

CR 10.1

Cursus constructie X1,1

B.J. Loopstra en C.S. Scholten.

1957

Cursus Constructie X-1 1957

onder leiding van
B.J. Loopstra en C.S. Scholten

Overzicht eenheden

Er wordt gebruik gemaakt van de volgende eenheden:

1. de T-eenheid
 - a. T1 (blauw)
 - b. T11 (blauw)
 - c. T2 (geel)
 - d. T22 (geel)
 - e. T3 (rood)
 - f. T33 (rood)
2. de D-eenheid (groen)
3. de E-eenheid (grijs)

Deze eenheden zijn naar de soort gekleurd zoals aangegeven. Het gehele arithmetische orgaan van de machine is opgebouwd uit deze eenheden

De contacten van de eenheden worden genummerd volgens het hieronder besproken systeem.

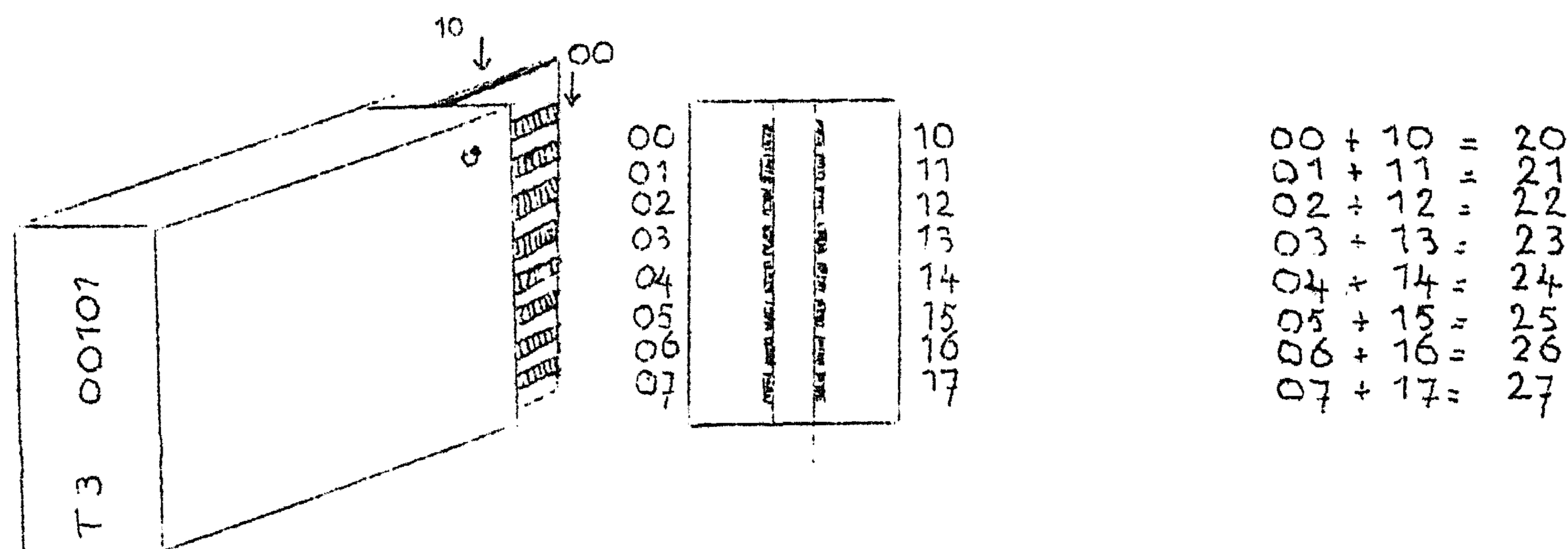


fig. 1.

Wanneer tegen de contactzijde van de eenheid wordt gekeken, is het nummer van het contact links bovenaan: 00, dat daaronder: 01, enz. Het contact rechts bovenaan draagt het nummer 10, dat daaronder 11, enz. Ook bestaat de mogelijkheid, dat twee contacten zijn doorverbonden. Als b.v. de contacten 02 en 12 onderling verbonden zijn, dan worden zij niet meer apart benoemd, maar tezamen n.l. contact 22.

Resumerend: wordt de eenheid gezien tegen de contactzijde, dan geldt: linkercontacten zijn "nul"-contacten, rechtercontacten zijn "tien"-contacten, doorverbonden contacten zijn "twintig"-contacten.

In de hoek bij contact 00 is een zwarte stip op de huls van de eenheid aangebracht, waardoor de bovenzijde van de eenheid is bepaald. Als tweede en niet onbelangrijkste bepaling geldt de wijze, waarop het nummer van de eenheid op de achterzijde is aangebracht. Wanneer de eenheid n.l. op de juiste wijze in de machine is geschoven, moeten alle nummers van rechts te lezen zijn, d.w.z. de onderzijde van de cijfers bevinden zich aan de rechterzijde van de machine.

N.B. Er is alleen gesproken over verticale plaatsing, maar het komt ook voor (in de geheugenkast), dat eenheden horizontaal zijn geplaatst. Dan moeten zij zodanig in de contactstrip worden geschoven, dat het opschrift normaal te lezen is. Dus de lettertekens van het opschrift moeten dan rechtop staan. Contact 00 is dan het contact uiterst links onderaan. Alle onderste contacten zijn de "nul"-contacten en alle bovenste contacten zijn de "tien"-contacten. De "twintig"-contacten veroorzaken geen moeilijkheden, aangezien boven- en ondercontact tezamen benoemd worden. (linkercontact is contact 20).

Elke plaats (contactstrip) in de machine draagt een bepaald nummer en hoort een bepaald type eenheid te bevatten. De eenheden bevinden zich in drie vlakken van de machine, n.l. de twee zijvlakken en de achterzijde, terwijl de achterzijde nog te verdelen is in twee vakken. Totaal zijn er dus vier vakken.

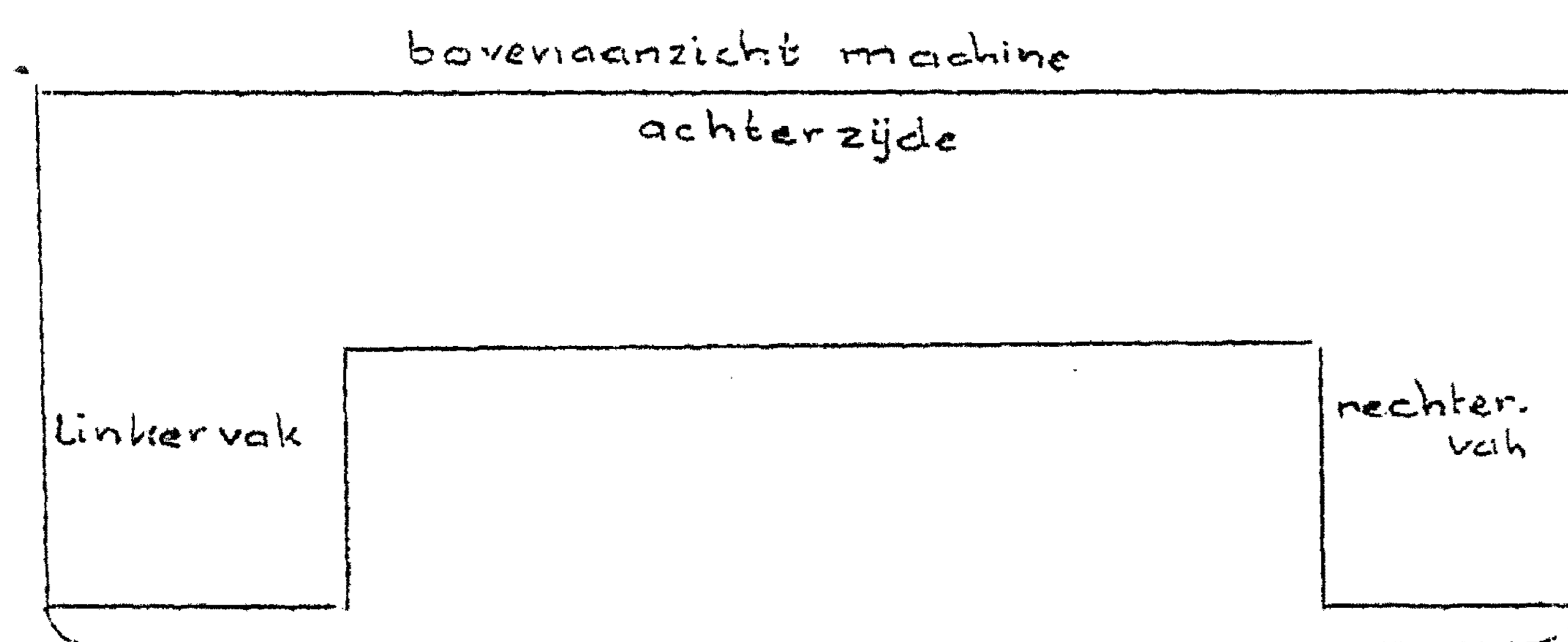


fig.2.

Het rechter vak bestaat uit 60 kolommen van elk 10 contactstrippen. Hier begint men te tellen links bovenaan (wanneer men tegen de soldeerpunten van de contactstrippen kijkt) met 0000. De strip daar direct onder is 0001 enz. tot de onderste strip 0009. Dan wordt weer bovenaan begonnen met nummer 0010, dit is dus de tweede contactstrip van links op de bovenste rij. De strip hier direct onder is 0011 enz. Dit vak telt zo

0000	0010	0020	0030	0040	0050	0060
0001	0011	0021	0031	0041	0051	0061

fig.3

door tot en met de eenheid 0599 en is daarmee vol. N.B. De rechter benedenhoek in dit vak is niet benut voor contactstrippen, maar voor aansluitmogelijkheden voor kabels. Deze kabelstekkers zijn geheel anders van vorm dan de contactstrippen, zodat vergissen is uitgesloten.

De achterzijde bevat 100 kolommen van elk 10 strippen. Ook deze strippen zijn genummerd. Begonnen wordt met nummer 1000 voor de eenheid links bovenaan. De eenheid daar direct onder is dus 1001, de eenheid vlak naast 1000 is 1010 enz. De achterzijde is verdeeld in twee vakken, terwijl de nummering doorloopt. Elk vak bevat 500 contactstrippen. In het rechtervak helemaal rechts onderaan van de achterzijde gezien bevindt zich dus de eenheid 1999.

Het linkervak bestaat ook weer uit 600 contactstrippen in 60 kolommen. Ook hier wordt verder geteld, door te beginnen links bovenaan met eenheid 2000 en zo vervolgens. De laatste eenheid is 2599.

Het type eenheid, dat moet worden geplaatst in een bepaalde contactstrip ligt vast op een tekening: het z.g. bollenveld (zo genoemd door de indeling in vakjes en het kleureffect van de diverse eenheden). Op deze tekening is het plaatsnummer aangegeven op een wijze, die het best verklaard kan worden, aan de hand van een voorbeeld. De plaats die met een kruisje is gemerkt (zie fig.4 blz. 4) wordt als volgt gevonden:

Volg de kolom recht naar boven, dan vindt men daar de drie meest significante cijfers (002). Weer uitgaande van het kruisje, de rij naar links of rechts volgen, dan vindt men zo het minst significante cijfer, in dit geval 1. Zo ontstaat een nummer van 4 cijfers n.l. 0021. Hiermede is dus de plaats bepaald. Het omgekeerde is ook mogelijk. B.v. gevraagd de plaats van

	000	001	002	003	004	005	006
0							
1			x				
2							
3							

contactstrip 0002. Zoek kolom 000 (de drie meest significante cijfers) en daarin rij 2.

In de vakjes wordt de soort-eenheid vermeld, die daar geplaatst behoort te zijn alsmede verdere gegevens betreffende de eenheid.

fig. 4

Verklaring van de eenheden en hun werking

Eenheid T1.

