Programmierbarer Grafikprozessor

EF 9366

Denjenigen, die den NDR-Computer nachgebaut haben, wird dieses IC bekannt vorkommen, es steckt auf der GDP64K (GDP = Graphic Display Prozessor). Schließlich will man ja wissen, was der Computer aus dem reingehackten Programm gemacht hat. Und deshalb ist dieser Baustein nicht nur für NDR-Computerfans interessant. Er erzeugt alle für das Aufbauen eines Bildes notwendigen Signale. Die meisten Monitore arbeiten im Prinzip genauso wie ein ganz normaler Fernseher. Aus den 625 Zeilen entstehen zwei Halbbilder; zuerst wird ein Bild mit Hilfe der ungeraden Zeilen aufgebaut, dann mit den geraden. Pro Halbbild stehen uns 20 ms zur Verfügung, d.h., es entsteht mit einer Frequenz von 50 Hz, das komplette Bild also mit der halben Frequenz. Doch das reicht aus, um ein einigermaßen flimmerfreies Bild zu erhalten. Noch besser ist allerdings der Trick, der im NDR-Computer verwendet wurde: für den Aufbau des Bildes werden dort nur die ungeraden Zeilen verwendet, und da steht das Bild wirklich bombenfest. Allzuviele Signale benötigt der Monitor nicht einmal, um vor unseren erstaunten Augen ein Bild entstehen zu lassen. Es müssen das Horizontal-Synchron-Signal, das Vertikal-Synchron-Signal (Bildwechsel) und die Bildinformation an sich (Video-Signal) vorhanden-sein. Diese drei Signale werden zusammengemischt zu dem sogenannten BAS-Signal, können somit über eine Leitung in den Fernseher gelangen und werden dort wieder getrennt. Die komplette Bildinformation steht im Bildwiederholspeicher, bei der GDP64K also in dem 64-K-Arbeitsspeicher. Ein gesetztes Bit bedeutet einen dunklen Bildpunkt.

Nun wäre es für den Anwender ausgesprochen mühsam, das von ihm gewünschte Bild Bit für Bit einzugeben. Der Grafik-Prozessor muß also schon eine ganze Reihe fertiger Bilder enthalten, die durch Ansprechen bestimmter Speicherbereiche des Prozessors hervorgeholt werden. Die schematische Übersicht in **Bild 1** zeigt die Register des EF 9366. Die dazugehörenden Adressen stehen in **Bild 2**. Programmiert wird der Baustein natürlich in Assembler, je nachdem, wie Sie den Baustein verwenden wollen, können Sie die Anfangsadresse verändern; im NDR-Computer beispielsweise steht das erste Register auf der Adresse 70_H. Die wichtigen Funktionen stehen im Kommandoregister **Bild 3**. Über das Kommandoregister werden wiederum die beiden Kontrollregister angesprochen, die für das Schreiben und Interrupts (CTRL 1) sowie die Lage der Buchstaben und die Strichart der Vektoren (CTRL 2) zuständig sind. Ab 20_H (damit ist der Inhalt = Befehl, nicht die Adresse gemeint) beginnt der wichtige ASCII-Zeichensatz, es ist der amerikanische Standardzeichensatz, deshalb sind auch keine Umlaute oder das „ß" verfügbar. Ansonsten ist aber die Arbeit mit dem EF 9366 recht vielseitig. Zum Beispiel können die

