die Rahmenfarbe und 53281 für die Hintergrundfarbe. Jeweils 16 Farben sind möglich (Tabelle 3). Zum Beispiel erzeugt der Befehl POKE 36879,95 beim VC 20 einen gelben Bildschirmrahmen und einen grünen Hintergrund. Mit POKE 53280,7 : POKE 53281,5 wird dasselbe am C 64 erreicht.

Beim C 64 enthält der Bildschirm 25 Zeilen zu je 40 Zeichen; der Bildschirmspeicher belegt die Adressen 1024 bis 2023. Der zugehörige Farbspeicher geht von 55296 bis 56295. Mit POKE 1024, 1 : POKE 55296, 2 erscheint zum Beispiel ein rotes »A« in der linken oberen Bildschirmcke. Der für den Anwender verfügbare RAM-Bereich beginnt bei Adresse 2048 (Bild 1).

Beim VC 20 wird die Angelegenheit etwas komplizierter. Die Anfangsadressen von Bildschirm- und Farbspeicher sind nämlich je nach Speicherausbau unterschiedlich (Bild 2). In der Grundversion und mit der 3-KByte-Erweiterung beginnt das Video-RAM bei Adresse 7680 und geht bis Adresse 8185. Das Farb-RAM belegt dann den Bereich von 38400 bis 38905. In dieser Konfiguration liegt der Bildschirmspeicher oberhalb des für den Anwender verfügbaren RAM-Bereiches, der ab Adresse 4096 (Grundversion) beziehungsweise 1024 (3-KByte-Erweiterung) beginnt. Sobald jedoch eine Speichererweiterung von mindestens 8 KBytes eingesteckt ist, wandert das Video-RAM nach »unten« und beginnt dann ab Adresse 4096. Dies geschieht, um für Basic-Programme einen zusammenhängenden Speicherbereich von der Adresse 4608 an aufwärts zu schaffen. Eine eventuell zusätzlich vorhandene 3-KByte-Erweiterung kann in diesem Fall nicht für Basic-Programme genutzt werden. Schließ-

	BLK	WHT	RED	CYAN	PUR	GRN	BLU	YEL
Hintergrund	SCHW	WEISS	ROT	TÜR- KIS	VIO- LETT	GRÜN	BLAU	GELB
SCHWARZ	8	9	10	11	12	13	14	15
WEISS	24	25	26	27	28	29	30	31
ROT	40	41	42	43	44	45	46	47
TÜR- KIS	56	57	58	59	60	61	62	63
VIOLETT	72	73	74	75	76	77	78	79
GRÜN	88	89	90	91	92	93	94	95
BLAU	104	105	106	107	108	109	110	111
GELB	120	121	122	123	124	125	126	127
ORANGE	136	137	138	139	140	141	142	143
HELLORAN.	152	153	154	155	156	157	158	159
ROSA	168	169	170	171	172	173	174	175
HELLTÜR- KIS	184	185	186	187	188	189	190	191
HELLVIO- L.	200	201	202	203	204	205	206	207
HELLGRÜN	216	217	218	219	220	221	222	223
HELLBLAU	232	233	234	235	236	237	238	239
HELLGELB	248	249	250	251	252	253	254	255

Tabelle 2. Farbkombinationen für Bildschirmrahmen und Hintergrund beim VC 20

VC 20 und C 64		nur C 64
0	schwarz	8
1	weiß	9
2	rot	10
3	türkis	11
4	violett	12
5	grün	13
6	blau	14
7	gelb	15
		orange
		braun
		hellrot
		grau 1
		grau 2
		hellgrün
		hellblau
		grau 3

Tabelle 3. Farbwerte bei C 64 und VC 20

lich verändert auch noch das Farb-RAM seine Lage und startet jetzt bei Adresse 37888.