De X-1 is een rekenmachine, d.w.z. er worden getallen verwerkt en wel (in het algemeen) binaire getallen. Een binair getal

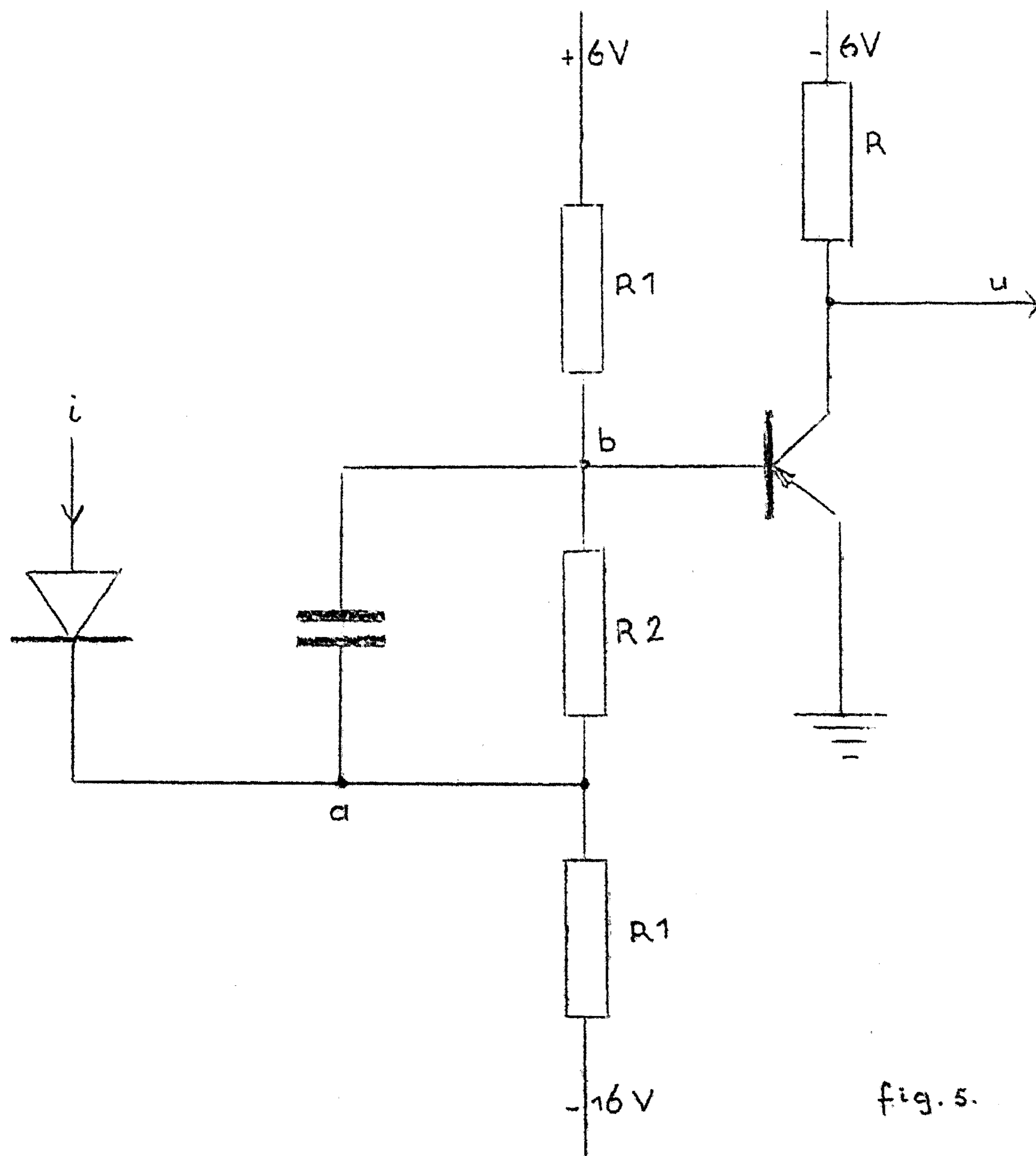


fig. 5.

wordt uitgedrukt in een combinatie van slechts twee cijfertekens. Gebruikelijk hiervoor is het cijfer 0 en het cijfer 1. Technisch onderscheiden we twee spanningsniveaux, die deze cijfers vertegenwoordigen, n.l. hoog en laag. Het verband met de cijfers 0 en 1 wordt later behandeld. De schakeling (zie fig. 5 en tek. E 15) heeft een in- en een

uitgang, resp. i en u genoemd. De weerstanden R_1 , R_2 en R_3 dienen voor de juiste spanningsdeling. Als het signaal hoog is, dan is de spanning iets lager dan 0 Volt (in het vervolg 0 Volt genoemd). Is het signaal laag, dan is de spanning iets hoger dan -6 Volt (in het vervolg -6 Volt genoemd).

1e geval. Stel de ingang is hoog (0 Volt, aarde), de spanning op punt a zal dan ca. - 0,3 Volt bedragen en op punt b: ca. + $\frac{1}{2}$ Volt. Door de + $\frac{1}{2}$ V aan de basis van de transistor zal er nagenoeg geen stroom lopen van de emitter naar de collector, zodat er dus ook geen stroom kan lopen door de weerstand R (Afgezien van een ev. belasting aan de uitgang u). Op punt u wordt dus - 6V gemeten, d.w.z. de uitgang is laag.

2e geval. Stel de ingang is laag (-6 V) de spanning, die zich dan op punt a in zal stellen is ca. -2 $\frac{1}{2}$ V en op punt b ca. - $\frac{1}{2}$ V. Door deze - $\frac{1}{2}$ V aan de basis van de transistor, zal de stroom van de emitter naar de collector "groot" worden. De spanning tussen emitter en collector ligt in de orde van grootte van 0.05 tot 0.2 Volt. De emitter ligt aan aarde, dus de collector zal 0.05 - 0.2 Volt negatief t.o.v. aarde zijn, d.w.z. punt u is iets lager dan 0 V (hierna genoemd 0 V, aarde) en dus hoog.

De condensator in de schakeling is aangebracht om snelle spanningsvariaties door te geven, d.w.z. het steile pulsfront te behouden.

Eenheid T11

Deze eenheid bevat twee eenheden T1.

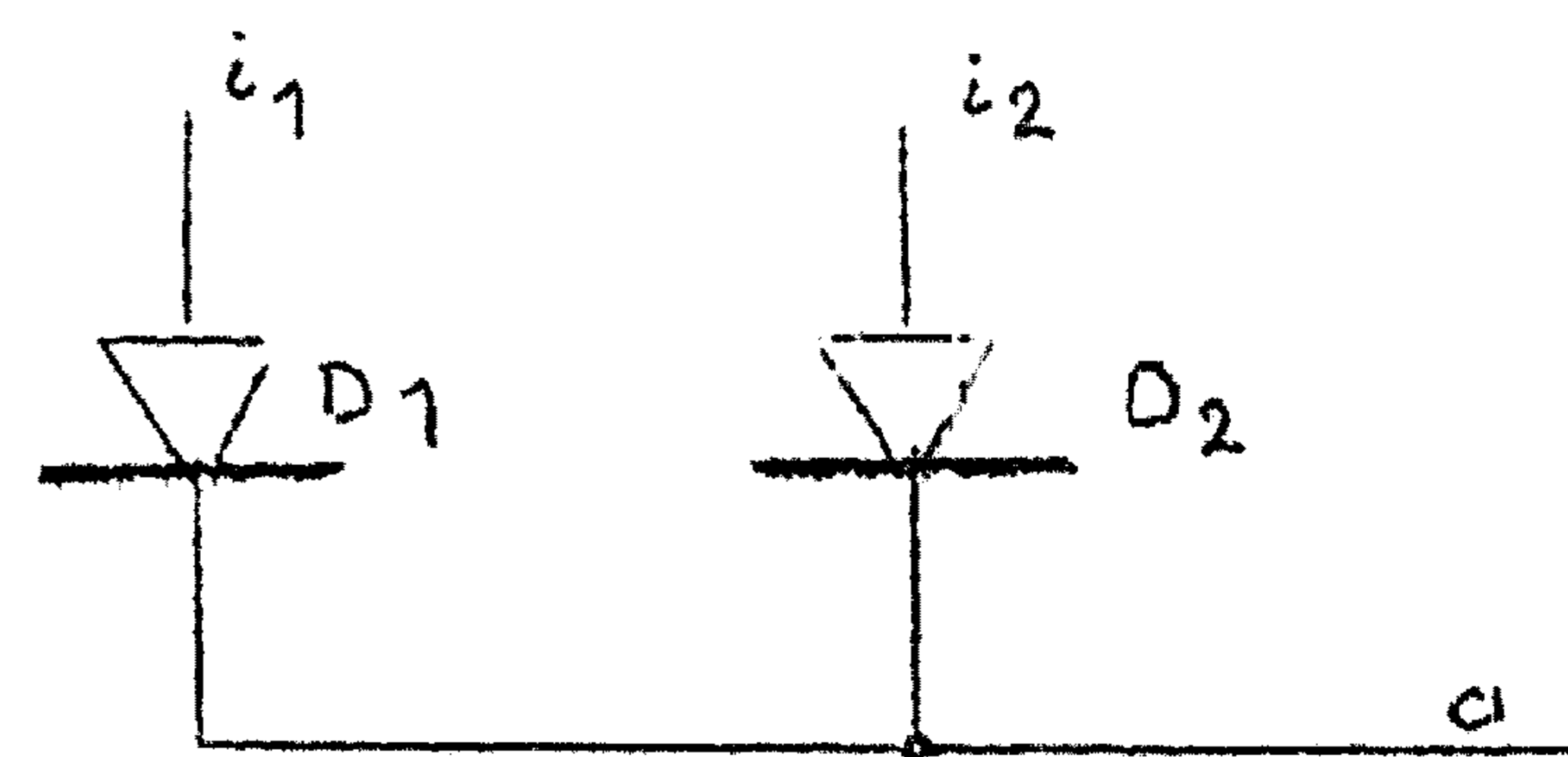


fig. 6

Eenheid T2

Het enige verschil van deze eenheid t.o.v. een eenheid T1 is, dat een T2 twee ingangen heeft (zie fig. 6 en tek. E15). Wordt aan i_1 een hoog signaal toegevoerd, dan is de uitgang dus laag (en omgekeerd). Wordt nu tevens ingang i_2 hoog gemaakt, dan heeft dit op de uitgang geen invloed, deze blijft laag. Wordt aan ingang i_1 een laag signaal toegevoerd en tevens aan ingang i_2 , dan is de uitgang hoog.

Wat zal er nu gebeuren, wanneer b.v. i_1 hoog wordt gemaakt en

i_2 laag? Wordt eerst ingang i_1 hoog gemaakt, dan is de uitgang u laag. Wordt nu tegelijkertijd i_2 laag gemaakt, dan zal D_1 blijven geleiden en punt a op ca. -0,3 V houden. Punt a is dan positief t.o.v. i_2 , zodat D_2 niet zal kunnen geleiden en zich als een grote weerstand gedraagt. Daardoor heeft i_2 in dit geval geen invloed op het spanningsniveau van a. Resumerend zien we dus, dat uitgang u slechts hoog kan worden, wanneer beide ingangen laag zijn.

Deze schakeling wordt een "èn-schakeling" genoemd, omdat de eerste ingang èn de tweede laag moeten zijn om de uitgang hoog te maken. Dit is dus beschouwd t.o.v. het lage spanningsniveau aan de ingang. Beschouwen we deze schakeling t.o.v. het hoge spanningsniveau aan de ingang, dan wordt deze schakeling een "òf-schakeling" genoemd, omdat slechts de één òf de andere hoog hoeft te zijn om de uitgang laag te maken. Een niet aangesloten ingang wordt beschouwd als een lage ingang. Deze namen zullen in het vervolg zonder verdere verklaring worden gebezigd.

Eenheid T22

Deze eenheid bevat twee eenheden T2.

