
ASCII- TOETSENBORD ENCODER, UART- SCHAKELING EN BAUDRATE- GENERATOR

Om opdrachten en gegevens in een computer te brengen is een toetsenbord zeer geschikt. Het indrukken van een toets moet voor de computer echter worden vertaald in een binaire code. De hiervoor meest gebruikte is de ASCII-code, waarvoor een encoder wordt beschreven. De ingetypte karakters worden tevens naar een printer of videodisplay gestuurd, zodat de programmeur de invoer kan volgen. Voor het serieel oversturen van deze ASCII-codes dient de UART-schakeling, die ook universeel toepasbaar is. Deze kan worden gestuurd door een nauwkeurige baudrate-generator om een foutloze en compatibele overdracht te verzekeren.

H.J.C. OTTEN

ASCII- toetsenbord- encoder

Inleiding

Een ASCII-toetsenbord is onmisbaar om informatie, bestaande uit gegevens en stuurcommando's, aan een computer toe te voeren. Deze ASCII-toetsenborden zijn geheel compleet te koop, inclusief de encoderschake-

ling, maar eveneens worden er losse toetsenborden aangeboden. Men kan ook zelf een toetsenbord opbouwen uit losse toetsenbordschakelaars.

Om van het indrukken van een toets tot de 7 bits ASCII-code te komen is een encoderschakeling nodig. Er zijn een aantal mogelijkheden:

1. De computer kan zelf het aftasten van het toetsenbord verzorgen met een PIA (Peripheral Interface Adapter) op een wijze die sterk lijkt op de hier beschreven schakeling. Deze mogelijkheid vervalt als het toetsenbord deel moet uitmaken van

een teletype-achtige in- en uitvoereenheid, die op de seriële teletype in- en uitgang van de computer wordt aangesloten.

2. Er bestaan IC's die het opwekken van de ASCII-code geheel verzorgen. Deze IC's hebben als nadeel dat ze niet zo makkelijk te verkrijgen en vrij prijzig zijn. Verder vereisen ze een extra negatieve voedingsspanning van - 12 V.
3. Een uit TTL-IC's opgebouwde encoder, zoals hierna wordt beschreven. De onderdelen hiervoor zijn goedkoop en makkelijk te verkrijgen. De flexibele opzet maakt uitbreidingen en wijzigingen eenvoudig. Zo is de encoder ook geschikt om andere codes, zoals de Baudot-code op te wekken.

ASCII-code

De ASCII-code (American Standard For Coded Information Interchange) is een veel gebruikte digitale code in computers. Ook de KIM-1 gebruikt deze code als er bijvoorbeeld informatie over de teletype in- en uitgang wordt verstuurd.

De ASCII-code bestaat uit 7 bits, waardoor er 128 combinaties van enen en nullen zijn mogelijk. In tabel 1 is te zien hoe de verschillende karakters in de ASCII-code worden voorgesteld. Allereerst is de eigenlijke ASCII-code, zeven binaire getallen, te zien: van links naar rechts het meest significante bit, b_6 , tot het laagste bit, b_0 . Dit binaire getal is in de kolom ernaast omgezet in het hexadecimale getal en daarnaast in het decimale getal. De volgende kolom vermeldt of het een normaal toetskarakter is of dat het met een speciale manipulatie, waarover later meer, wordt opgewekt. De naam van het ASCII-karakter is vervolgens vermeld met daarachter eventuele opmerkingen.

De 128 mogelijke combinaties zijn ruim voldoende om behalve de letters van het alfabet, de getallen 0 tot en met 9 en vele leestekens en speciale karakters, ook een groot aantal stuurtekens mogelijk te maken. Velen daarvan zijn bedoeld om een teletype, een elektrische schrijfmachine die ook door de computer kan worden gestuurd, te bedienen, zoals carriage return (wagen terug), line feed (volgende regel) en space (spatie). Aan de stuurtekens wordt in ieder computerprogramma weer een andere betekenis gehecht, zodat een ASCII-toetsenbord ze allemaal moet kunnen opwekken.

ASCII-toetsenbord

Om ASCII-karakters in de computer te voeren wordt een toetsenbord gebruikt. Indrukken van een toets heeft als gevolg dat op de zeven uitgangen

de bij de toets behorende ASCII-code verschijnt, terwijl op een achtste uitgang een strobe-sigitaal, een digitale puls, verschijnt om aan te geven dat er een toets is ingedrukt.

Om de gehele ASCII-code op deze wijze op te wekken zouden er 128 toetsen nodig zijn. Omdat dit erg onpraktisch is, worden de meest gebruikte karakters als toets uitgevoerd, zoals letters, cijfers en leestekens en wordt aan de meeste toetsen een dubbele en soms drievoudige functie gegeven. Bij een gewone schrijfmachine hebben vele toetsen ook een dubbele functie. Het ASCII-toetsenbord lijkt erg op het toetsenbord van een schrijfmachine, maar er zijn toch wel een paar verschillen. De indeling is ongeveer hetzelfde, letters in het midden, boven een rij cijfertoesen en onder een spatiebalk. Toegevoegd zijn een aantal toetsen voor besturing, waaronder altijd carriage return en line feed. Gewoonlijk zijn de kleine letters niet te krijgen met de gewone lettertoetsen, maar zijn de hoofdletters de normale functie. De kleine letters zijn de shift-functie, dat wil zeggen dat

een kleine letter wordt verkregen door de desbetreffende lettertoets in te drukken en tegelijk de shift-toets ingedrukt te houden. De niet als aparte toets uitgevoerde stuurttoetsen, de control-karakters, zijn te verkrijgen als control-functie van de lettertoetsen, dat wil zeggen dat een control-karakter wordt verkregen door een lettertoets in te drukken terwijl de controltoets ingedrukt wordt gehouden. In tabel 1 is te zien in de kolom „soort” of het karakter een normale toets is (aangegeven door „N”), de shift-functie (aangegeven door SH/, gevolgd door de naam van de toets, of de control-functie (aangegeven door CTL/, gevolgd door de naam van de toets). Sommige control-toetsen zijn als losse toetsen uitgevoerd, maar ook als control-functie beschikbaar. Zo wordt het spatiekarakter (\$20) verkregen met de spatiebalk, maar ook als control-functie van het „@”teken.