Der Bildschirm des VC 20 ist aufgeteilt in 23 Zeilen zu je 22 Zeichen. Insgesamt sind also 506 Bildschirmstellen vorhanden, das sind etwa halb so viele wie beim C 64. Eine Anpassung des Bildschirmlayouts ist also in fast allen Fällen erforderlich. In der Regel dürfte das kein Problem darstellen. Bei der Anpassung von VC 20-Programmen an den C 64 wird man des öfteren PRINT-Anweisungen zusammenfassen, da eine Bildschirmzeile beim C 64 fast doppelt so viele Zeichen wie eine entsprechende VC 20-Zeile aufnehmen kann. Im umgekehrten Fall ist das Einfügen von PRINT-Befehlen sinnvoll, um eine übersichtliche Bildschirmdarstellung zu erhalten. Ernste Schwierigkeiten kann es nur bei der Ausgabe von Tabellen auf dem Bildschirm geben. Eine sechsspaltige

Zahlentabelle zum Beispiel läßt sich auf dem C 64 ganz gut darstellen, beim VC 20 wird man bei derartigen Versuchen unangenehm an die arg begrenzte Zeilenbreite erinnert. In solchen Fällen kann man versuchen, weniger interessante Spalten der Tabelle einfach fortzulassen. Wenn das nicht erwünscht oder möglich ist, hilft nur noch der CMD-Befehl, um die Ausgabe der entsprechenden Tabelle auf den Drucker umzuleiten.

Das Betriebssystem von C 64 und VC 20 ist weitgehend identisch. Natürlich gibt es einige Unterschiede bei den Ein-/Ausgabeoperationen, bedingt schon alleine durch das unterschiedliche Bildschirmformat. Fast alle nutzbaren Adressen in der Zeropage oder in den anderen vom Betriebssystem benutzten Speicherbereichen haben jedoch die gleiche Bedeutung. So befindet sich zum Beispiel bei beiden Computern der Kassettenpuffer zwischen den Adressen 828 und 1019 und mit »? PEEK(43) + 256 * PEEK(44)« erhält man beidesmal die Anfangsadresse des Basic-Programms.

Eine wichtige Ausnahme von dieser Regel ist der USR-Vektor. Beim VC 20 befindet er sich in den Speicherstellen 1 und 2 am Anfang der Zeropage, in Adresse 0 steht immer ein »JMP«-Befehl. Jedesmal bei der Ausführung der USR-Funktion ver-

zweigt das Basic zur Adresse 0 und aufgrund des dort stehenden »JMP« sofort weiter zur Adresse, die in sich in den Speicherzellen 1 und 2 befindet. Beim C 64 befinden sich dagegen am Anfang der Zeropage zwei Register des 6510-Mikroprozessors, so daß der USR-Vektor hier in die

\$C000 und \$DFFC entsprechen den C 64 Adressen zwischen \$A000 und \$BFFC, die Differenz ist also gerade \$2000 oder dezimal 8192. In Adresse \$BFFD steht beim C 64 ein Sprung nach \$E000, um den RAM- und I/O-Bereich zwischen \$C000 und \$DFFF zu überbrücken. Durch diesen

noch nicht erwähnt. Gemeint sind die zur Programmierung von Musik und Geräuscheffekten benutzten Register. Leider ist die Art und Weise der Tonerzeugung bei beiden Computern völlig unterschiedlich, so daß sich keine äquivalenten POKE-Befehle angeben lassen.

Der C 64 verfügt nämlich über einen vollwertigen Synthesizer-Baustein, während der VC 20 nur »normale« Tongeneratoren besitzt. Tabelle 4 enthält für alle VC 20 Besitzer eine Übersicht über die beim C 64 zur Tonerzeugung benutzten Register. Diese Tabelle dient allerdings

Adresse			Inhalt
Stimme 1	Stimme 2	Stimme 3	
54272	54279	54286	Tonfrequenz (Low-Byte)
54273	54280	54287	Tonfrequenz (High-Byte)
54274	54281	54288	Tastaturverhältnis Rechteckgenerator (Low)
54275	54282	54289	Tastaturverhältnis Rechteckgenerator (High)
54276	54283	54290	Wellenform
54277	54284	54291	Anschlag/Abschwellen
54278	54285	54292	Halten/Ausklängen
54296			Lautstärke

Mögliche Wellenformen: Dreieck (17), Sägezahn (33), Rechteck (65), Rauschen (129)

Tabelle 4. Die wichtigsten Register des C 64-Soundchips

NOTE	WERT	NOTE	WERT
C	135	G	215
C#	143	A ^b	217
D	147	A	219
E ^b	151	B	221
E	159	H	223
F	163	C	225
F#	167	C#	227
G	175	D	228
A ^b	179	E ^b	229
A	183	E	231
B	187	F	232
H	191	F#	233
C	195	G	235
C#	199	A ^b	236
D	201	A	237
E ^b	203	B	238
E	207	H	239
F	209	C	240
F#	212	C#	241