Eenheid T3

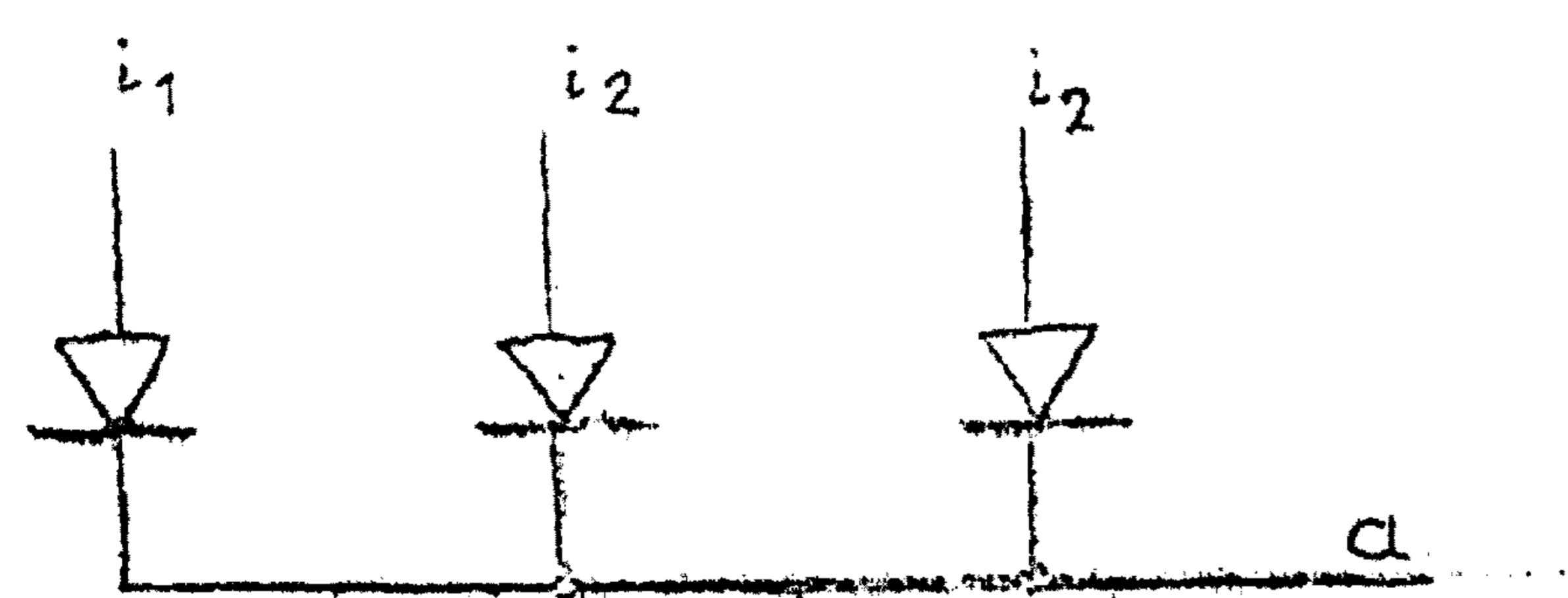


fig.7

De eenheid T3 heeft drie ingangen en is verder identiek aan T1. Hier is de uitgang dus steeds laag, wanneer slechts één van de ingangen hoog is. De uitgang wordt slechts hoog wanneer alle ingangen laag zijn.

Eenheid T33

Deze eenheid bevat twee eenheden T3.

Vermeld dient te worden, dat de eenheden T1, T2 en T3 alleen contacten van de "twintig"-serie bezitten.

Eenheid D (zie tek. E15)

Een D-eenheid bevat 5 dioden in drie groepen, n.l. 2-1-2 en dient om de mogelijkheden van de T-eenheden uit te breiden.

Door van één D-eenheid gebruik te maken, bestaat de mogelijkheid om van b.v. een T3 een T4, T5, T6, T7 of T8 te maken. De eenheden T4 t/m T8 kunnen dus slechts door uitwendige bedrading gemaakt worden.

Om een eenheid T3 uit te breiden tot b.v. een T5 moet contact 25 van de T-eenheid worden doorverbonden met contact 22 of 25 van de D-eenheid. En om een T8 te krijgen moet contact 25 van de T-eenheid worden doorverbonden met de contacten 22, 24 en 25 van de D-eenheid. Verdere uitbreiding is mogelijk door meer D-eenheden toe te passen.

Een complete schakeling van een T-eenheid inclusief de eventuele uitbreiding met D-eenheden wordt een "niveau" genoemd.

Met de tot nu toe behandelde schakeling van eenheden is alleen de "èn-schakeling" mogelijk (beschouwd t.o.v. lage ingangssignalen).

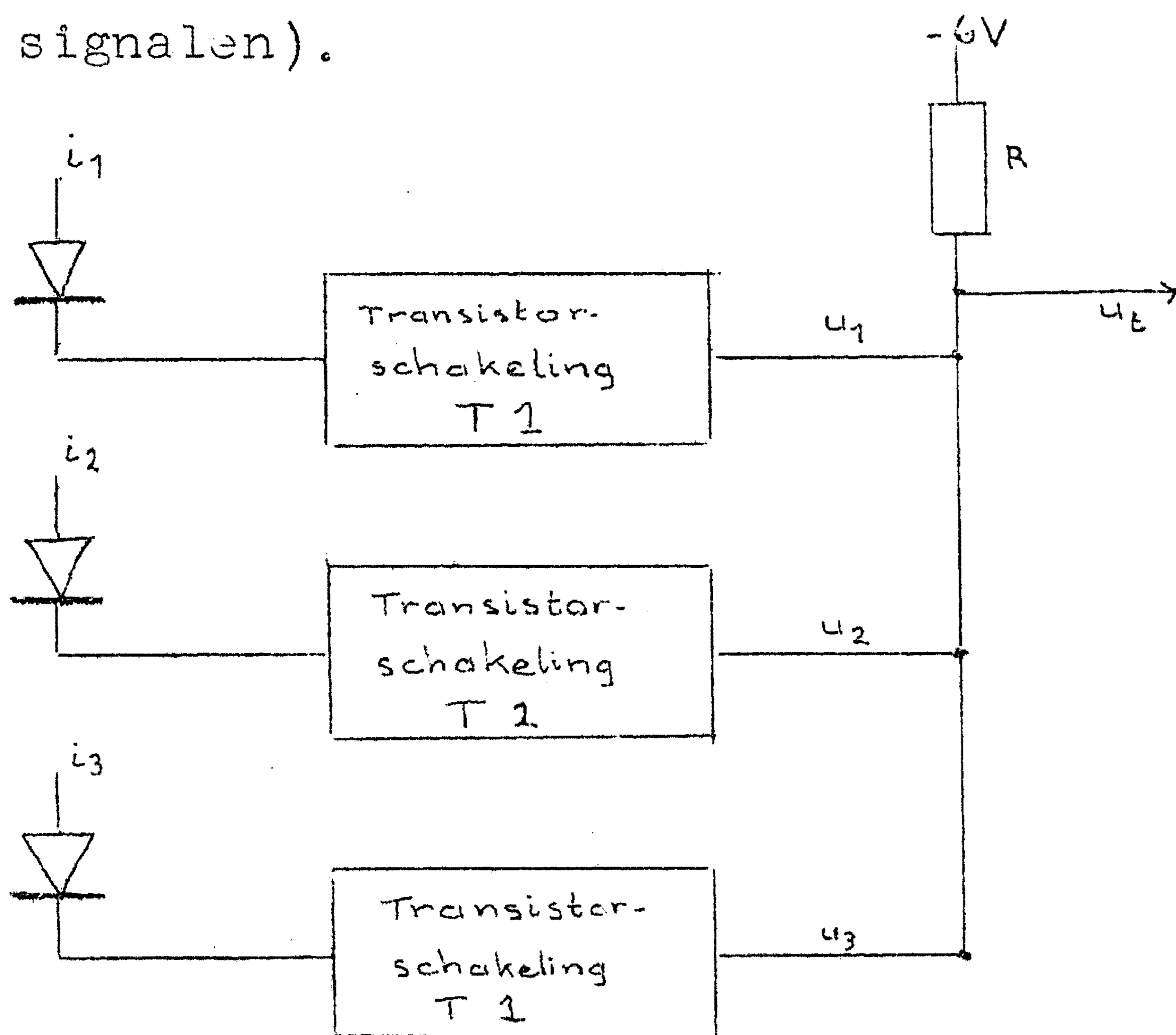


fig. 8

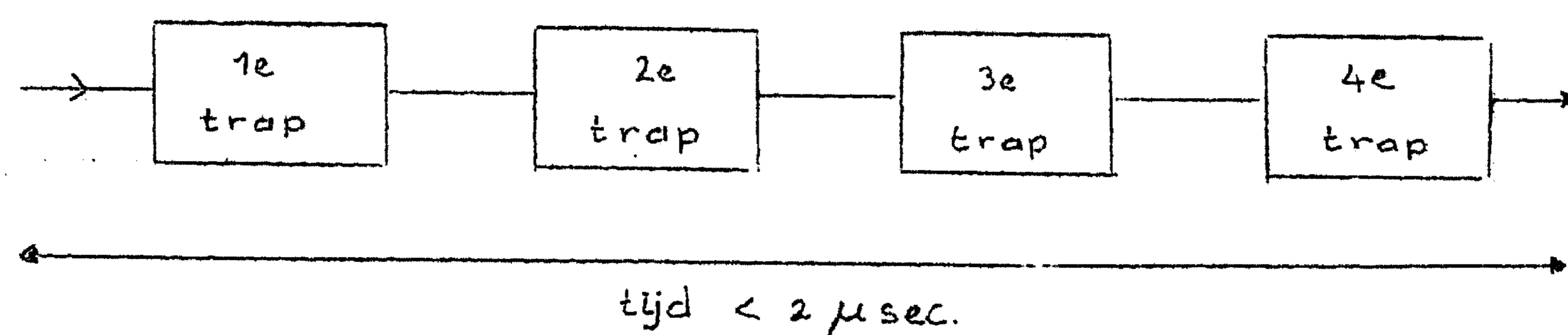
Verbinden we nu van drie eenheden T1 de uitgangen als in fig. 8 is getekend, dan ontstaat een "òf-schakeling" (beschouwd t.o.v. lage ingangssignalen). Immers wanneer twee uitgangen laag zijn, b.v. u_1 en u_2 en u_3 is hoog (d.w.z. i_3 is laag), dan worden alle uitgangen tezamen hoog, want uitgang u_3 brengt

alle collectoren via de collector van de andere T1 op 0 V, zodat u_t 0 V, (dat is hoog) wordt. Gebruiken we drie eenheden T2 in plaats van drie eenheden T1, waarbij de uitgang van slechts één T2 hoog was (d.w.z. alle ingangen van die betreffende T-eenheid zijn laag), dan zouden weer alle uitgangen ten gevolge van die ene hoog worden. Deze schakeling wordt

een "plaatje" genoemd.

Belasting van de eenheden

De uitgang van een niveau reageert na een zekere tijd op een signaal aan de ingang. De grenzen, waarbinnen deze reactie moet plaatsvinden zijn bij het ontwerp van de machine vastgelegd. Deze tijd mag maximaal $\frac{1}{2} \mu\text{sec}$ bedragen (tijdsvertraging). Staan b.v. 4 trappen achter elkaar, dan moet de laatste uitgang binnen $2 \mu\text{sec}$ reageren op een signaal aan de eerste ingang. Het is zeer belangrijk, dat deze tijden binnen de gestelde



grenzen blijven, aangezien hiervan de goede werking van de machine afhangt.

De genoemde tijds-

fig. 9 vertraging is boven-

dien afhankelijk van de belasting van een uitgang. Immers stroom leveren kost tijd. Behoeft er geen stroom geleverd te worden, dan is de uitgangsspanning veel eerder op peil, dan wanneer dit wel het geval is. (Het effect van lange toevoeleidingen wordt hierbij buiten beschouwing gelaten). Nu is gebleken dat een niveau nog voldoet aan de gestelde eisen, wat betreft de tijdsvertraging, wanneer deze wordt belast met niet meer dan 5 ingangen. (Een geleidende diode in een T-eenheid trekt ca. $1\frac{1}{2}$ mA). Dit wordt (vrij willekeurig) de "dynamisch" toelaatbare belasting genoemd.

Mag de tijdsvertraging maximaal $1 \mu\text{sec}$ bedragen, dan mag een uitgang met 10 ingangen worden belast. (Dit wordt (nog willekeuriger) de "statisch" toelaatbare belasting genoemd.