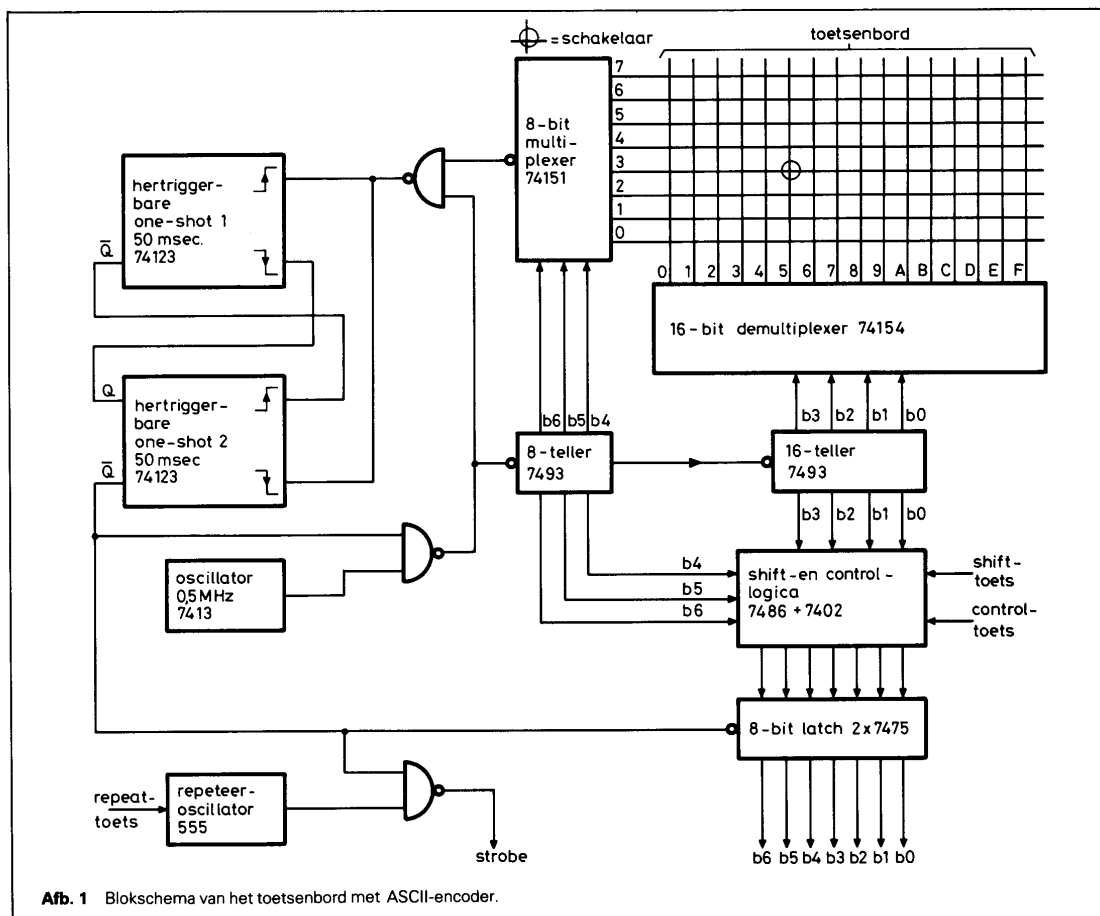
Encoder

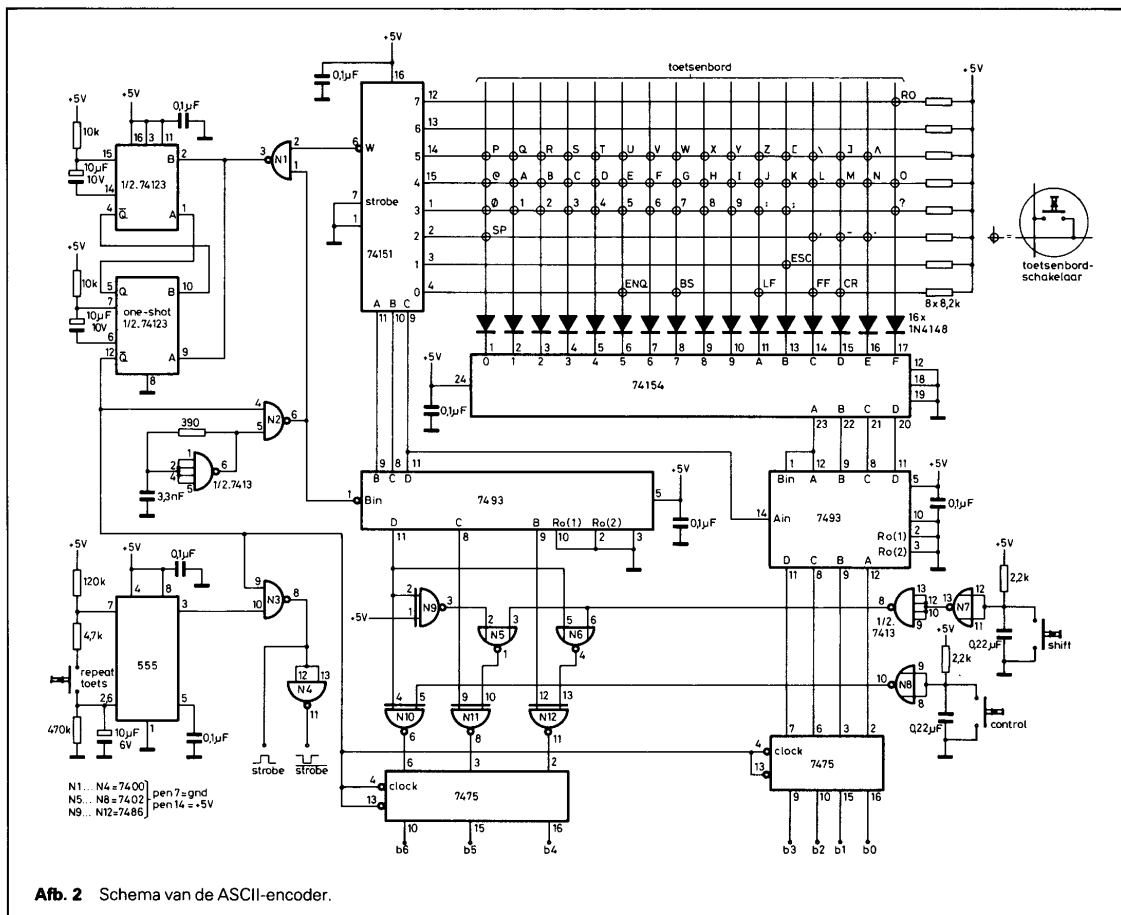
Uit het voorgaande is duidelijk geworden dat het indrukken van een toets op het toetsenbord moet worden om-

gezet in de zeven bits ASCII-code. Er moet een strobe-sigitaal worden opgewekt om aan te geven dat er een karakter beschikbaar is en de shift- en control-functies moeten worden uitgevoerd. Al deze functies worden door de encoderschakeling gerealiseerd.

Het blokschema is in afb. 1 te zien, het prinsipschema in afb. 2. Een klokoscillator rond een 7413 stuurt een 8-teller (7493). De uitgangen van deze 8-teller worden naar een 8 bits multiplexer gevoerd. Door het lopen van de teller worden de ingangen van de multiplexer, die door trekweerstand normaal hoog zijn, om beurten met de uitgang van de multiplexer verbonden. De uitgang van de multiplexer zal dus normaal hoog zijn.

De overloop van de 8-teller stuurt een 16-teller. De uitgangen van deze 16-teller worden naar een 16 bits demultiplexer gevoerd. De uitgangen van deze demultiplexer worden achtereenvolgens door het lopen van de 16-teller even laag. We hebben nu 16 stuurlijnen die achtereenvolgens even laag worden en 8 aftastlijnen die ach-





tereenvolgens worden bekeken. Door de trekweerstand is de uitgang van de multiplexer hoog. Als nu een verbinding tussen een stuurlijn en een aftastlijn wordt gemaakt, en dit gebeurt als een toets wordt ingedrukt, wordt de uitgang van de multiplexer laag op het moment dat zowel de stuurlijn als de aftastlijn aan de beurt is. Het laag worden van de uitgang gebeurt dus bij een bepaalde stand van de beide tellers. Het uitgangssignaal van de multiplexer heeft via een poortschakeling tot gevolg dat er geen telpulsen meer aan de tellers worden toegevoerd. De tellers stoppen en de stand van de tellers identificeert on dubbelzinnig welke toets is ingedrukt. De tellerstand, 7 bits, wordt naar een latch (2×7475) gevoerd en daarin opgeslagen.

Het indrukken van twee toetsen tegelijkertijd sluit twee uitgangen kort als ze bij dezelfde aftastlijn behoren. De uitgangen zijn hiertegen beschermd door middel van dioden. Welke code wordt opgewekt bij het gelijktijdig indrukken van twee toetsen is volledig willekeurig, het zogenaamde „roll-over” principe.

Antidenderschakeling

Na het indrukken van een toets wordt de uitgang van de multiplexer laag en stopt de teller. Het uitgangssignaal van de multiplexer, een 1 naar 0 overgang, wordt met het kloksignaal in een NAND-poort gebracht. Daar de tellers reageren op een 1 naar 0 overgang, zal bij een coincidentie, die optreedt bij het indrukken van een toets, het kloksignaal laag zijn. Pas als het kloksignaal weer hoog wordt, zal het uitgangssignaal van de multiplexer worden doorgegeven aan de antidenderschakeling. Door het uitpoorten met het kloksignaal zijn schijncoincidenties tengevolge van contactdenier en het gebruik van asynchrone tellers uitgeschakeld. Het coincidentiesignaal wordt vervolgens aan twee „one-shots”, twee monostabiele multivibratoren die hertriggeerbaar zijn (74123), toegevoerd. De ene one-shot levert het strobe-signaal dat de tellers blokkeert en ook naar buiten is gevoerd. De andere one-shot blokkeert door het hertriggere van de strobe one-shot tot de toets minimaal 50 msec is losgelaten. Contactdender is zo effectief ondervangen zonder de typesnel-

heid te beperken. Deze antidiender-
schakeling is door R. E. Martens in
Radio Bulletin beschreven, in het ju-
linummer van 1976.

Het strobe-signaal wordt via N3 naar buiten gevoerd en is een puls van 50 msec. Tijdens de strobe-puls is de ASCII-code beschikbaar op de uitgangen van de latch en door het inklokken op het strobe-signaal blijft de code ook na de strobe-puls beschikbaar tot een nieuwe toets wordt ingedrukt. Via N4 is eveneens een geïnvverteerde strobe-puls beschikbaar.