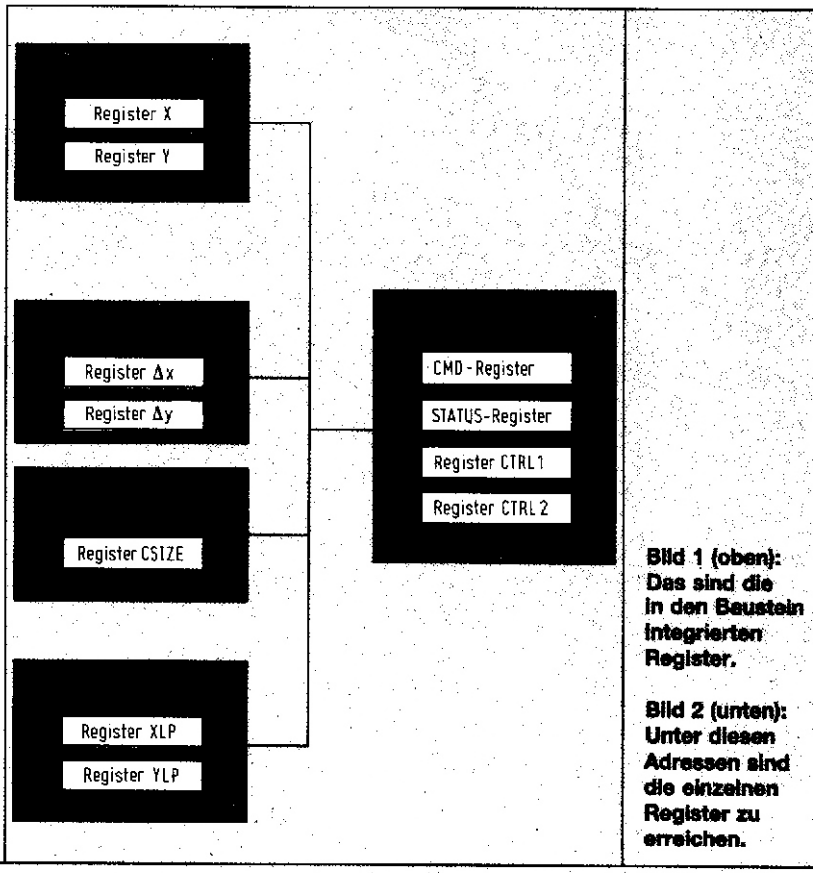


Bild 1 (oben): Das sind die in den Baustein integrierten Register.

Bild 2 (unten): Unter diesen Adressen sind die einzelnen Register zu erreichen.

Registeradresse					Registerfunktionen	
binär				hexa-dezi-mal	Lesen R/$\overline{W}$ = 1	Schreiben R/$\overline{W}$ = 0
A3	A2	A1	A0			
0	0	0	0	0	STATUS	CMD-Befehlsregister
0	0	0	1	1	CTRL1 , 1.Kontrollregister, Schreibvorgänge und Interruptsteuerung	
0	0	1	0	2	CTRL 2 , 2.Kontrollregister, Zeichenstrichart der Vektoren, Orientierung der Buchstaben	
0	0	1	1	3	CSIZE , Register für Zeichengröße in X- und Y-Achse	
0	1	0	0	4	nicht benutzt	
0	1	0	1	5	DELTAX , Delta-X-Register (siehe Vektorbefehle)	
0	1	1	0	6	nicht benutzt	
0	1	1	1	7	DELTAY, Delta-Y-Register (siehe Vektorbefehle)	
1	0	0	0	8	X-Register, Bit 0...3 (MSB), X-Koordinaten (256...4095)	
1	0	0	1	9	X-Register, Bit 0...7 (LSB), X-Koordinaten (0...255)	
1	0	1	0	A	Y-Register, Bit 0...3 (MSB), Y-Koordinaten (256...4095)	
1	0	1	1	B	Y-Register, Bit 0...7 (LSB), Y-Koordinaten (0...255)	
1	1	0	0	C	X-Lichtgriffelregister, Bit 2...7, Bit 1 immer 0, Bit 0 Status	nicht benutzt
1	1	0	1	D	Y-Lichtgriffelregister, Bit 0...7	nicht benutzt
1	1	1	0	E	nicht benutzt	
1	1	1	1	F	nicht benutzt	

b7 b6 b5 b4 / b3 b2 b1 b0			0 0 0 0	0 0 0 1	0 0 1 0	0 0 1 1	0 1 0 0	0 1 0 1	0 1 1 0	0 1 1 1	1 0 0 0	1 0 0 1	1 0 1 0	1 0 1 1	1 1 0 0	1 1 0 1	1 1 1 0	1 1 1 1
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0 0 0 0	0	Schreibfunktion, im 1. Kontrollregister wird Bit Nr. 1 gesetzt		Space	0	@	P	`	p									
0 0 0 1	1	Löschfunktion, im 1. Kontrollregister wird Bit Nr. 1 gelöscht		!	1	A	Q	a	q									
0 0 1 0	2	schreiben, im 1. Kontrollregister wird Bit Nr. 0 gesetzt		"	2	B	R	b	r									
0 0 1 1	3	unsichtbar schreiben, im 1. Kontrollregister wird Bit Nr. 0 gelöscht		#	3	C	S	c	s									
0 1 0 0	4	Bildschirm löschen		$	4	D	T	d	t									
0 1 0 1	5	X- und Y-Register auf 0 setzen		%	5	E	U	e	u									
0 1 1 0	6	Bildschirm und X- sowie Y-Register löschen		&	6	F	V	f	v									
0 1 1 1	7	Bildschirm und alle Register außer Lichtgriffel löschen, Schriftgröße am kleinsten		'	7	G	W	g	w									
1 0 0 0	8	Bildschirm kurz hell für Lichtgriffelposition		(	8	H	X	h	x									
1 0 0 1	9	Lichtgriffelfunktion einschalten		)	9	I	Y	i	y									
1 0 1 0	A	Rechteck ausgeben, Größe P × Q vom Zeichengrößen-Register abhängig		*	:	J	Z	j	z									
1 0 1 1	B	Quadrat ausgeben, Größe P × Q vom Zeichengrößen-Register abhängig		+	;	K	[	k	{									
1 1 0 0	C	Bildschirm hell, Zeichen dunkel		,	<	L	\	l	¦									
1 1 0 1	D	X-Register löschen		-	=	M	]	m	}									
1 1 1 0	E	Y-Register löschen		.	>	N	↑	n	~									
1 1 1 1	F	Direktzugriff auf Bildschirmspeicher (nur bedingt verwendbar)		/	?	O	←	o	▒									