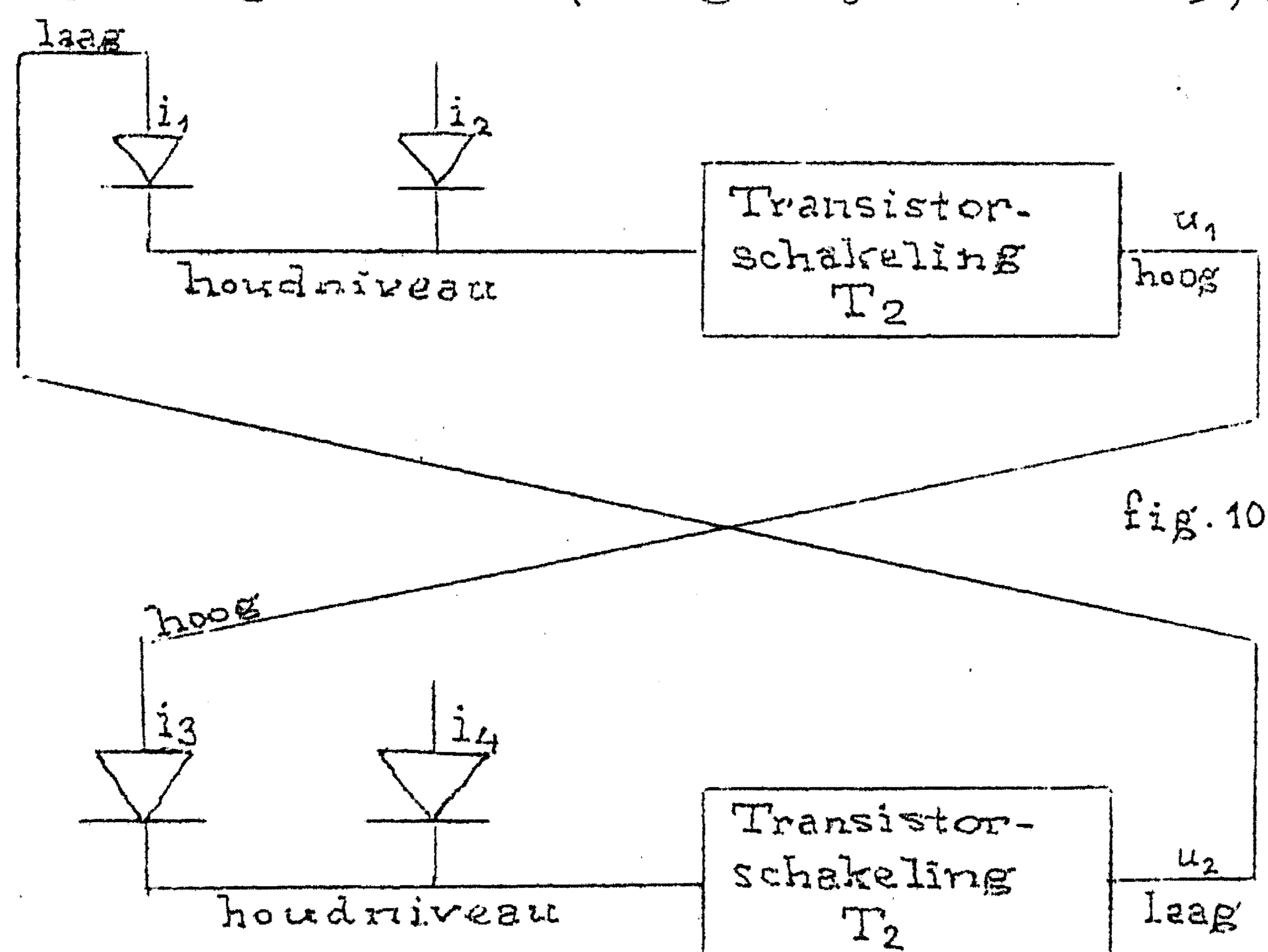
Resumerend: wanneer de vorm van de flanken, van het pulsfront, belangrijk is, dan mag een uitgang van een T-eenheid slechts met vijf ingangen worden belast. Is de tijdsvertraging minder belangrijk, dan geldt alleen de vraag: "Kan de transistor het gestelde aantal ingangen voeden?"

Het bovenstaande geldt natuurlijk niet in het algemeen, want het is best mogelijk, dat er aan de uitgang b.v. acht ingangen zijn aangesloten, waarvan er nooit meer dan 5 tegelijk stroom

aan de uitgang onttrekken. Dit "nooit" moet natuurlijk onomstotelijk vaststaan. Moeten meer dan 5 ingangen worden aangesloten, terwijl tevens het pulsfront bewaard moet worden, dan wordt gebruik gemaakt van een speciale eenheid.

Eenheid E (zie tek. E15)

De E-eenheid bevat twee identieke schakelingen, die elk voor zich in staat zijn om een belasting van maximaal 15 ingangen te voeden met behoud van het steile pulsfront (omkering binnen $\frac{1}{2}\mu$. sec). In deze schakeling bevindt zich een transistor, die door een andere instelling dan de transistor uit de T-eenheid tot deze grotere stroomleverantie in staat is. Aangezien deze transistor ook weer het signaal omkeert, dienen de voorafgaande T-eenheden het signaal nu niet om te keren. Om dit te bereiken zijn de transistoren in de T-eenheden als emitter-volgers geschakeld, zodat zij het signaal in dezelfde polariteit afgeven, als waarin zij het binnen krijgen. Zie de schakeling op de tek. LE15. Om maximaal 30 ingangen te voeden zijn twee eenheden E nodig, dus 4 identieke schakelingen, volgens tek. LE16. Hierbij dient wederom in het oog te worden gehouden, dat geen extra omkering plaatsvindt. Deze schakeling wordt toegepast, daar anders de transistors in de T-eenheden te zwaar zouden worden belast en de omkering niet binnen een $\frac{1}{2}\mu$. sec. zou plaatsvinden. Is deze tijdslimiet niet van belang, dan mogen twee helften van een E-eenheid worden aangesloten en elk direct worden belast met 15 ingangen. De contact aansluitingen van de E-eenheid zijn zodanig, dat er geen bovenkant is aan te wijzen. Daarom is dan ook geen stip aangebracht (vergelijk tek. E15).



Met de besproken eenheden is het mogelijk alle denkbare logische operaties uit te voeren. Als voorbeeld zal een "flip-flop" worden bekeken. Ingang i_1 is door verbonden met uitgang u_2 en evenzo i_3 met u_1 . Wanneer nu door een signaal op ingang i_4 dat niveau hoog wordt,

wordt uitgang u_2 laag en daarmee ingang i_1 eveneens laag, zodat u_1 hoog wordt. Uitgang u_1 houdt nu ingang i_3 hoog, waarmee de cyclus is gesloten.

Wordt nu i_4 laag en i_2 hoog, dan zal u_1 laag worden, waardoor i_3 laag wordt (i_3 en i_4 is een "èn-schakeling") en zo wordt u_2 nu hoog. En hiermede is wederom de cyclus gesloten. Het niveau, waardoor de flip-flop in een bepaalde stand wordt gehouden, wordt het "houdniveau" genoemd.

Beschrijving van de bedradingsboeken

Omdat het in de praktijk ondoenlijk is gebleken schema's van rekenmachines te tekenen, in die zin, dat elk draadje op de schema's voorkomt, zijn bedradingsboeken aangelegd. De voedingsspanningen staan op een lijst apart, zodat op de bladen in de boeken alleen de signalen voorkomen. Alle signalen benodigd voor een "plaatje" staan op één "blaadje". Alleen wanneer het eerste blad vol is, dan wordt gebruik gemaakt van een vervolgblad. Dit vervolgblad is dus eigenlijk een onderdeel van het eerste blad. Wanneer er een inverse van het eerste signaal is, staat dit eveneens volledig op hetzelfde blad. In dit geval staan dus de gegevens van twee plaatjes op één blad.

De bladen.

Op de bovenste regel staat:

1. de datum
2. het tekeningnummer

De gegevens in deze vakjes zijn alleen van belang voor intern gebruik.

3. de code.

In dit vakje wordt de "naam" van het plaatje geplaatst, afgekort in een enkele letter of een korte lettergroep. Bij sommige signaalnamen is een accent geplaatst. Dit maakt het onderscheid in polariteit mogelijk. Voor een signaal met een accent geldt de afspraak:

laag is 1

hoog is 0, terwijl het accent steeds bij de naam vermeld moet worden.

Voorbeeld: signaal A' wordt uitgesproken: "A-accent".

Bij signalen zonder accent geldt:

hoog is 1

laag is 0.

In het spraakgebruik wordt duidelijk gemaakt dat er geen accent aanwezig is.

Voorbeeld: Signaal B wordt uitgesproken: "B-onaccent", dit wil dus zeggen zonder accent.

De twee signalen met de bijbehorende onderscheiding komen b.v. voor bij de flip-flop, zoals deze is besproken in het voorgaande. Wanneer van deze flip-flop het signaal van de eerste uitgang met een accent wordt gemerkt, moet het andere signaal zonder een accent worden aangegeven.

Op de tweede regel staat de naam van het signaal voluit. Hieronder bevinden zich vijf kolommen. In horizontale richting zijn over alle vijf kolommen de gegevens betreffende één niveau vermeld.

1e Omschrijving

In deze kolom wordt het nummer en eventueel de naam van het niveau vermeld.

2e Plaats

Hier wordt de plaats van de eenheid (niveau) in de kast aangegeven met het daarbij behorende uitgangcontact. De uitgangcontacten die in deze kolom genoemd worden zijn (per plaatje) doorverbonden.

3e Ingangen

Deze kolom dient om de signalen te vermelden, die op dit niveau binnenkomen. Tevens wordt aangegeven het ingangcontact, waarop elk signaal moet worden aangesloten.

4e Plaats

Van eventueel gebruikte D-eenheden wordt in de 4e kolom de plaatsaanduiding vermeld.

5e Ingangen

Hier worden de signalen met bijbehorende ingangcontacten aangegeven, die op de niveau-uitbreiding binnenkomen, welke in de D-eenheid is ondergebracht.

Zoals eerder is vermeld worden de gegevens van het inverse van een signaal (indien aanwezig) op hetzelfde blad aangetekend.

Het bovenste gedeelte van de vijf kolommen bevat dan de gegevens betreffende het accentsignaal en het onderste deel die van het onaccentsignaal.

Onder laatstgenoemde vijf kolommen zijn wederom twee regels opgesteld, verdeeld in twee kolommen.

In de linkerkolom worden de gegevens vermeld van de "accent-uitgang". Op de bovenste regel staat de kleur van de draad, die verbonden wordt aan deze uitgang. De tweede regel vermeldt van de uitgang zelf zowel het contactnummer als de naam.

In het overblijvende gedeelte van de linkerkolom zijn alle ingangen genoemd, die met elkaar en deze uitgang zijn verbonden (dus ook deze uitgang zelf).

In de rechterkolom worden dezelfde gegevens vermeld, maar dan betreffende de "onaccent"-uitgang. Wanneer slechts één van beide genoemde signalen aanwezig is (er dus geen inverse is), dan wordt er ook maar één kolom gebruikt.

Bijzondere vermeldingen

Indien de uitgang gevormd wordt door een E-eenheid, dan is het noodzakelijk, dat dit ook op het blad te lezen is. Hiervoor is geen kolom of hokje aanwezig op het blad. Daarom wordt dit onder deze rubriek vermeld. Het onderste gedeelte van de vijf kolommen over de gehele breedte wordt hiervoor gebezigd.

Vermeld worden hier de plaats van de E-eenheid en de nummers van de contacten van de E-eenheid die moeten worden doorverbonden met die van de T-eenheid, alsmede de onderlinge doorverbindingen van de E-eeenheden zelf

Als voorbeeld zijn twee originele bladen bijgevoegd. De schakeling van een flip-flop is hierop ingevuld. Bij het eerste voorbeeld is de flip-flop samengesteld uit vier eenheden, terwijl bij het tweede voorbeeld slechts twee eenheden zijn toegepast. Wanneer de gegevens van deze eenheden in het bollenveld ingevuld worden, dan ontstaat het volgende beeld. (fig. 11). In het bovenste gedeelte van elk vakje wordt de naam van het plaatje aangetekend, in het midden het nummer van het niveau, terwijl in het onderste deel van het vakje het type van de gebruikte

eenheid wordt genoteerd.

	186	187	188	189	190	191	192
		B'	B'	B	B		
5		1	2	1	2		5
		T1	T2	T1	T2		

fig. 11

	185	186	187	188	189	190	
			B'	B			
5			1,2	1,2			5
			T22	T22			

fig. 12.

Moeten de gegevens van het tweede voorbeeld worden ingevuld in het bollenveld, dan gebeurt dit als volgt:

(zie fig. 12).

Nu staan beide niveau's in één vakje vermeld, omdat een dubbele eenheid is toegepast. Bij gebruik van een D-eenheid kunnen drie nummers voorkomen, wanneer de drie diodenschakelingen, die zich in deze eenheid bevinden gebruikt zijn voor

uitbreiding van drie verschillende niveau's.

Voorbeeld 1, Toegepast: twee eenheden T1 en twee eenheden T2

OMSCHRIJVING		PLAATS	INGANGEN	PLAATS	UITBREIDING	INGANGEN
NAAM: "Blaadje"						
		DATUM:		Tek.nr.		CODE: B'/B
1 Houd	1875	B'	20			
	27					
2 Lees	1885	A'	2ct'			
	27	20	21			
1 Houd	1895	B	20			
	27					
2 Lees	1905	A	2ct'			
	27	20	21			
KLEUR: geel				KLEUR: geel/wit		
UITGANG: B'-1895-27				UITGANG: B-1875-27		
1	U'			1	U	
2	B'-1875-20			2	B-1895-20	

Voorbeeld 2, Toegepast: twee eenheden T22

		DATUM:		Tek.nr.		CODE: B'/B	
NAAM: "Blaadje"							
OMSCHRIJVING	PLAATS	INGANGEN		PLAATS	UITBREIDING INGANGEN		
1 Houd	1875	B'					
	27	00					
2 Lees		A'	2ct'				
		10	11				
1 Houd	1885	B					
	27	00					
2 Lees		A	2ct'				
		10	11				
KLEUR: geel				KLEUR: geel/wit			
UITGANG: B'-1885-27				UITGANG: B-1875-27			
1	U'			1	U		
2	B'-1875-00			2	B-1885-00		

De aandacht wordt erop gevestigd dat het mogelijk is, direct alle gegevens betreffende een plaatje van de bladen uit het bedradingsboek af te lezen, zonder dat men zich eerst de schakeling voor de geest haalt en daarna pas de werking bekijkt.