Repeat-toets

Na de strobe-puls blijft de code op de uitgangen van de latch beschikbaar. Als nu hetzelfde karakter weer moet worden opgewekt, wat bijvoorbeeld met spaties vaak zal voorkomen, kan de toets weer worden ingedrukt. Het is veel handiger en sneller om een repeat-toets te gebruiken. Het indrukken en ingedrukt houden van deze toets wekt namelijk een reeks strobe-pulsen op, terwijl de code op de uitgangen van de latch blijft staan. Het indrukken van de repeat-toets zal dus een oscillator moeten starten die

stopt als de repeat-toets weer wordt losgelaten. Een dergelijke oscillator is rond een timer IC 555 opgebouwd. De schakeling is die van een normale oscillator rond een 555 als de repeat-toets is ingedrukt. Wordt deze toets losgelaten, dan wordt de condensator van 10 μ F niet meer opgeladen en blijft de uitgang hoog. Door de keuze van weerstanden en condensator is de zo ontstane puls, die in N3 wordt gemengd met de gewone strobe-pulsen, ongeveer even lang als de strobe-pulsen, zodat het voor de computer lijkt of iedere keer dezelfde toets wordt ingedrukt.

Er bestaat een kleine kans dat de repeat-toets wordt losgelaten als de condensator nog is opgeladen. De uitgang is dan laag en zal dat blijven, waardoor de gewone strobepulsen worden geblokkeerd. Om dit te voorkomen is over de condensator een weerstand van 470 k Ω geplaatst die de condensator onlaadt, zodat de strobe-pulsen niet blijvend worden geblokkeerd.

Controle-toets

Uit tabel 1 blijkt dat er een groot aantal stuurcodes zijn, de control-codes, waarvan er maar een paar als toets worden uitgevoerd. De andere control-codes kunnen worden opgewekt met de control-toets. Als we in tabel 1 de codes van de letters (hex 40 – 5F) vergelijken met die van de control-codes, dan zien we dat deze gelijk zijn op bit b_6 na. Als we b_6 nu van een 1 in een nul veranderen en de lettertoets indrukken krijgen we de control-code. b_6 wordt, alvorens naar de latch te worden gevoerd, naar een „exclusive or“-poort gevoerd. De andere ingang van de poort is aangesloten op de control-toets. Als de control-toets niet is ingedrukt staat er een 0 op de ingang. De EXOR-poort laat het signaal op de andere ingang dan onveranderd door naar de uitgang en naar de latch. Is de control-toets ingedrukt, dan staat er een 1 op de ingang en de EXOR-poort invertteert b_6 . Was b_6 een 1, wat bij de letters het geval is, dan wordt er een 0 in de latch gezet, zodat de control-code in de latch komt. Tabel 2 verduidelijkt dit nog.

Shift-toets

Veel toetsen hebben door de control-toets een dubbele functie. Door de shift-toets krijgen sommige toetsen een drievoudige functie. Zo wordt bijvoorbeeld een lettertoets gebruikt om de code voor een hoofdletter te krijgen, met de control-toets wordt een control-code verkregen, de shift-toets met de lettertoets levert de code voor een kleine letter. In tabel 1 is te zien welke codes als shift-functie van gewone toetsen

Tabel 1 ASCII-codes.

Binair b_6 b_5 b_4 b_3 b_2 b_1 b_0	Hex	Dec	Soort	Naam	Opmerkingen
0 0 0 0 0 0 0	00	000	CTL/@	NUL	Null
0 0 0 0 0 0 1	01	001	CTL/A	SOH	Start of heading
0 0 0 0 0 1 0	02	002	CTL/B	STX	Start of text
0 0 0 0 0 1 1	03	003	CTL/C	ETX	End of text, Break
0 0 0 0 1 0 0	04	004	CTL/D	EOT	End of transmission
0 0 0 0 1 0 1	05	005	CTL/E	ENQ	Enquiry; Here is
0 0 0 0 1 1 0	06	006	CTL/F	ACK	Acknowledge
0 0 0 0 1 1 1	07	007	CTL/G	BEL	Bell
0 0 0 1 0 0 0	08	008	CTL/H	BS	Backspace
0 0 0 1 0 0 1	09	009	CTL/I	HT	Horizontal tabulation
0 0 0 1 0 1 0	0A	010	N	LF	Line feed
0 0 0 1 0 1 1	0B	011	CTL/K	VT	Vertical tabulation
0 0 0 1 1 0 0	0C	012	CTL/L	FF	Form feed
0 0 0 1 1 0 1	0D	013	N	CR	Carriage return
0 0 0 1 1 1 0	0E	014	CTL/N	SO	Shift out
0 0 0 1 1 1 1	0F	015	CTL/O	SI	Shift in
0 0 1 0 0 0 0	10	016	CTL/P	DLE	Data link escape
0 0 1 0 0 0 1	11	017	CTL/Q	DC1	Device control 1
0 0 1 0 0 1 0	12	018	CTL/R	DC2	Device control 2
0 0 1 0 0 1 1	13	019	CTL/S	DC3	Device control 3
0 0 1 0 1 0 0	14	020	CTL/T	DC4	Device control 4
0 0 1 0 1 0 1	15	021	CTL/U	NAK	Negative acknowledge
0 0 1 0 1 1 0	16	022	CTL/V	SYN	Synchronous idle
0 0 1 0 1 1 1	17	023	CTL/W	ETB	End of transmission block
0 0 1 1 0 0 0	18	024	CTL/X	CAN	Cancel
0 0 1 1 0 0 1	19	025	CTL/Y	EM	End of medium
0 0 1 1 0 1 0	1A	026	CTL/Z	SUB	Substitute
0 0 1 1 0 1 1	1B	027	CTL/[	ESC	Escape
0 0 1 1 1 0 0	1C	028	CTL/\	FS	File separator
0 0 1 1 1 0 1	1D	029	CTL/]	GS	Group separator
0 0 1 1 1 1 0	1E	030	CTL/^	RS	Record separator
0 0 1 1 1 1 1	1F	031	CTL/_	US	Unit separator
0 1 0 0 0 0 0	20	032	N	SP	Space
0 1 0 0 0 0 1	21	033	SH/1	!	
0 1 0 0 0 1 0	22	034	SH/2	"	
0 1 0 0 0 1 1	23	035	SH/3	#	
0 1 0 0 1 0 0	24	036	SH/4	\$	
0 1 0 0 1 0 1	25	037	SH/5	%	
0 1 0 0 1 1 0	26	038	SH/6	&	
0 1 0 0 1 1 1	27	039	SH/7	'	
0 1 0 1 0 0 0	28	040	SH/8	(	
0 1 0 1 0 0 1	29	041	SH/9	)	
0 1 0 1 0 1 0	2A	042	SH/:	*	
0 1 0 1 0 1 1	2B	043	SH/;	+	
0 1 0 1 1 0 0	2C	044	N	,	
0 1 0 1 1 0 1	2D	045	N	-	
0 1 0 1 1 1 0	2E	046	N	.	
0 1 0 1 1 1 1	2F	047	SH/?	/	Ook wel normale toets
0 1 1 0 0 0 0	30	048	N	$\emptyset$	
0 1 1 0 0 0 1	31	049	N	1	
0 1 1 0 0 1 0	32	050	N	2	
0 1 1 0 0 1 1	33	051	N	3	
0 1 1 0 1 0 0	34	052	N	4	
0 1 1 0 1 0 1	35	053	N	5	
0 1 1 0 1 1 0	36	054	N	6	
0 1 1 0 1 1 1	37	055	N	7	
0 1 1 1 0 0 0	38	056	N	8	
0 1 1 1 0 0 1	39	057	N	9	
0 1 1 1 0 1 0	3A	058	N	:	
0 1 1 1 0 1 1	3B	059	N	;	
0 1 1 1 1 0 0	3C	060	SH/,	<	
0 1 1 1 1 0 1	3D	061	SH/-	=	

Tabel 2 Werking van de control-toets.