Stimmlagen-Befehle	X =	Funktion
POKE 36878,X	0 bis 15	setzt die Lautstärke
POKE 36874,X	128 bis 255	spielt Note
POKE 36875,X	128 bis 255	spielt Note
POKE 36876,X	128 bis 255	spielt Note
POKE 36877,X	128 bis 255	Geräuscheffekte

Tabelle 5. Notenwerte und Tongeneratoradressen beim VC 20

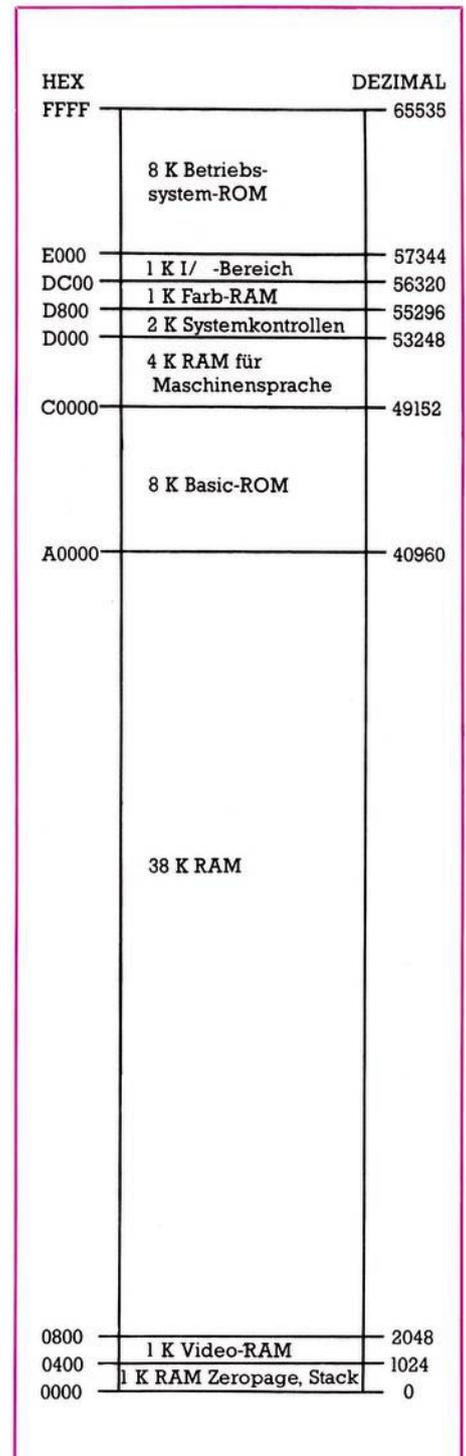
Adressen 785 und 786 verlegt wurde. Bei Programmen, welche die USR-Funktion verwenden, müssen diese unterschiedlichen Adressen unbedingt beachtet werden.

Wie aus den Bildern 1 und 2 hervorgeht, liegt der Basic-Interpreter beim C 64 in einem anderen Adreßbereich als beim VC 20. Da die Routinen aber bis auf Ausnahmen (USR-Funktion) völlig gleich sind, kann man die Adressen sehr einfach umrechnen: VC 20-Adressen zwischen

Sprungbefehl entsteht im folgenden eine Adressendifferenz um drei Bytes. Von allen C 64-Adressen zwischen \$E000 und \$E37A müssen daher diese drei Bytes abgezogen werden, um die entsprechenden VC 20-Adressen zu erhalten.

Einige häufig vorkommende POKE-Adressen wurden bisher

Bild 1. Speicherbelegung des Commodore 64 im Grundzustand. Im Bereich \$D000 bis \$DFFF sind zusätzlich 4 KByte Zeichengenerator-ROM überlagert



wirklich nur zur Orientierung, welche POKE-Befehle beim C 64 zur Tonerzeugung dienen. Eine Simulation des C 64-Synthesizers ist mit dem VC 20 nicht möglich. Es empfiehlt sich daher, bei der Programm-anpassung zunächst einmal alle derartigen POKE-Befehle fortzulassen und später eigene Sound-Routinen einzufügen.