Inlezen en terugstellen

Eerst zullen enkele uitdrukkingen voor bepaalde begrippen worden vastgelegd, zodat we deze uitdrukkingen in het vervolg steeds zonder nadere verklaring kunnen gebruiken.

Een flip-flop moet van de 0-stand (het accent-sigitaal is hoog, het onaccentsigitaal is laag) in de 1-stand (accent-laag, onaccent-hoog) gezet kunnen worden, terwijl ook het omgekeerde mogelijk moet zijn.

Wanneer een flip-flop in b.v. de 1-stand staat, dan zegt men: "de inhoud van die flip-flop is 1". Dit wordt ook met symbolen aangegeven. Bijv. $(B) = 1$, dit leest men als volgt: "de inhoud van B is 1".

Als flip-flop B dezelfde stand moet aannemen als flip-flop A, dan wordt genoteerd: $(A) \rightarrow B$. Dit wordt uitgesproken: "B neemt de inhoud van A over".

Wordt een flip-flop door een signaal in de 1-stand gebracht, dan noemt men dat het "inlezen" van die flip-flop.

Wanneer een flip-flop door een signaal in de 0-stand wordt gebracht, dan noemt men dat het "terugstellen" van de flip-flop. In principe zijn er vier gevallen mogelijk, n.l.:

- $(A) \rightarrow B$ terwijl:
1. $(A) = 0 \quad (B) = 0$
 2. $(A) = 1 \quad (B) = 0$
 3. $(A) = 0 \quad (B) = 1$
 4. $(A) = 1 \quad (B) = 1$.

Het overnemen van (A) in flip-flop B zal nu worden verklaard aan de hand van het voorbeeld op blz. 14. Dit overnemen kan op twee manieren gebeuren. Brengt men het A'-signaal op het ene plaatje van de flip-flop en het A-sigitaal op het andere, dan noemt men dit "dubbelzijdig overnemen". Brengt men een signaal, dat de flip-flop moet overnemen, op één van beide plaatjes, dan noemt men dit "enkelzijdig overnemen". Het

signaal $2ct'$ bepaalt de voorwaarde, waarop (A) alleen maar mag worden overgenomen. Dit signaal verandert periodiek van potentiaal en is kort in tijdsduur ten opzichte van het signaal dat de inhoud van flip-flop A weergeeft.

Geval 1. $(A) = 0, (B) = 0$, d.w.z. het signaal A' is hoog, evenals B' . De flip-flop zal nu niet op signaal A' reageren, want het houdniveau van het plaatje B' is reeds hoog. Evenmin reageert de flip-flop op signaal A, dat laag is, aangezien het signaal $2ct'$ hoog en het houdniveau van B laag is. Wanneer signaal $2ct'$ laag wordt, dan zijn alle ingangen van plaatje B laag. Ook dit heeft geen invloed op de flip-flop, d.w.z. (B) blijft nul.

Geval 2. $(A) = 1, (B) = 0$. Om de flip-flop in de 1-stand te brengen moet één niveau van het accent plaatje laag worden. Het houdniveau heeft in dit geval maar één ingang en daarop is het signaal B' aangesloten. Maken we echter van het tweede niveau beide ingangen laag, dan heeft dit tot gevolg dat de flip-flop omgaat van de 0- in de 1-stand.

Op de eerste van deze ingangen (contact 20) is het A' -signaal aangesloten, dat een 1 vertegenwoordigt en dus laag is. De flip-flop moet nu in de 1-stand gaan staan t.g.v. het inlezen van signaal $A' = 1$. Vandaar dat dit niveau het leesniveau wordt genoemd. Het signaal op de tweede ingang (contact 21) is het signaal $2ct'$, d.w.z. het A' -signaal kan alleen worden ingelezen gedurende de tijd, dat signaal $2ct'$ laag is.

Doordat de signalen A' en $2ct'$ beide laag zijn wordt uitgang B hoog en hiermee het houdniveau van B. Signaal A is dan ook hoog, zodat beide niveau's een laag uitgangssignaal geven n.l. B' .

Is (A) ingelezen, dan mag $2ct'$ weer hoog worden. Dit heeft dan geen invloed op het plaatje B' , want het houdniveau is laag, evenmin heeft het invloed op het plaatje B, want het houdniveau hier is al hoog. Signaal A is dan ook nog hoog, omdat dit signaal veel langer in tijdsduur is dan signaal $2ct'$.

Geval 3. $(A) = 0, (B) = 1$. Op een bepaald moment wordt het signaal $2ct'$ laag, terwijl signaal A (onaccent) ook laag is. (vertegenwoordigt een nul). Hierdoor wordt het tweede niveau van plaatje B laag, signaal B' wordt hoog enz.

De flip-flop is nu "teruggesteld".

Geval 4. $(A) = 1$, $(B) = 1$. Het houdniveau van plaatje B' is laag. Tengevolge van signaal A' wordt de eerste ingang van het leesniveau laag. Wanneer nu ook nog 2ct' laag wordt, dan is het gehele tweede niveau laag. Dit heeft op de flip-flop geen effect, omdat het houdniveau al laag was. Signaal A is hoog, wanneer 2ct' laag wordt, dit niveau blijft dus hoog. Ook dit heeft geen gevolg voor de flip-flop, zodat $(B) = 1$ blijft.

Om het "enkelzijdig" overnemen duidelijk te maken, zullen we de flip-flop van blz. 14 wijzigen in het onderstaande voorbeeld:

1 Houd	B'	2ct
2 Lees	A'	2ct'

1 Houd	B
--------	---

Alle gegevens, die niet belangrijk zijn voor deze verklaring, zijn weggelaten. In het kort zullen we ook hier de vier gevallen nagaan, die bij het dubbelzijdig overnemen zijn besproken. Geval 1. $(A) = 0$, $(B) = 0$ Signaal B' is hoog, dus het houdniveau is hoog, evenals het leesniveau t.g.v. signaal A'. De signalen 2ct en 2ct' mogen nu beurtelings hoog en laag worden, zonder dat dit invloed heeft op plaatje B', omdat van beide niveau's constant één ingang hoog is. De flip-flop blijft dus een nul vertegenwoordigen.

Geval 2. $(A) = 1$, $(B) = 0$. De flip-flop staat in de 0-stand en moet signaal A', dat laag is, overnemen. Wanneer nu 2ct' laag wordt, dan zal het tweede niveau van plaatje B' de flip-flop doen omgaan in de 1-stand. Hierdoor wordt signaal B hoog, signaal B' laag, maar signaal 2ct is nog hoog. Het houdniveau van het B'-plaatje kan zijn functie dus pas uitoefenen, wanneer 2ct (onaccent) laag wordt. Zou het moment, waarop 2ct' hoog wordt vroeger zijn, dan het tijdstip, waarop 2ct laag wordt, dan oefent het houdniveau zijn functie nog niet uit, terwijl het leesniveau weer hoog wordt. Daardoor zou de flip-flop B weer teruggesteld worden. Om nu zeker te zijn, dat flip-flop B inderdaad (A) inleest, is het dus beter, wanneer 2ct' iets

later (in tijd) komt dan $2ct$ (fig. 14) zodat het houdniveau

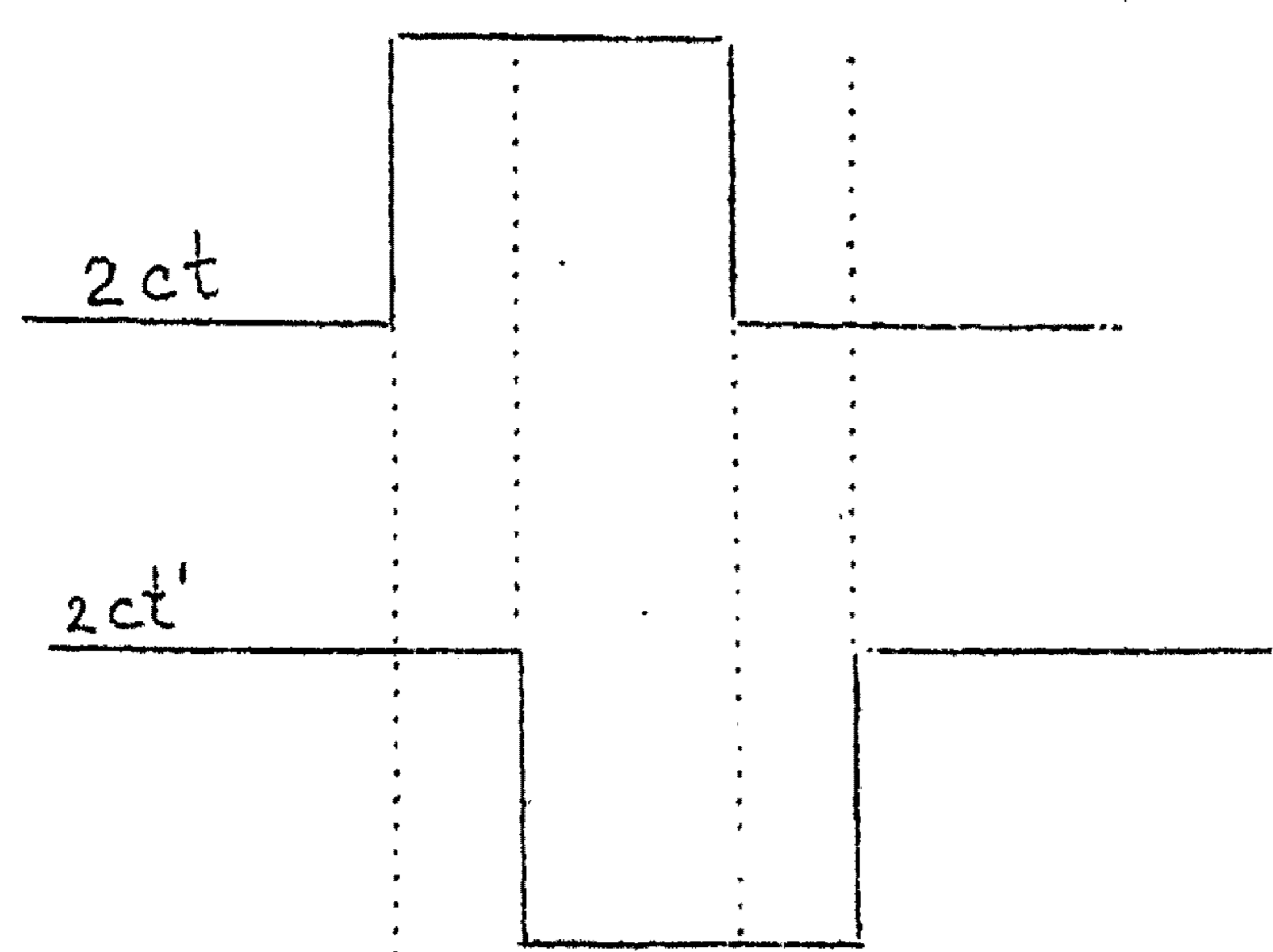


fig. 13

laag is, voordat het leesniveau hoog wordt. Nu blijft de flip-flop in de 1-stand staan.

Geval 3. (A) = 0, (B) = 1.

Het signaal A' is hoog, daardoor wordt het leesniveau hoog.

Wanneer nu het houdniveau ook hoog wordt door signaal

$2ct$, dan gaat flip-flop B om in de nul-stand en is dus teruggesteld.

Geval 4. (A) = 1, (B) = 1. Het houdniveau van plaatje B' is, evenals het signaal A', laag. Nu zal het signaal $2ct$ op een zeker tijdstip hoog worden en daardoor het houdniveau B' ook hoog maken. De flip-flop is nu teruggesteld. Even later (zie fig. 14) wordt signaal $2ct'$ laag, maakt het leesniveau laag en zet daarmee de flip-flop in de 1-stand. Het houdniveau gaat zijn functie nu uitoefenen op dezelfde wijze, als besproken is bij het 2e. geval.