b_6	control-toets	b_6 in latch
0	0	0
1	0	1
0	1	1
1	1	0

b_6 doorlaten

control-functie

Vervolg Tabel 1

0	1	1	1	1	1	0	3E	062	SH/.	>	Ook wel SH/ /
0	1	1	1	1	1	1	3F	063	N	?	
1	0	0	0	0	0	0	40	064	N	@	
1	0	0	0	0	0	1	41	065	N	A	
1	0	0	0	0	1	0	42	066	N	B	
1	0	0	0	0	1	1	43	067	N	C	
1	0	0	0	1	0	0	44	068	N	D	
1	0	0	0	1	0	1	45	069	N	E	
1	0	0	0	1	1	0	46	070	N	F	
1	0	0	0	1	1	1	47	071	N	G	
1	0	0	1	0	0	0	48	072	N	H	
1	0	0	1	0	0	1	49	073	N	I	
1	0	0	1	0	1	0	4A	074	N	J	
1	0	0	1	0	1	1	4B	075	N	K	
1	0	0	1	1	0	0	4C	076	N	L	
1	0	0	1	1	0	1	4D	077	N	M	
1	0	0	1	1	1	0	4E	078	N	N	
1	0	0	1	1	1	1	4F	079	N	O	
1	0	1	0	0	0	0	50	080	N	P	
1	0	1	0	0	0	1	51	081	N	Q	
1	0	1	0	0	1	0	52	082	N	R	
1	0	1	0	0	1	1	53	083	N	S	
1	0	1	0	1	0	0	54	084	N	T	
1	0	1	0	1	0	1	55	085	N	U	
1	0	1	0	1	1	0	56	086	N	V	
1	0	1	0	1	1	1	57	087	N	W	
1	0	1	1	0	0	0	58	088	N	X	
1	0	1	1	0	0	1	59	089	N	Y	
1	0	1	1	0	1	0	5A	090	N	Z	
1	0	1	1	0	1	1	5B	091	N	[	
1	0	1	1	1	0	0	5C	092	N	\	
1	0	1	1	1	0	1	5D	093	N	]	
1	0	1	1	1	1	0	5E	094	N	^	
1	0	1	1	1	1	1	5F	095	N	—	Ook wel ↑ Ook wel ← of Delete Niet gedefinieerd
1	1	0	0	0	0	0	60	096	SH/@		
1	1	0	0	0	0	1	61	097	SH/A	a	
1	1	0	0	0	1	0	62	098	SH/B	b	
1	1	0	0	0	1	1	63	099	SH/C	c	
1	1	0	0	1	0	0	64	100	SH/D	d	
1	1	0	0	1	1	1	65	101	SH/E	e	
1	1	0	0	1	1	0	66	102	SH/F	f	
1	1	0	0	1	1	1	67	103	SH/G	g	
1	1	0	1	0	0	0	68	104	SH/H	h	
1	1	0	1	0	0	1	69	105	SH/I	i	
1	1	0	1	0	1	0	6A	106	SH/J	j	
1	1	0	1	0	1	1	6B	107	SH/K	k	
1	1	0	1	1	0	0	6C	108	SH/L	l	
1	1	0	1	1	0	1	6D	109	SH/M	m	
1	1	0	1	1	1	0	6E	110	SH/N	n	
1	1	0	1	1	1	1	6F	111	SH/O	o	
1	1	1	0	0	0	0	70	112	SH/P	p	
1	1	1	0	0	0	1	71	113	SH/Q	q	
1	1	1	0	0	1	0	72	114	SH/R	r	
1	1	1	0	0	1	1	73	115	SH/S	s	
1	1	1	0	1	0	0	74	116	SH/T	t	
1	1	1	0	1	0	1	75	117	SH/U	u	
1	1	1	0	1	1	0	76	118	SH/V	v	
1	1	1	0	1	1	1	77	119	SH/W	w	
1	1	1	1	0	0	0	78	120	SH/X	x	
1	1	1	1	0	0	1	79	121	SH/Y	y	
1	1	1	1	0	1	0	7A	122	SH/Z	z	
1	1	1	1	0	1	1	7B	123	SH/[	{	
1	1	1	1	1	0	0	7C	124	SH/\		
1	1	1	1	1	0	1	7D	125	SH/]	}	
1	1	1	1	1	1	0	7E	126	SH/^		
1	1	1	1	1	1	1	7F	127	N	DEL Delete of Rub out	

Tabel 3 Werking van de shift-toets.

b ₆	shift-toets	b ₄ in latch	b ₅ in latch
0	0	b ₄	b ₅
1	0	b ₄	b ₅
0	1	$\overline{b_4}$	b ₅
1	1	b ₄	$\overline{b_5}$

worden verkregen. In de kolom „soort” is te zien bij welke toets de shift-code behoort. Vergelijken van de toets-code en de shift-code geeft als resultaat:

1. De codes 2X moeten worden omgezet in de codes 3X, de codes 3X moeten worden omgezet in de codes 2X. Dit is mogelijk door b₄ te inverteren ten gevolge van het indrukken van de shift-toets. Hierbij is b₆ voor alle codes 0.
2. De codes 4X en 5X moeten worden omgezet naar de codes 6X respectievelijk 7X. Hiertoe moet b₅ worden geïnverteerd. b₆ is in dit geval 1.

Het bovenstaande is samengevat in tabel 3. Zowel b₄ als b₅ moeten kunnen worden geïnverteerd. Twee EXOR-poorten, N₁₁ en N₁₂, maken dit mogelijk op dezelfde wijze als bij de control-toets. Alleen is het al of niet inverteren nu afhankelijk van de shift-toets en de toestand van b₆, zoals in tabel 3 is te zien. Het stuursignaal voor het inverteren wordt met N₅, N₆ en N₉ samengesteld.

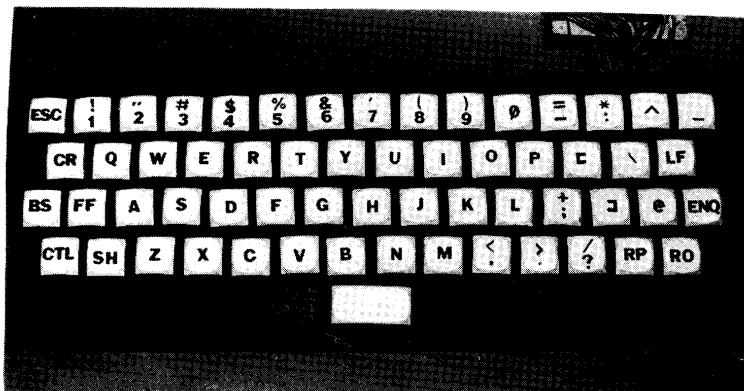
Toetsenbord

Het toetsenbord moet uit „normaal-open”-drukschakelaars zijn opgebouwd. Dit kan een gewone toetsenbordschakelaar zijn, maar ook toetsen met micro-switches of reed-contacten zijn bruikbaar.

Er zijn reeds gemonteerde toetsenborden te koop in vele uitvoeringen en prijzen en daarmee samenhangende kwaliteit. Het kan echter voordeliger en handiger zijn om zelf een toetsenbord samen te stellen uit losse toetsen, waar een ruim aanbod van is, ook weer verschillend in prijs en kwaliteit. Het voordeel van het zelf bouwen van een toetsenbord is de vrijheid die we dan hebben in de opstelling van de toetsen. De meeste toetsen zullen wel op de normale (schrijfmachine) manier worden geplaatst, maar de stuurtoetsen, de control-toetsen, kunnen naar eigen inzicht worden geplaatst. Veel gebruikte toetsen kunnen als aparte toets in plaats van als control-functie worden opgenomen.

Een ander voorbeeld van de vrijheid die we hebben is de „?”-toets. Vaak is dit de shift-functie van de „/”-toets. Omdat in vele (Microsoft) Basic-implementaties het vraagteken als verkorte notatie van „PRINT” wordt gebruikt, is het handiger het vraagteken als normale toets uit te voeren. Het „/” is dan de shift-functie.

Een aantal toetsen kunnen we ook voor hardwarebesturingssignalen reserveren. Zo kan de toets „BREAK” zowel CTL/C, een ASCII-stuurcode, als wel STOP (Non Maskable Interrupt), een hardware stopsignaal, als functie hebben. Meestal zijn toetsen-



Afb. 3 Zelfgebouwd toetsenbord met losse schakelaars op een printplaat.

bordschakelaars bedoeld voor printmontage, zodat het opbouwen van een toetsenbord eenvoudig is door een print te maken. De toetsen komen aan de, eerst met een verfspuitbus bewerkte, componentenzijde en worden aan de koperzijde vastgesoldeerd. Op de print kunnen al een groot aantal verbindingen tussen de toetsen (op dezelfde stuur- of aftastlijn) via printbanen worden gemaakt. De rest moet met losse bedrading, eveneens aan de koperzijde, gebeuren. Zoals uit het schema blijkt heeft elke stuurlijn en elke aftastlijn een (hexadecimaal) nummer. De in tabel 1 opgenomen code komt overeen met deze nummers, eerst het aftastlijnnummer, daarna het stuurlijnnummer.