C 64-Besitzer haben es hier etwas besser: Mit etwas Geschick und einem guten Handbuch zur Musikprogrammierung können sie dem C 64 auch alle VC 20-Töne entlocken. Als Referenz hierzu kann Tabelle 5 mit

den Daten zu den Tongeneratoren des VC 20 dienen. Im Zweifelsfalle sollte man aber auch hier eher auf originalgetreue Tonuntermalung verzichten und sich damit eine ganze Menge Arbeit sparen.

Und noch ein Punkt, wo der VC 20 hardwaremäßig benachteiligt ist: Der C 64 verfügt nämlich gleich über zwei Joystickports (mit den Adressen 56320 und 56321), während der VC 20 sich mit einem Anschluß zufrieden geben muß, der zu allem Überfluß auch noch ein recht kompliziertes Abfrageprogramm erfordert. Der Feuerknopf und die

Schalter 0, 1 und 2 des Joysticks werden nämlich beim VC 20 über VIA #1 gelesen, während der Zustand von Schalter 3 über VIA #2 abgefragt wird. Normalerweise sind Joystickabfragen in Programmen leicht zu finden. Halten Sie beim VC 20 nach PEEKs in die Speicherstellen 37137 und 37152 Ausschau und beim C 64 nach entsprechenden Abfragen der Adressen 56320 und 56321. Anschließend muß die gesamte Joystickroutine für den jeweiligen Rechner neu geschrieben werden, da die Art der Abfrage einfach zu unterschiedlich ist.

Bei der Steuerung mittels Drehreglern (sogenannten Paddles) ist die Anpassung wesentlich einfacher zu realisieren. Die Paddle-Werte werden beim C 64 aus den Registern 54297 und 54298 ausgelesen. Beim VC 20 sind es die Register 36872 und 36873. Das Umschreiben besteht hier also lediglich im Einsetzen der entsprechenden Adressen in die PEEK-Befehle.

Wo sich das Umschreiben lohnt

An dieser Stelle muß noch einmal deutlich darauf hingewiesen werden, daß die Zahl derjenigen Programme, die mit vertretbarem Aufwand vom C 64 zum VC 20 oder umgekehrt übertragen werden können, doch verhältnismäßig gering ist. Bei Spielprogrammen ist in der Regel äußerste Vorsicht geboten, da hier aus Geschwindigkeitsgründen zumeist mit Routinen in Maschinensprache gearbeitet wird. Außerdem ist bei Spielen in der Regel das Bildschirmlayout fest vorgegeben, so daß wegen der unterschiedlichen Bildschirmkapazität beider Computer sehr wahrscheinlich größere Probleme auftreten werden. Von Sprites und hochauflösender Grafik einmal ganz zu schweigen.

Dagegen gibt es viele Anwendungsprogramme, welche den Computer nicht zu einer hochspezialisierten Spielmaschine machen, sondern ganz einfach Problemlösungen in Basic anbieten. Dabei kann es sich beispielsweise um ein Textverarbeitungsprogramm, eine Lagerverwaltung oder ganz einfach um ein Programm zum Ausdrucken eines Jahreskalenders handeln. Für fast alle derartigen Programme sollte es möglich sein, eine Anpassung mit Hilfe der hier abgedruckten Tabellen und Adressenvergleiche vorzunehmen. (ev)

Bild 2. Speicherbelegung des VC 20 bis 3-KByte-Erweiterung (links) und ab 8-KByte-Erweiterung (rechts)

HEX		DEZIMAL		HEX
FFFF		65535		FFFF
	8 K Betriebssystem-ROM			
E000		57344		E0000
	8 K Basic-ROM			
C000		49152		C000
	8 K Steckmodulbereich			
A000	frei für I/O-Erweiterungen	40960		A000
9800	0,5 K Farb-RAM	38912		9800
9600	frei	38400		9600
9400	1 K Systemkontrollen	37888		9400
9000	4 K Zeichengenerator-ROM	36864		9000
8000		32768		8000
			8 K Erweiterung 3 (+24 K)	
		24576		6000
	frei		8 K Erweiterung 2 (+ 16 K)	
		16384		4000
			8 K Erweiterung 1 (+8 K)	
2000	0,5 K Video-RAM	8192		2000
1E00	3,5 K RAM (Grundversion)	7680		
		4608	3,5 K RAM (Grundversion)	
1000	Erweiterung +3 K RAM	4096	0,5 K Video-RAM	1200
			frei oder 3 K für Maschinensprache	1000
0400	1 K RAM Zeropage, Stack	1024		0400
0000		0	1 K RAM Zeropage, Stack	0000