Blokschema

Alle gegevens, die de X-1 nodig heeft, worden uitgedrukt in een reeks binaire tekens. Zo'n reeks tekens noemt men een "woord". In de X-1 kan een woord uit maximaal 27 tekens (bits) bestaan. Een woord is niet altijd een getal, maar kan ook een opdracht of een codewoord voorstellen. Stelt het woord een getal voor, dan geeft het meest significante bit aan, of het getal positief of negatief is, zodat er 26 bits overblijven voor de getalwaarde. Wanneer een woord een opdracht tot een zekere handeling (in het vervolg kortweg: opdracht) voorstelt, bestaat dit woord uit twee gedeelten n.l. het functiegedeelte (12 meest significante bits) en het adresgedeelte (15 minst significante bits). In het functiegedeelte staat, wat er met het woord moet gebeuren, dat zich op de plaats bevindt, die door het adresgedeelte wordt bepaald.

Het eenvoudigste blokschema, dat het principe van de X-1 weer-geeft, is getekend in fig. 14. Het deel van de machine, waarin

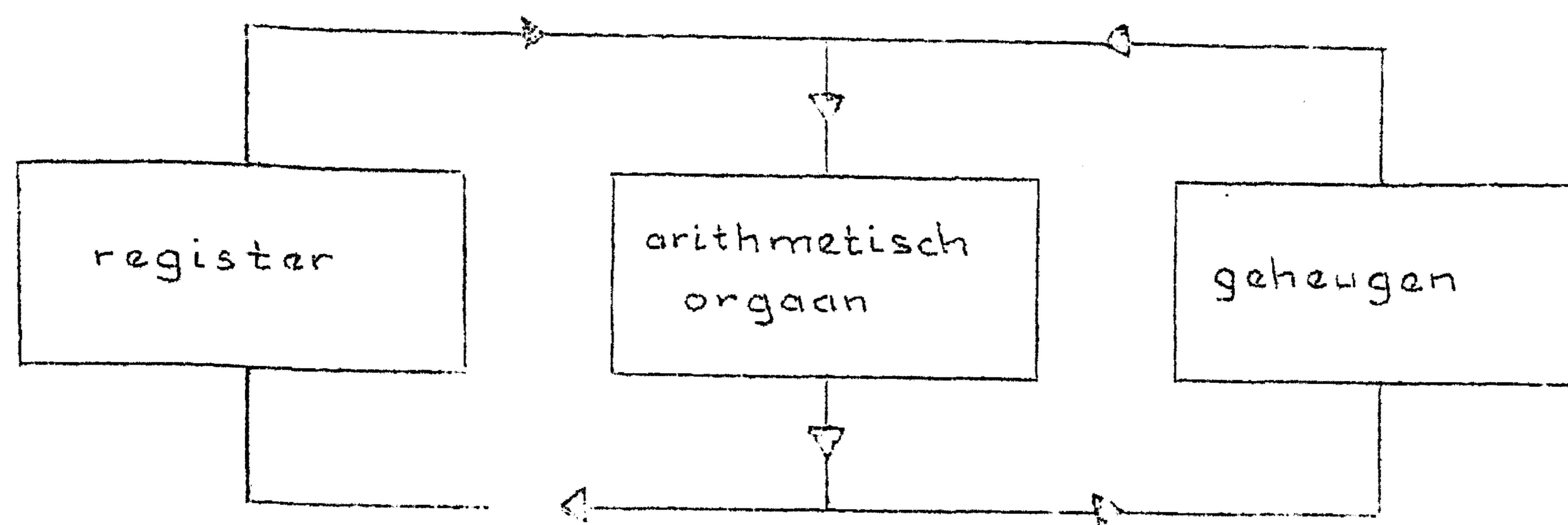


fig. 14.

de getallen worden bewerkt, heet het arithmetisch orgaan. In het geheugen kunnen vele woorden opgeborgen worden, m.a.w. het geheugen onthoudt alle woorden, die erin zijn "geschreven".

Iedere keer, dat het arithmetisch orgaan een bepaald woord nodig heeft, kan dit uit het geheugen worden gehaald. Het register kan ook onthouden, maar slechts één woord. Meestal heeft het arithmetisch orgaan twee woorden nodig om b.v. een optelling uit te voeren. Nu kunnen er niet twee woorden tegelijkertijd uit het geheugen worden toegevoerd aan het arithmetisch orgaan. Wel kan tegelijkertijd één woord uit het register en één woord uit het geheugen komen en dan aan het arithmetisch orgaan worden toegevoerd, dat ze dan verwerkt. De uitkomst kan zowel in het geheugen worden geschreven, als in het register.

Tek. X-3 geeft een uitgebreid blokschema van de X-1 weer. Rechts op de tekening is het geheugen getekend met vlak daaronder een versterker. Deze laatste dient om de zwakke signalen uit het magnetische geheugen te versterken, zodat ze op een bruikbaar spanningsniveau komen. Daarna wordt dit woord in het M-register overgenomen.

Bij de lijntjes, die de verbindingswegen voorstellen, staan getallen. Zo'n getal geeft het aantal gescheiden kanalen aan, waaruit die verbindingsweg bestaat. Volgen we de lijn uit het M-register, volgens de getekende pijlen, dan zien we, dat het woord o.a. opnieuw in het geheugen kan worden geschreven. Het geheugen bestaat uit twee delen n.l. het dode geheugen en het levende geheugen. De woorden in het dode geheugen zijn niet te vernietigen, omdat ze door de wijze van bedraden daarin zijn vastgelegd.

In het levende geheugen kan door de machine een woord worden ge-

schreven. Wordt dit woord overgenomen in het M-register, dan vernietigt men door deze handeling dit woord in het geheugen. Op deze zelfde plaats kan nu elk willekeurig woord worden geschreven. Moet het gelezen woord bewaard blijven, dan moet het opnieuw uit het M-register in het levende geheugen worden geschreven. Het geheugen is opgebouwd uit ferrietkernen. Het adresgedeelte van een opdracht heeft ruimte voor 15 binaire cijfertekens, d.w.z. er zijn $2^{15} = 32768$ nummers voor adressen beschikbaar. In het geheugen is dan ook plaats voor 32768 woorden van 27 bits.

Het lezen van een woord uit het geheugen duurt $16 \mu\text{sec}$, evenals het schrijven van een woord in het geheugen.

Vervolgens kan het functiegedeelte (12 bits) in het opdrachtre-gister functiegedeelte (ORF) worden geschreven.

Ten derde kan het woord (27 bits) uit het M-register in de logische opteller (LO) worden gebracht. Wat er daar gebeurt met dit woord zullen we later beschrijven.

In de laatste plaats kan een woord (27 bits) worden toegevoerd aan het coderend circuit "check". Dit circuit heeft een controlerende functie. Een woord bestaat uit een combinatie van enen en nullen. Nu zijn er twee mogelijkheden, het woord bevat een even aantal enen of een oneven aantal. Wanneer het aantal oneven is plaatst dit circuit een 0 voor het woord, is het aantal even, dan plaatst het een 1 voor het woord, zodat het totaal aantal enen in een woord steeds oneven is.

Vandaar dat er in het M-register ruimte is voor 28 bits. Indien ergens in een woord een cijfer verkeerd gelezen wordt, is het totaal aantal enen even i.p.v. oneven, waardoor de fout geconstateerd wordt. Is er een even aantal cijfers verkeerd gelezen, dan wordt dat woord door de machine toch goedgekeurd. De controle is dus niet ideaal, maar de mogelijkheid om verkeerd te lezen, zonder dat dit wordt geconstateerd, is vrijwel opgeheven.

Aan het M-register kan een woord van 5 bits worden toegevoerd uit de bandlezer. (BL). Dit apparaat leest, zoals de naam zegt een band. In deze papieren band kunnen gaatjes geponst zijn en wel vijf gaatjes per rij (pentade). Een pentade kan dus $2^5 = 32$ verschillende gedaanten hebben. Een gaatje op een bepaalde plaats

wordt opgevat als een één, het ontbreken van een gaatje als een nul. De band wordt afgetast met vijf fotocelletjes. De pentade, die door de fotocellen gelezen is, kan via versterkers worden overgenomen in de vijf minst significante trappen van het M-register.

Het terugleesregister (TR) kan eveneens een woord (6 bits) in het M-register schrijven. De X-1 kan ook een schrijfmachine bedienen. Het terugleesregister dient om de aanslagen op de schrijfmachine te controleren. De kaartapparatuur kan de gegevens van de ponskaarten in woorden van 12 bits in het M-register schrijven. De laatste mogelijkheid om in het M-register te schrijven wordt gevormd door het uitregister (U). In dit register komt de uitkomst van het arithmetisch orgaan. Deze uitkomst moet in het geheugen kunnen worden geschreven en dit kan alleen via het M-register.

Links onder op het blokschema is het opdrachtregister adresgedeelte (ORA) getekend. Hierin kan het adres tijdelijk bewaard worden. Daarboven staat de opdrachtteller (OT). Deze telt, zoals de naam zegt, de opdrachten, zodat steeds bekend is aan welke opdracht de machine bezig is. Bovendien wordt steeds automatisch het nummer van de opdracht met één verhoogd, zodat ook bekend is welke de volgende instructie zal zijn.

De inhoud van het B-register (B-16 bits) kan eventueel worden opgeteld bij het adresgedeelte van een opdracht, die uit het geheugen komt. Het getal in het B-register moet zowel positief als negatief kunnen zijn. Daarom zijn er 16 bits n.l. 1 bit voor het teken en de 15 adresbits. Ook is dit register te gebruiken als arithmetisch register. Het A- en S-register zijn in het algemeen gelijkwaardig. Alleen bij bepaalde opdrachten, b.v. vermenigvuldiging worden ze tezamen gebruikt, zodat dan een register van 54 bits ontstaat. Hierbij blijven beide tekenbits in functie, zodat resteren 52 cijferbits.

Het coderendcircuit opteller-in (OI) bepaalt welk register een woord mag zenden naar de logische opteller (LO).

De logische opteller, de carryvormer (CV) en de somvormer (SV) samen voeren de werkelijke optelling uit. De uitkomst wordt door het U-register overgenomen en vandaar in de eerste plaats naar

het A-register of het S-register gezonden. Het woord in het U-register kan ook worden overgenomen in het B-register, de opdrachtteller, het opdrachtregister-adresgedeelte en in het M-register.

Het type-ponsregister (TP) kan ook woorden (6 bits) overnemen uit het U-register. Het type-ponsregister bedient de schrijfmachine of de bandponser.

In de laatste plaats kan een woord uit het U-register naar de kaartapparatuur worden gezonden, om daar te worden verwerkt.

Het functiegedeelte van de opdracht.

De omschrijving van het functiegedeelte van de opdrachten wordt uitvoerig behandeld in rapport EL-1-N. De binaire code staat in kolom 3 van de tekeningen X-1 en X-2.

Kloksignaal en cijfertijden.