Afwijkende ASCII-codes

Vaak wordt de zeven bits ASCII-code uitgebreid tot 8 bits. Het aantal mogelijkheden neemt hierdoor toe tot 256. De hier beschreven schakeling kan hiertoe worden uitgebreid door een 16-teller in plaats van de 8-teller te gebruiken en ook een 16 bits multiplexer. Een achtste latch is nog over in een 7475.

Bij videodisplay's dient het achtste bit meestal als stuurteken voor het invertieren van het karakter, dat wil zeggen dat in plaats van een wit teken op een zwarte achtergrond, een zwart teken op een witte ondergrond wordt geschreven. Toevoegen van een „reverse“-toets die de gewenste 0 of 1 in het achtste bit plaatst (ook weer in de achtste latch) maakt ook dit mogelijk. Geheel afwijkende codes, zoals de Baudot-code kunnen ook met deze schakeling worden opgewekt. De shift-en control-logica is dan niet nodig. Het aantal bits, waaruit de code bestaat, bepaalt wat voor soort tellers en (de-)multiplexers moeten worden toegepast. De toetsen worden dan natuurlijk op andere knooppunten van stuur- en aftastlijnen geplaatst, afhankelijk van de definitie van de code.

UNIVERSELE UART/T-SCHAKELING

Inleiding

De meeste single-board computers zijn naast een hexadecimaal toetsenbord en LED-display ook voorzien van een teletype in- en uitgang. Deze teletype-aansluiting is een zogenoemde seriële transmissie-aansluiting, terwijl veel randapparaten, zoals ASCII-toetsenborden en ASCII-printers parallel in- en uit-gangen hebben. De benodigde parallel-serie en serie-parallel omzettingen worden door een UART/T-schakeling uitgevoerd.

Asynchrone seriële overdracht

Bij de teletype-aansluiting wordt gebruik gemaakt van asynchrone seriële overdracht. De gegevens worden in serie achter elkaar over één lijn verzonden. Als we een blok digitale gegevens over één lijn gaan zenden, moeten er tussen zender en ontvanger duidelijke afspraken bestaan over de manier, waarop dit zal gebeuren. Zo moeten begin en einde van de uitzending bijvoorbeeld duidelijk zijn te onderscheiden. Bij seriële overdracht worden de volgende regels gevolgd:

1. In rust is de uitgang van de zender hoog.

2. Het begin van de uitzending is een overgang van hoog naar laag. De zenderuitgang blijft een bepaalde tijd, de bittijd, laag.
3. Achtereenvolgens worden de over te brengen databits op de zenderuitgang gezet, ieder bit gedurende één bittijd.
4. Om de ontvanger een controle op de ontvangen data te kunnen laten uitvoeren wordt de pariteit van de uitgezonden data bepaald (de binaire som, zonder carry, van de databits). Gedurende één bittijd wordt de pariteit op de zenderuitgang gezet.
5. De uitzending wordt besloten met een aantal (meestal twee) stopbits. Gedurende één bittijd is bij een stopbit de uitgang hoog.

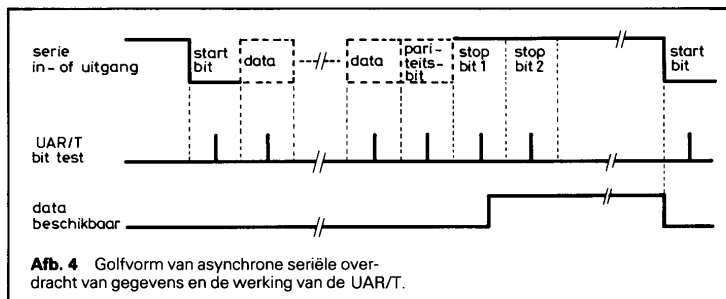
De gebeurtenissen op de uitgang van de zender zijn te zien in afb. 4.

Zender en ontvanger hoeven niet te worden gesynchroniseerd door bijvoorbeeld een gemeenschappelijk kloksignaal, ze moeten alleen dezelfde waarde voor de bittijd aanhouden. Dit wordt bepaald door de baudwaarde van de overdracht.

Baud

Uit het bovenstaande blijkt dat bij asynchrone seriële overdracht de gegevens als blok van een aantal bits worden uitgezonden, vergezeld van een aantal besturingsbits. Het datablok, meestal 7 of 8 bits, stelt samen een ASCII-karakter voor. De snelheid waarmee gegevens met asynchrone seriële overdracht worden getransporteerd kan men daarom uitdrukken in het aantal karakters per seconde dat maximaal kan worden verwerkt. Karakters per seconde wordt in de Engelse literatuur afgekort tot CPS. Men kan de snelheid ook uitdrukken in het aantal databits dat per seconde maximaal kan worden verwerkt. De eenheid hiervoor is bits per seconde: BPS. Een derde snelheidsmaat krijgt men door te kijken naar het totaal aantal bits dat per seconde kan worden verwerkt. Dit noemt men baudsnelheid (Eng: Baudrate), de eenheid heet baud.

Het verband tussen CPS, BPS en baud is afhankelijk van de afspraak



Afb. 4 Golfvorm van asynchrone seriële overdracht van gegevens en de werking van de UART/T.

tussen zender en ontvanger over besturingsbits en aantal databits. Voor een veel voorkomend voorbeeld willen we dit eens uitwerken. Stel dat de data als volgt wordt verzonden. Eerst komt het startbit, daarna volgen zeven databits (één ASCII-karakter), een pariteitsbit en twee stopbits. In totaal worden er dus 11 bits uitgezonden, waarvan zeven databits die één karakter voorstellen. Stel dat de karakter-snelheid 10 CPS is, er worden dus tien karakters per seconde overgebracht. De bitsnelheid is nu 70 BPS, want elk karakter bestaat uit zeven bits en er zijn totaal 10 karakters, zodat het aantal databits 70 per seconde bedraagt.

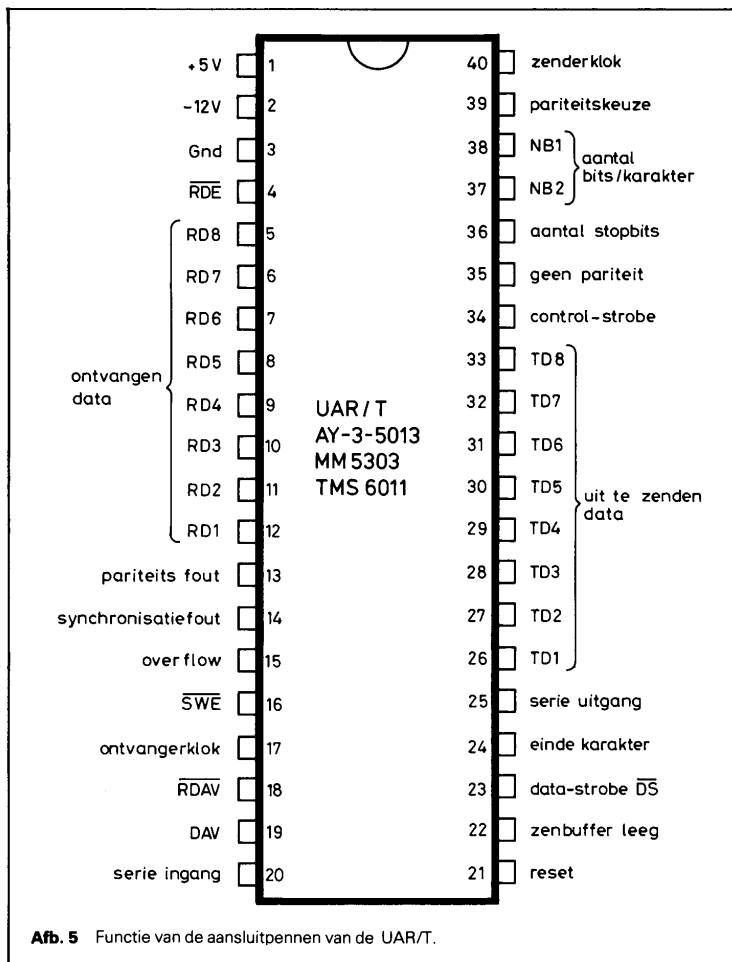
De baudsnelheid is nu 110 baud. Elk karakter bestaat uit zeven databits, één pariteitsbit en drie besturingsbits, in totaal 11. Het totale aantal bits is dus 110 per seconde. In bittijd uitgedrukt is de baudsnelheid altijd gelijk aan één gedeeld door de bittijd.