Kurzvektorerzeugung, siehe Kurzvektordefinition

Vektorrichtung, siehe Kurzvektordefinition

Kurzvektordefinition:

b7	b6 b5	b4 b3	b2 b1 b0
1	|ΔX|	|ΔY|	Richtung

Dimension

ΔX	ΔY	Vektorlänge
0	0	0 Punkte
0	1	1 Punkt
1	0	2 Punkte
1	1	3 Punkte

Richtung:

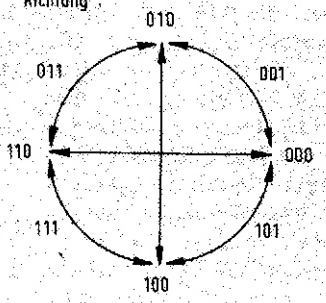
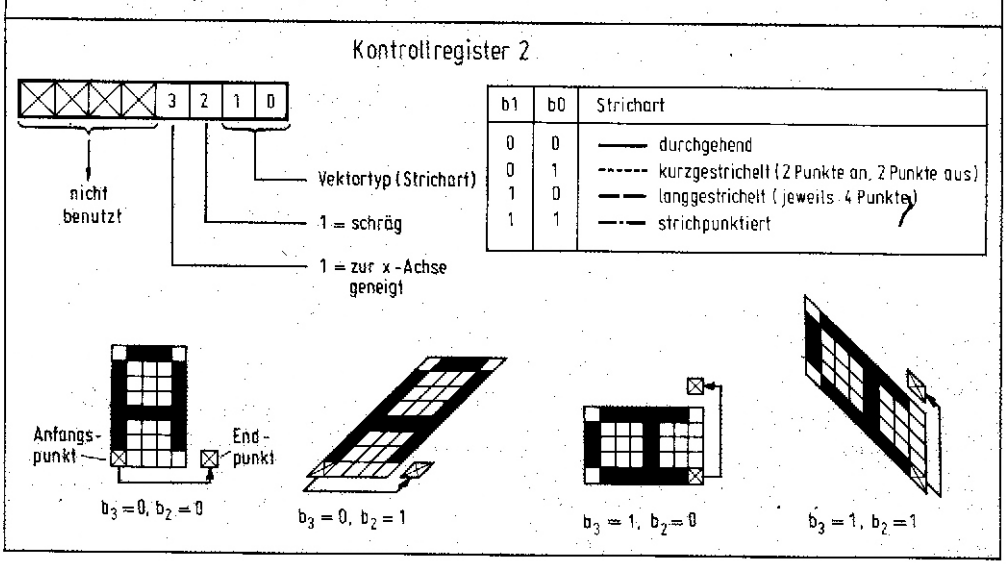


Bild 3 (oben):
Im Kommandoregister sind der ASCII-Zeichensatz, die Vektorbefehle und einige andere den Bildschirm betreffende Befehle untergebracht.

Bild 4 (unten):
So lassen sich die Buchstaben kippen.



Zeichen des ASCII-Satzes schräg geschrieben werden oder in einer anderen Größe. Das Schrägschreiben wird vom zweiten Kontrollregister gesteuert (s. **Bild 4**), die Größe wird im C-Size-Register, unter der Adresse 3, festgelegt. Wo der Buchstabe dann erscheint, stehen im X/Y-Register, das die Adressen 08_H bis $0B_H$ belegt, also pro Koordinate zwei Bytes, da mit 16-Bit-Worten gearbeitet wird; max. 4095.

Die bisher erläuterten Möglichkeiten des EF 9366 sind zwar ganz nett, aber unterscheiden ihn kaum von anderen Typen. Doch der Grafikprozessor beinhaltet nicht nur den ASCII-Zeichensatz, er kann auch Kurzvektoren erzeugen, die in Strichart und -länge sehr einfach zu programmieren sind. Im Kontrollregister 2 sind die beiden untersten Bits für die Strichart zuständig. In das Kommandoregister (s. Bild 2) sind die Länge (1...3 Punkte) und eine der acht Richtungen einzugeben. Durch Hintereinanderfügen von Kurzvektoren lassen sich Striche in beliebiger Länge und Strichart zeichnen. Dafür muß ein kleines Assemblerprogramm geschrieben werden, das, ausgehend vom Anfangspunkt, die jeweils neuen Koordinaten für den nächsten Vektor errechnet. Da die Kurzvektoren auch aus einem Punkt bestehen können, lassen sich auch Kreise oder kompliziertere Kurven zeichnen. Da muß man allerdings schon ziemlich fit in der Programmierung sein.

Nach Unterlagen von Thomson