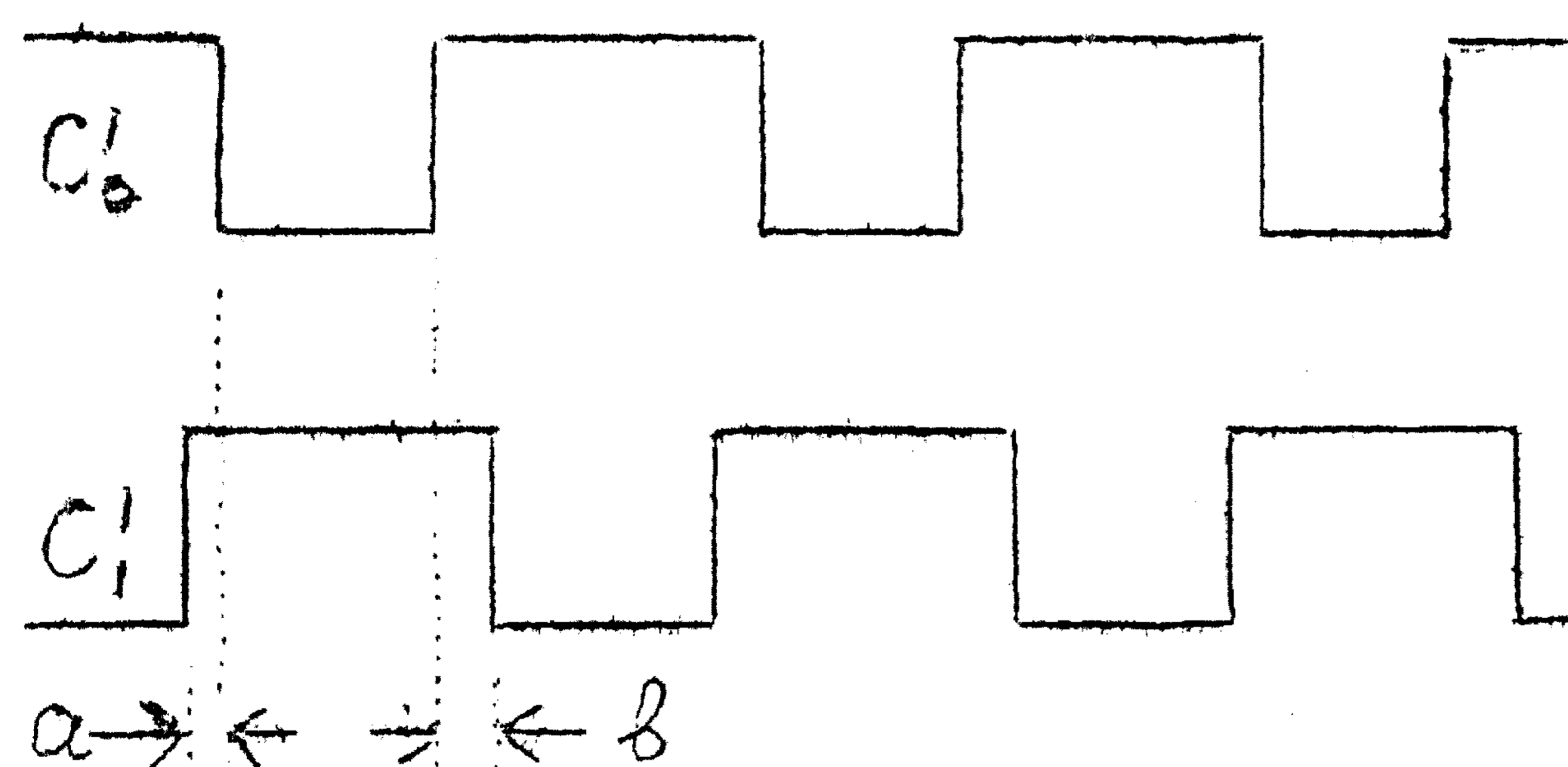


fig. 15

Uit het oscillatorsignaal worden twee blokpulsen (zie fig. 15) gevormd. Deze blokpulsen zijn synchroon. De afstanden a en b, die in de figuur zijn aangegeven, mogen nul zijn, zodat deze blokpulsen dan invers zijn. Deze pulsen worden het kloksignaal genoemd.

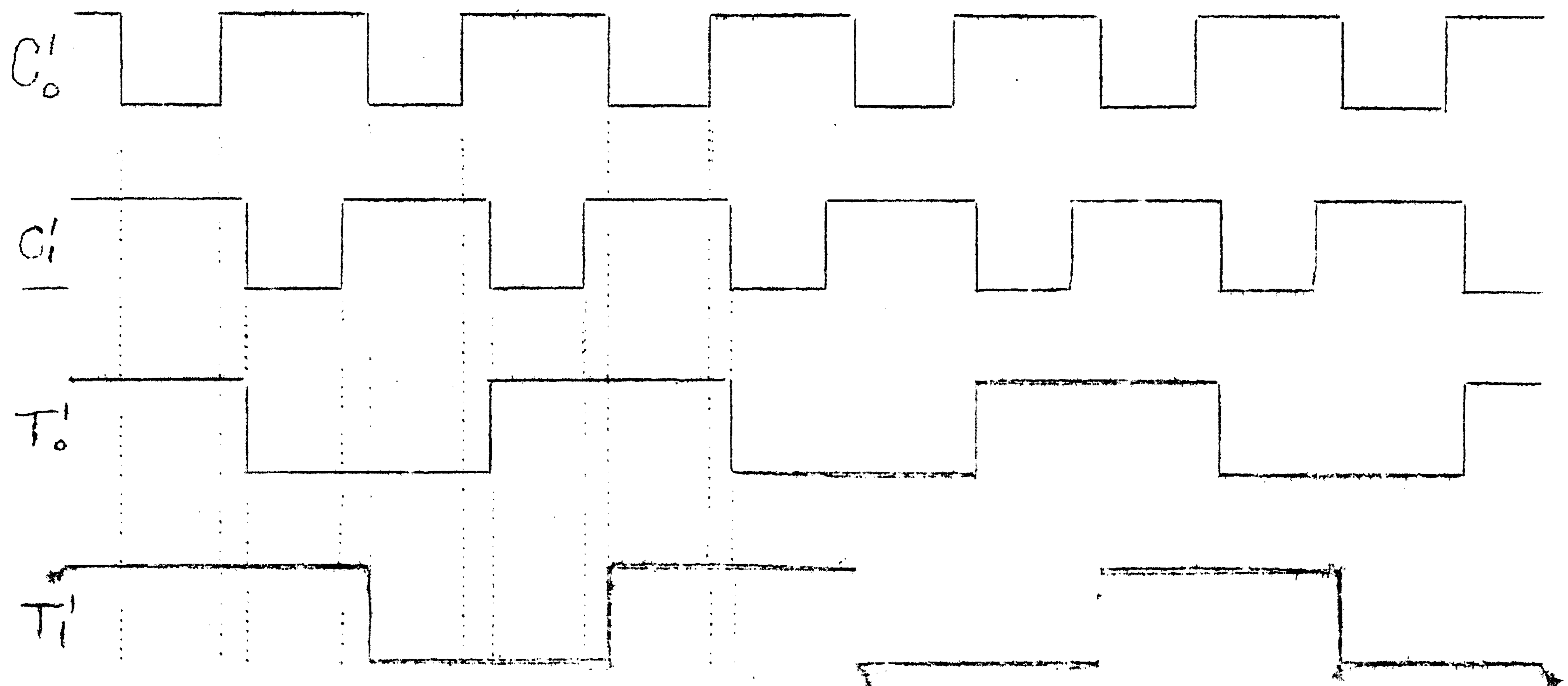
Het kloksignaal voeren we bij voorbeeld aan een teller toe, die uit twee flip-flops bestaat, n.l.:

		To'/To			T1'/T2	
1	Houd	To'		1	Houd	T1'
2	Lees	T1	C1'	2	Lees	To' Co'
<hr/>						
1	Houd	To		1	Houd	T1
2	Lees	T1'	C1'	2	Lees	To Co'

Om te beginnen nemen we aan dat To' en T1' beide hoog zijn. Gaat nu Co' laag, dan heeft dit op flip-flop T1 geen invloed, want To' blijft hoog. Het leesniveau van plaatje T1 wordt

laag, ook dit heeft geen invloed, want het houdniveau T_1 was reeds laag. Wanneer nu Co' weer hoog gaat, dan heeft ook dit geen invloed op de flip-flop, want houdniveau T_1 blijft laag. Gaat op zijn beurt C_1' laag, dan wordt het leesniveau To' laag (T_1 was nog laag), de flip-flop To gaat om, zodat To' nu ook laag is. Dit heeft nog geen invloed op flip-flop T_1 , want Co' is nog hoog. Wordt nu C_1' weer hoog, dan heeft dit geen gevolg. Daarna gaat Co' weer laag, dit maakt het leesniveau T_1' laag (To' was laag). Hierdoor gaat flip-flop T_1 om. Op flip-flop To heeft dit geen invloed, want C_1' is nog hoog. Wordt Co' weer hoog, dan heeft dit geen gevolgen. Vervolgens gaat C_1' laag, het leesniveau To wordt laag en dus To' hoog. Flip-flop To is dus omgegaan. Dit heeft echter geen gevolgen voor flip-flop T_1 , want Co' is nog hoog. Deze wisselwerking blijft zo steeds voortduren, alleen afhankelijk van de signalen Co' en C_1' . De aanname, dat het moment, waarop To' en T_1' , beide hoog zijn, wordt gevolgd door een Co' signaal, was kennelijk verkeerd. Bij een werkende teller brengt een kloksignaal altijd reactie teweeg bij een van beide flip-flops.

In figuur 16 zijn de boven beschreven gebeurtenissen in beeld gebracht.



figuur 16.

Wanneer we figuur 16 nader beschouwen, dan blijkt het volgende:

1^e signaal To' gaat beurtelings laag en hoog, iedere keer, dat signaal $C1'$ laag is,

2^e signaal $T1'$ gaat beurtelings laag en hoog, iedere keer, dat signaal Co' laag gaat.

De conclusie hieruit is, dat de frequentie van de tellersignalen de helft bedraagt van de frequentie van de kloksignalen.

Van de beide flip-flops kan de inhoud ook in tabelvorm worden genoteerd:

(T1)	(To)
0	0
0	1
1	1
1	0

Een dergelijke tabel noemt men een code. Bij deze code (Gray-code) verandert bij elke overgang (volgende regel) steeds één bit. Bovenstaande tabel kunnen we nog uitbreiden, zowel naar beneden, als naar links. Steeds blijft hij dan voldoen aan de gestelde voorwaarde, dat steeds maar één bit mag veranderen.

Dit is gedaan in onderstaande tabel:

	(T2)	(T1)	(To)
a	0	0	0
b	0	0	1
c	0	1	1
d	0	1	0
e	1	1	0
f	1	1	1
g	1	0	1
h	1	0	0

De aldus uitgebreide tabel is te verwezelijken met drie flip-flops, waarbij de inhouden van de flip-flops dus acht combinaties mogelijk maken. Er bestaan enkele mogelijkheden om een teller te maken, die aan de voorwaarde van de tabel voldoet. Eén van de schakelingen is als volgt:

To'/To	$T1'/T1$	$T2'/T2$
1 Houd To'	1 Houd $T1'$	1 Houd $T2'$
2 Lees $C1'$ $T1$ $T2$	2 Lees Co' $T2$ To'	2 Lees Co' $T1'$ To
3 Lees $C1'$ $T1'$ $T2'$		
1 Houd To	1 Houd $T1$	1 Houd $T2$
2 Lees $C1'$ $T1'$ $T2$	2 Lees Co' $T2'$ To'	2 Lees Co' $T1$ To
3 Lees $C1'$ $T1$ $T2'$		

We stellen de eis, dat de flip-flop T_0 om moet gaan iedere keer, dat C_1' laag is. De inhoud van T_0 moet volgens de tabel tweemaal 1 worden per cyclus en wel bij de overgang van a naar b, hierbij blijft gedurende de overgang gelden: $(T_1)=0$, $(T_2)=0$ en bij de overgang van e naar f, hierbij blijft gelden: $(T_1)=1$, $(T_2)=1$. Het houdniveau T_0' moet dus laag worden, wanneer de signalen C_1' , T_1 en T_2 laag zijn en de tweede maal, als de signalen C_1' , T_1' en T_2' laag zijn. Beide groepen signalen zijn op een leesniveau van het T_0' plaatje aangesloten, waarmee het gewenste resultaat is verkregen.

Ook moet de inhoud van T_0 tweemaal nul worden n.l. bij de overgang van e naar d, hierbij blijft $(T_1)=1$ en $(T_2)=0$ en bij de overgang van g naar h, hierbij blijft $(T_1)=0$ en $(T_2)=1$.

Het houdniveau T_0 moet laag worden, als de signalen C_1' , T_1' en T_2 laag zijn en de tweede keer tengevolge van de signalen C_1' , T_1 T_2' . De flip-flops T_1 en T_2 kunnen op de zelfde wijze worden afgeleid.