UAR/T-IC

Een schakeling die de hierboven beschreven functies van zender en ontvanger moet verrichten is in principe op te bouwen uit een serie naar parallel schuifregister, de ontvanger, en een parallel naar serie schuifregister, de zender. Daar moeten nog heel wat besturings- en beslissingsschakelingen aan worden toegevoegd, die zorg dragen voor startbit, stopbits, opwekking van het pariteitsbit en de controle daarvan. Het resultaat zou een zeer complexe schakeling worden. Gelukkig wordt door vele IC-fabrikanten een speciaal voor asynchrone zend- en ontvangdoeleinden bedoeld IC gemaakt. Deze UAR/T-IC's (Universal Asynchronous Receiver-Transmitter) zijn zeer algemeen van opzet. Hier wordt gebruik gemaakt van de UAR/T AY-3-5013 van General Instruments of de equivalenten MM5303 van National en TMS 6011 van Texas Instruments. De UAR/T is te scheiden in twee onafhankelijke blokken, de zender en de ontvanger (er kan tegelijkertijd een karakter worden verzonden en ontvangen met verschillende baudrate). Gemeenschappelijk is alleen de keuze van de parameters die de opbouw van het karakter (aantal databits bijvoorbeeld) bepalen. De aansluitingen van dit 40-pens IC zijn in afb. 5 te zien.

Zender

De timing van de zender wordt afgeleid van de zenderklok. Eén bittijd duurt 16 perioden van de zenderklokfrequentie, aan te sluiten op pen 40. Als de data-strobe, pen 23, laag is, worden de databits, aanwezig op pen 26 tot en met 33, in een intern register opgeslagen. Als de data-strobe van



Afb. 5 Functie van de aansluitpennen van de UAR/T.

laag naar hoog gaat worden de databits, voorzien van start-, pariteits- en stopbits op serie-uitgang pen 25 gezet, ieder gedurende een bittijd.

Ontvanger

Als de serie-ingang, pen 20, laag wordt, start de ontvanger. Als na 8 klokpulsen de ingang nog steeds laag is, is het een geldig startbit. Na 16 klokpulsen, de ontvangerklok moet op pen 17 worden aangesloten, wordt het eerste databit binnen gehaald, vervolgens het tweede etc. Na ontvangst van geldige stopbits worden de ontvangen databits op pen 5 tot en met 12 gezet. De timing van de ontvanger is in afb. 4 te zien.

Keuze parameters

Bij het bespreken van de asynchrone seriële overdracht is vermeld hoe data wordt overgebracht. Daar werd alleen vastgelegd dat de uitzending begint met een startbit, gevolgd door de databits, eventueel een pariteitsbit, en geëindigd met stopbits. Deze beschrijving is juist voor de ASCII-code,

Tabel 4 Parameters voor het aantal databits.

NB1 pen 38	NB2 pen 37	Aantal databits
0	0	5
1	0	6
0	1	7
1	1	8

maar ook voor de Baudot-code die voor telex wordt gebruikt. De UAR/T kan beide opwekken en verwerken, doordat met een aantal ingangen de vorm van de uitzending kan worden vastgelegd. We kunnen bij de UAR/T kiezen uit 5, 6, 7 of 8 databits, die worden verzonden. Bij minder dan 8 worden de databits op pen TD1, TD2 etc. verzonden en de ontvangen databits op pen RD1, RD2 enz. gezet. Het aantal databits per karakter wordt vastgelegd door de spanning op pen 37 (NB2) en op pen 38 (NB1), zoals te zien is in tabel 4.

Ook het pariteitsbit kent een aantal mogelijkheden. Het kan worden weg-

Tabel 5 Pariteitsbit, dat wordt verzonden.

Geen pariteit pen 35	Pariteitskeuze pen 39	Pariteit
0	0	oneven
0	1	even
1	X	geen

gelaten of de soort pariteit (even of oneven) kan worden ingesteld door de spanning op ingang 35 en 39, zie tabel 5.

Het aantal stopbits wordt bepaald door de spanning op pen 36. Een „0” geeft twee stopbits. Als voor 5 databits is gekozen, wordt er echter 1½ databit verzonden en bij ontvangst gecontroleerd. Hierdoor wordt ook de Baudot-code mogelijk. Een „1” op pen 36 geeft één stopbit.

De parameters worden in een intern parameterregister opgeslagen als de control-strobe, pen 34, hoog is. In onze toepassing is dit opslaan van geen belang, maar het is mogelijk de UAR/T op de databus van een microprocessorsysteem aan te sluiten, waarbij een selectiesignaal, de control-strobe, nodig is om het parameterregister te laden met de gegevens op de databus. De parameters kunnen dus worden veranderd, bijvoorbeeld tussen zenden en ontvangen.

Statuswoord

De UAR/T heeft een aantal uitgangen waarop het verloop van zenden en

ontvangen is te volgen. De verzameling van deze uitgangen heet statuswoord.

Pen 13 geeft aan of de pariteit van een ontvangen karakter in overeenstemming is met de gekozen parameters (even of oneven, wel of geen pariteit). Een fout wordt aangegeven door het hoog worden van de uitgang; dit geldt voor alle foutuitgangen. Een synchronisatiefout, optredend als er geen geldig stopbit wordt gevonden, is op pen 14 te zien.

Overflow treedt op als de op de datauitgangen aangesloten apparaten (printer, videodisplay) niet aan de UAR/T hebben laten weten dat het ontvangen karakter is verwerkt. Dit signaal wordt op RDAV aangesloten. Een „0” is het teken dat het karakter is verwerkt. Overflow is te vinden op pen 15.

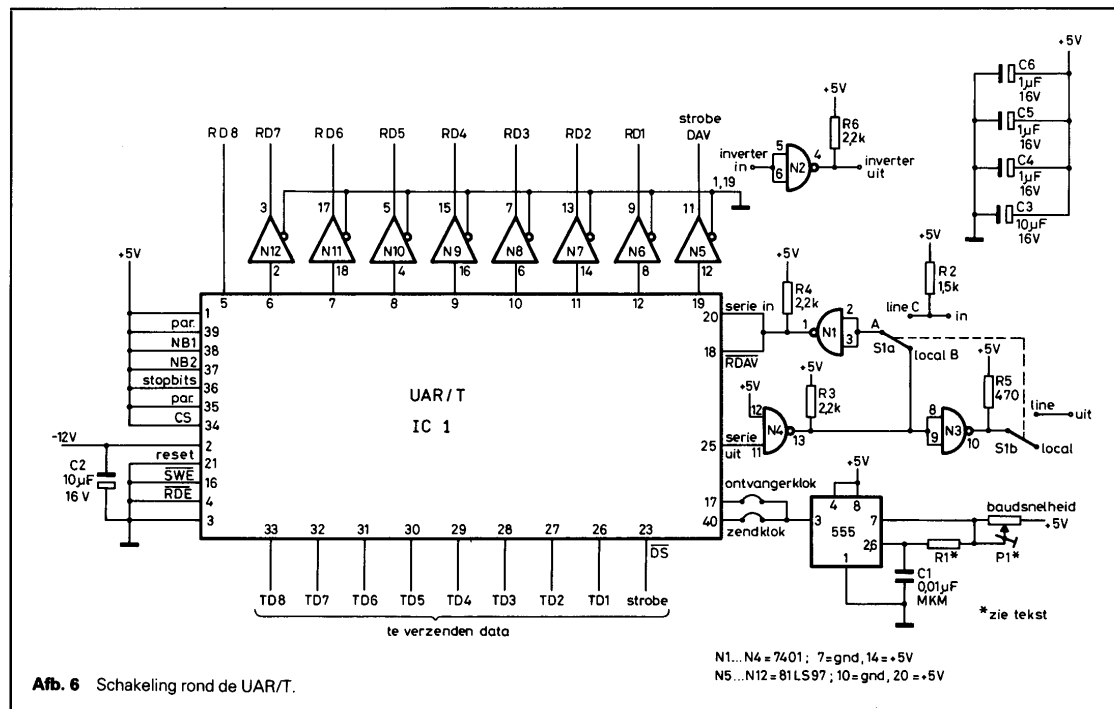
DAV is de uitgang, waarop de UAR/T met een „1” kenbaar maakt dat een volledig karakter is ontvangen en de databits op de data-uitgangen beschikbaar zijn. RDAV (pen 18) „reset” DAV (pen 19). Nadat RDAV laag is geworden, wordt DAV ook weer laag. Als RDAV permanent laag is, zal DAV één klokperiode van de ontvangerklok hoog zijn na ontvangst van een karakter.

Een op de zender betrekking hebbende uitgang is pen 22, zendbuffer leeg. Deze pen is hoog als er een nieuw karakter in de zendbuffer kan worden geladen. Het zenden hoeft dan nog niet klaar te zijn (pariteitsbit en stop-

bits nog). Pen 24, einde karakter, geeft aan dat de zender klaar is en dat een nieuwe data-strobe op pen 23 mag worden gegeven. Het statuswoord, zoals hierboven besproken, is een verzameling uitgangen die (behalve pen 24- einde karakter) tri-state uitgangen hebben. Een nul op pen 16, SWE, activeert de statuswoorduitgangen. Het is mogelijk ook de statuswoorduitgangen op de databus van een microprocessor aan te sluiten, waarbij SWE als selectiesignaal dient. Ook de datauitgangen, pen 5 tot en met 12, zijn tri-state uitgangen. Het selectiesignaal is RDE, pen 4. Een nul op RDE activeert de datauitgangen. Als de UAR/T in een microprocessorsysteem wordt opgenomen, kunnen datauitgangen, datauitgangen, statuswoord en parameterkeuze-ingangen allen op de databus worden aangesloten. In onze toepassing werkt de UAR/T zelfstandig en worden de tri-state ingangen altijd geactiveerd. De parameterkeuze ligt vast door de control-strobe hoog te laten zijn. Pen 21, RESET, is een ingang, waarmee na het inschakelen diverse registers in de beginstaat worden gebracht. Gewoonlijk wordt RESET permanent aan nul gelegd.

Schakeling

In afb. 6 is het prinsipeschema te zien van de UAR/T-schakeling. Voor de UAR/T-schakeling is een print ontworpen, waarvan het printontwerp is gegeven in afb. 7, de componentenop-



stelling in afb. 8. Centraal staat natuurlijk het UAR/T-IC, aangevuld met een klokgenerator en een interface in de vorm van een 7401-IC en een databuffer-IC 81LS97.

De parameterkeuze, zoals die in het prinsipschema is aangegeven, is vastgelegd op 8 databits, geen pariteitsbits en twee stopbits. Deze parameterkeuze is vastgelegd door draadbruggen op de print, zodat met behulp van tabel 4 en 5 en de gegeven beschrijving elk gewenst formaat kan worden ingesteld. De aangegeven parameterkeuze zal bij de meeste single-board computers, zoals KIM, SYM, AIM-65 etc., voldoen.

Uitgangsbuffer

Het UAR/T-IC is TTL-verenigbaar. Dat wil zeggen dat op de uitgangen van het IC een maximale belasting van 1 TTL-ingang mag worden aangesloten. Als op de datauitgangen een lange kabel wordt aangesloten, bijvoorbeeld naar een printer op één meter afstand, is het beter de uitgangen te bufferen. Het IC 81LS97 bevat 8 buffers, waarmee de strobe en zeven datauitgangen worden gebufferd. Van het IC 7401 is nog een poort over. De aansluitingen daarvan zijn als inverter geschakeld en naar buiten gevoerd. Deze inverter kan dienen om de strobe-puls te invertieren.

Klokkrequentie

Op de print is een klokgenerator aanwezig in de vorm van een oscillator rond het IC 555. Zowel de zend- als ontvangklok kunnen hierop worden aangesloten via draadbruggen. De universele opzet laat het toe andere klokk frequenties voor zowel zender als ontvanger te gebruiken door middel van de draadbruggen. De klokk frequentie moet 16 maal de gewenste baudrate bedragen. In tabel 6 is het verband tussen veel voorkomende baudrates en de klokk frequentie te zien. De waarden van de frequentiebepalende weerstanden bij de klokgenerator zijn afhankelijk van de gewenste klokk frequentie. De klokk frequentie volgt uit:

$$f = \frac{1,44}{(P_1 + 2 R_1) \times C_1}$$

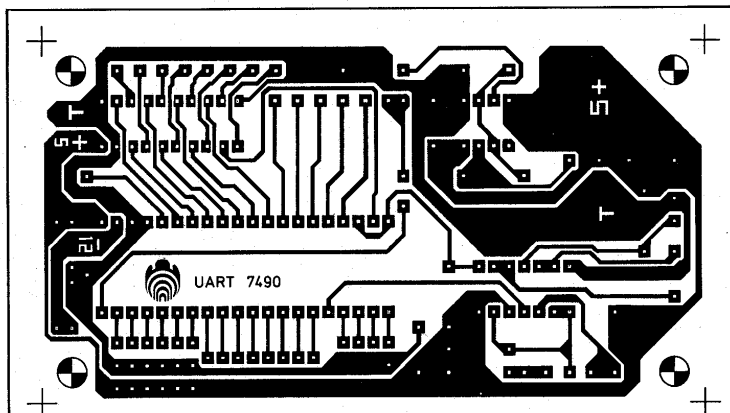
Kiezen we

$$R_1 = \frac{1}{2} P_1 \text{ en } C_1 = 0,01 \mu\text{F},$$

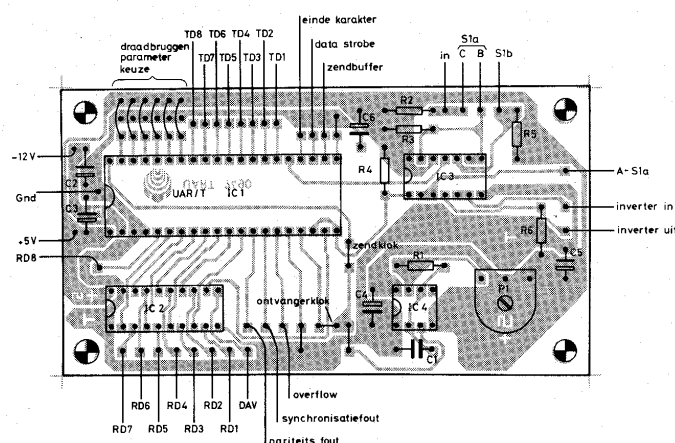
$$P_1 = \frac{0,7 \times 10^8}{f}$$

Voor P₁ kiezen we dan een potmeter met een waarde tweemaal die van de berekende P₁. R₁ is met de bovenstaande keuze gelijk aan de helft van de berekende P₁.

Nemen we bijvoorbeeld een baudrate van 110 baud, dan zal de klokk frequentie



Afb. 7 UAR/T-print.



Afb. 8 Componentenopstelling van de UAR/T-print.

tie 1760 Hz zijn. De berekende P₁ is dan gelijk aan 40 kΩ. We kiezen een instelpotmeter van 100 kΩ voor P₁ en voor R₁ een weerstand van 22 kΩ. Het UAR/T-principe laat een maximale afwijking van 5 % van de frequentie toe. Instellen van de juiste frequentie is het eenvoudigst met een frequentiemeter, maar een experimenteel vastgestelde frequentie in samenwerking met de computer voldoet ook. De exacte frequentie is niet belangrijk zo-

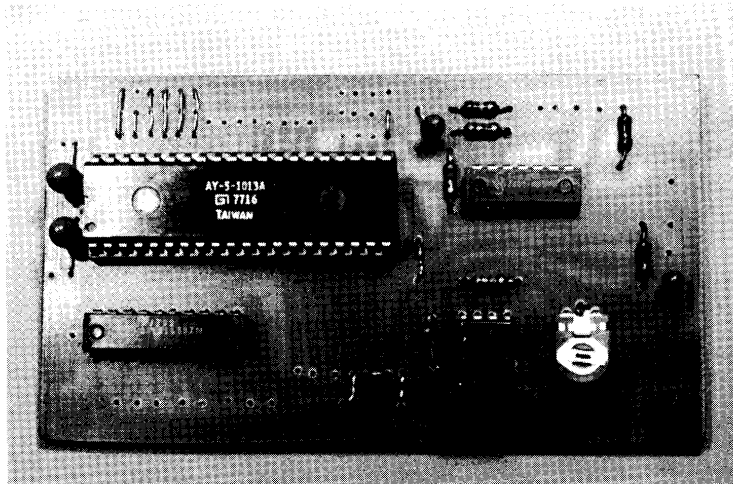
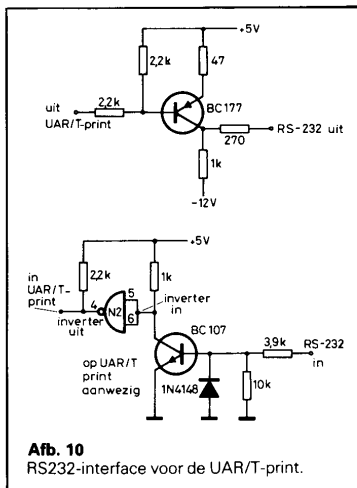
lang zender en ontvanger elkaar verstaan.

Computerinterface

De UAR/T-schakeling is geschikt voor full-duplex. Dat wil zeggen dat de UAR/T data naar de computer verzendt en de computer deze data weer naar de UAR/T terugzendt. Met de schakelaar S1 kan worden gekozen uit „line”, waarbij de data naar de computer wordt verstuurd en data van de computer kan worden ontvangen, en „local”, waarbij de data, door de UAR/T verzonden, rechtstreeks naar de ontvanger van de UAR/T wordt gevoerd. Niets wordt naar de computer gestuurd of ervan ontvangen. De aansluiting op de computer is afhankelijk van de interfaceschakeling van de computer. Is dit een interface volgens de RS232-standaard dan is in afb. 10 te zien wat er aan de UAR/T-schakeling moet worden toegevoegd. Bij de veel voorkomende zogenoemde „20 mA current loop” is aansluiten minder eenvoudig. Een voorbeeld van een 20mA-aansluiten op de KIM-1, AIM-65 of PC100 is in afb. 11 te zien.

Tabel 6 Baudrate en daarvoor nodige klokk frequentie van de UAR/T.

Baudrate (baud)	Klokk frequentie UAR/T (kHz)
75	1,2
110	1,76
150	2,4
300	4,8
600	9,6
1200	19,2
2400	38,4
4800	76,8
9600	153,6



Afb. 9 UAR/T-print met onderdelen.

Vaak is het handig de computer- en UAR/T-print te voorzien van een RS232C-interface. Voor de meeste computers is daarvoor wel een eenvoudig schema beschikbaar.

Voeding

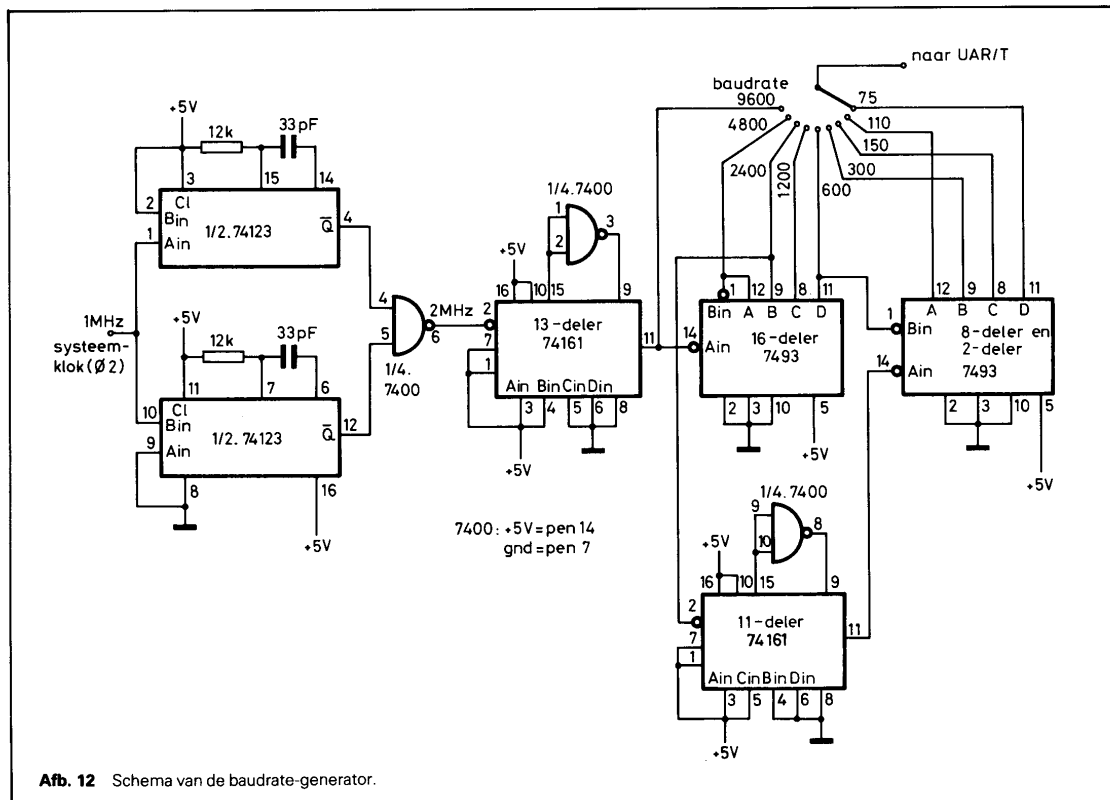
Een voedingsspanning van +5 V, 100 mA en -12 V, 5 mA is voldoende. Voor RS232C is -12 V, 25 mA nodig.

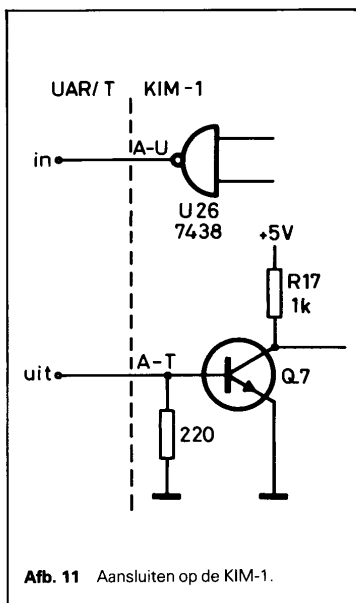
Testen van de UAR/T-print

De juiste werking van de UAR/T-print is eenvoudig vast te stellen door de schakelaar S1 op de stand „local” te zetten en een karakter op de data-ingangen te zetten en een strobe-puls te genereren. Op de datauitgangen moet het verzonden karakter terug zijn te vinden en een strobe-puls op de strobeuitgang verschijnen.

Baudrate-generator

De meeste microcomputers hebben een 1 of 2 MHz systeemklok, afkomstig van een kristalgenerator. Van deze systeemklok moet de baudrate-generator de gangbare baudrate-fre-





Afb. 11 Aansluiten op de KIM-1.

quenties afleiden. De gangbare baudrates die de schakeling tegelijk voortbrengt zijn 75, 110, 300, 600, 1200, 2400, 4800 en 9600 baud. De benodigde frequenties daarvoor zijn 16 maal zo hoog. De frequentie voor 9600 baud is zo 76800 Hz. Door de 1 MHz klok te verdubbelen tot 2 MHz en dan door 13 te delen komen we op de 9600 baudrate binnen 0,16%. Door verder door twee te delen worden de lagere baudrates gegenereerd. Alleen de 110 baud levert problemen op. Daartoe is een 11-deler opgenomen die de voor 110 baud benodigde frequentie toch nog binnen 0,6% van 1760 Hz brengt, n.l. 1748 Hz. In het algemeen laat het UAR/T-principe een afwijking van $\pm 5\%$ toe, zodat deze benadering nauwkeurig genoeg is.

Schakeling

In afb. 12 is de schakeling van de baudrate-generator te zien. Met behulp van twee monostabiele multivibratoren (74123) wordt het 1MHz-sig-naal verdubbeld tot 2 MHz. Bij een

systeem met een 2MHz-klok kan deze verdubbeling achterwege blijven. Het 2MHz-sig-naal wordt vervolgens door een als 13-deler geschakelde 74161 terug gebracht tot het 9600baud-kloksignaal. Consequent door twee delen (7493 IC's) levert alle klokfrequenties tot 75 baud. De 110 baud wordt van de 2400 baud afgeleid door te delen door 22 (74161 en gedeelte 7493).

Als bijproduct levert de baudrate-generator ook de voor de Kansas City benodigde zeer stabiele 2400 en 4800 Hz op. De keuze van de baudrate kan, zoals in afb. 12 is aangegeven, door een draaischakelaar worden gemaakt. Door de diverse frequenties naar een multiplexer en de uitgang van de multiplexer naar UAR/T te voeren kan de keuze van de baudrate ook onder software-controle verlopen, als de multiplexerstuuringsgangen door drie uitgangen van een PIA worden bestuurd. Een geschikte multiplexer is bijvoorbeeld de 74151. ■