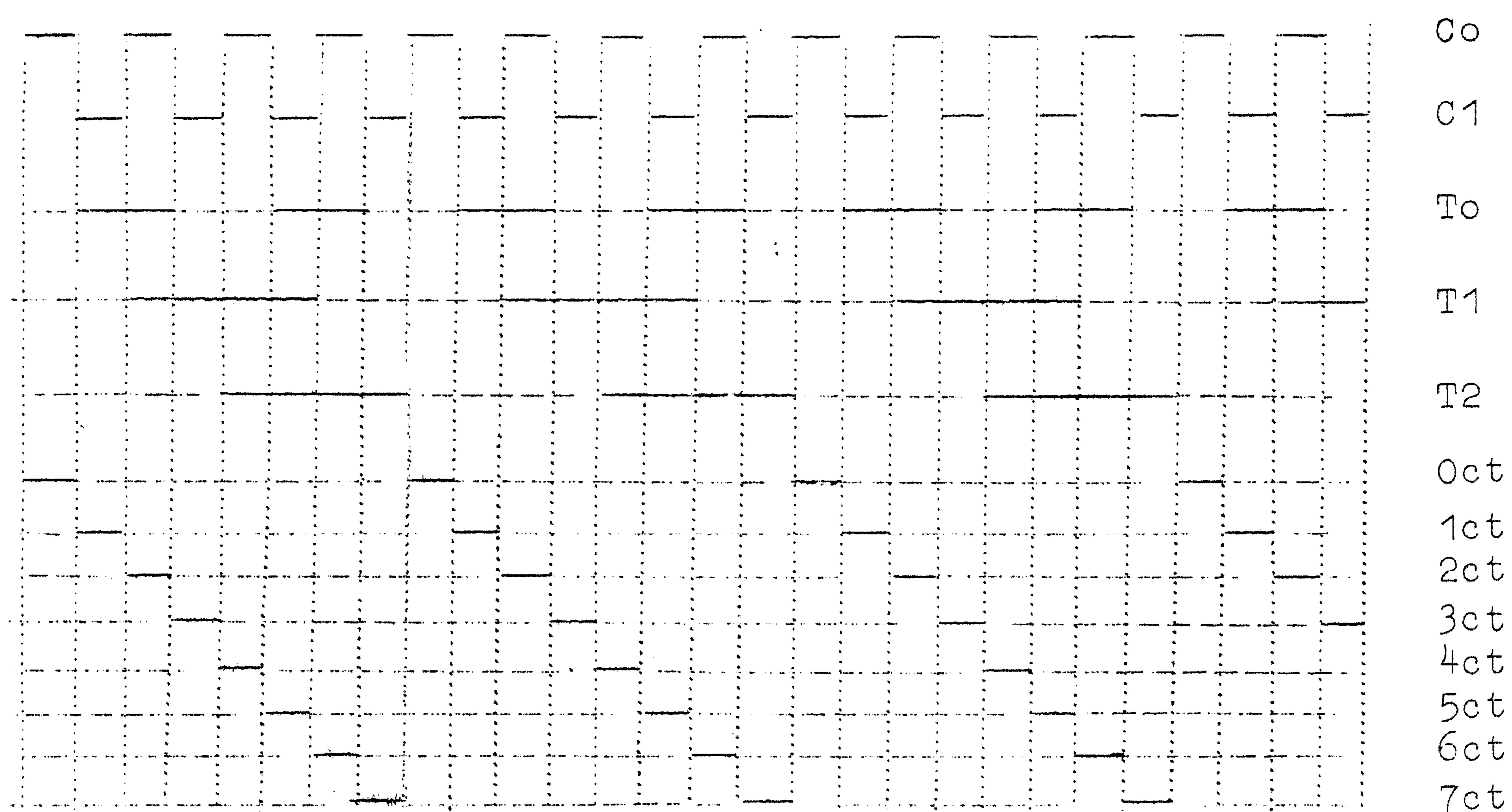
De uitgangssignalen van deze drie tellers kunnen worden toegevoerd aan acht niveau's, en wel op een zodanige wijze, dat wanneer de teller achtereenvolgens alle standen doorloopt, bij elke stand slechts één van deze niveau's laag gaat. Op het niveau dat laag moet gaan, gedurende stand a, zijn de signalen T_2 , T_1 en T_0 aangesloten. Op het niveau, behorende bij stand b, zijn de signalen T_2 , T_1 en T_0' aangesloten, op het niveau behorende bij stand c, de signalen T_2 , T_1' en T_0' , enz.. De aldus samengestelde teller wordt ook wel "ring" genoemd.

De hierboven besproken signalen duiden we aan als de zgn. cijfertijden (ct), waarbij 0ct overeenkomt met a, 1ct met b enz..

De overgang van 0ct naar 1ct wordt veroorzaakt door kloksignaal C_1' , de overgang van 1ct naar 2ct door signaal C_0' enz.. De kloksignalen, die een overgang tengevolge hadden, zijn in onderstaande tabel geplaatst naast de cijfertijd, die zij veroorzaakt hebben. Uit dit lijstje blijkt, dat de even cijfertijden het gevolg zijn van C_0' en de oneven van C_1' . Wanneer de pulsen in een diagram worden getekend, dan ontstaat het beeld, zoals fig. 17 dat

	2	1	0	
0ct	0	0	0	Co'
1ct	0	0	1	C1'
2ct	0	1	1	Co'
3ct	0	1	0	C1'
4ct	1	1	0	Co'
5ct	1	1	1	C1'
6ct	1	0	1	Co'
7ct	1	0	0	C1'

geeft. Hierin geeft een getrokken lijn aan wanneer het daarachter vermelde signaal hoog is. Uit dit diagram blijkt duidelijk, dat de cijfertijdpulsen steeds op de voorgaande aansluiten en dat als de laatste is geweest weer met 0ct wordt begonnen. Op grond van hun functie worden de drie flip-flops "cijfertijdring" (CTR) genoemd.



Zoals gezegd ontstaan de cijfertijden door de uitgangssignalen van de teller op niveau's aan te sluiten. Aan de uitgangen van deze laatste niveau's verschijnen de onaccent cijfertijden. Om nu de accentsignalen te krijgen, worden de onaccentsignalen elk op een niveau aangesloten, zodat de verlangde signalen aan de uitgangen van laatstgenoemde niveau's ontstaan. Tevens is hiermee bereikt, dat de accentcijfertijden iets later komen dan de

onaccent. Dit kan noodzakelijk zijn, zoals in het geval, dat is besproken op blz. 19 bij fig. 13.

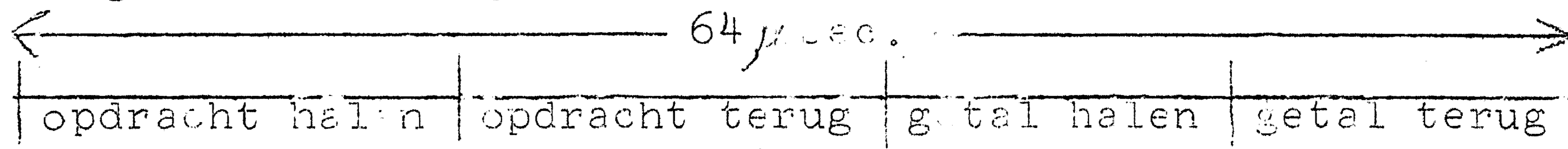
Het signaal Oct ontstaat door het feit, dat de signalen T2, T1 en To laag zijn. Dit komt eenmaal per cyclus voor. De situatie dat de signalen Co', T1 en To tegelijkertijd laag zijn, doet zich eveneens eenmaal per cyclus voor en wel op het moment, dat Oct' laag is. Deze signaalgroep is aangesloten op het leesniveau van plaatje T2. Wanneer we nu de drie ingangen van dit leesniveau vervangen door één ingang en daarop het Oct' signaal aansluiten, dan voldoen de flip-flops nog steeds aan de gestelde eisen. De actie van dit leesniveau heeft tot doel T2' hoog te maken gedurende de tijd, dat Oct' laag is. Maar gedurende die zelfde tijd is signaal Oct hoog. Sluiten we nu Oct op de 2^e ingang van het houdniveau T2' aan in plaats van Oct' op plaatje T2, dan wordt T2' hoog gedurende Oct. Door de uitbreiding van het houdniveau T2' met één ingang, kan het gehele leesniveau van plaatje T2 vervallen. Dit betekent een aanzienlijke kostenbesparing. Wanneer de drie flip-flops op de beschreven wijze zijn vereenvoudigd, worden de schakelingen als volgt:

To'/To		T1'/T1		T2'/T2	
1 Houd To' 3ct	7ct	1 Houd T1' 6ct		1 Houd T2' Oct	
1 Houd To 1ct	5ct	1 Houd T1 2ct		1 Houd T2 4ct	

Hieruit blijkt duidelijk de grote besparing, die het gevolg is van deze vereenvoudiging.

Het terugstellen van de tellers. (CCTR).

Een cijfertijdpulsje is 2μ sec. lang, d.w.z. iedere 16μ sec. begint de cyclus weer opnieuw. Veronderstel dat de machine de volgende handelingen moet uitvoeren:



Veronderstel verder, dat deze vier handelingen totaal 64μ sec. kosten, waarbij de onderlinge tijdsduur niet gelijk behoeft te zijn, dan zal de laatste handeling voltooid zijn aan

het eind van 7ct, wanneer het opdracht halen begon bij de aanvang van Oct. Dit houdt in dat een volgende handeling weer op een Oct kan beginnen. Als nu b.v. de inhoud van het B-register moet worden opgeteld bij het adres van deze opdracht, voordat hij wordt teruggeschreven, dan duurt dat b.v. 4μ .sec. De totaal tijd wordt daardoor 68μ .sec. en nu is de laatste handeling niet klaar vlak voor Oct, zodat 12μ .sec. verloren zouden gaan in afwachting van de eerstvolgende Oct-impuls. Daarom moet de teller teruggesteld kunnen worden na afloop van de laatste handeling, zodat direct met de volgende serie kan worden begonnen. Om dit doel te bereiken is het signaal CCTR ingevoerd. Er zijn verschillende gevallen, waarin signaal CCTR de teller terug moet stellen. De signalen, die de voorwaarde van één geval bepalen, zijn aangesloten op één leesniveau van plaatje CCTR'. Er zijn dus evenveel leesniveau's als gevallen, waarin CCTR in werking moet treden. Op elk leesniveau is een oneven cijfertijdsignaal aangesloten, waardoor bereikt wordt, dat de flip-flop alleen omgaat bij een sig-

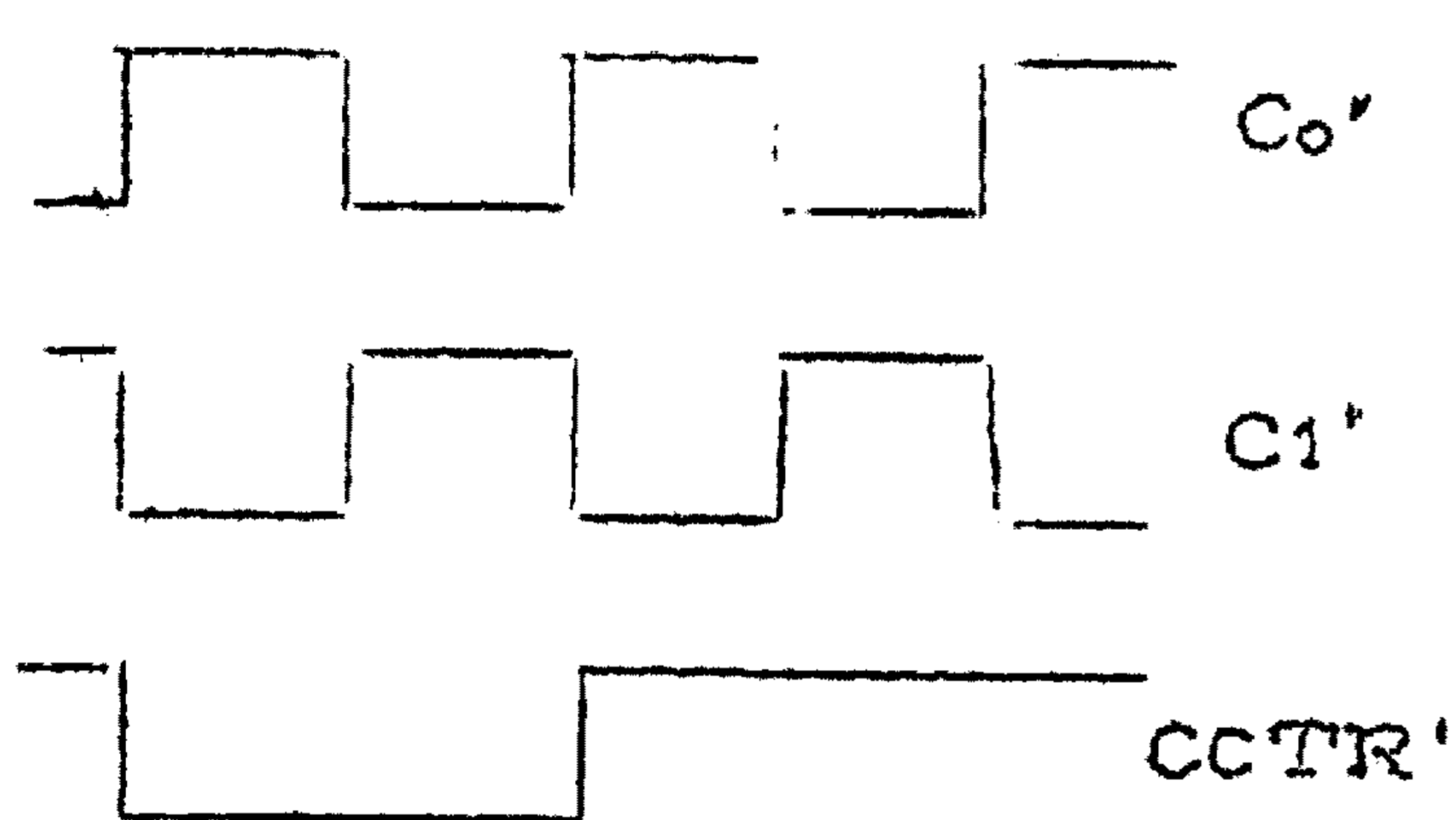


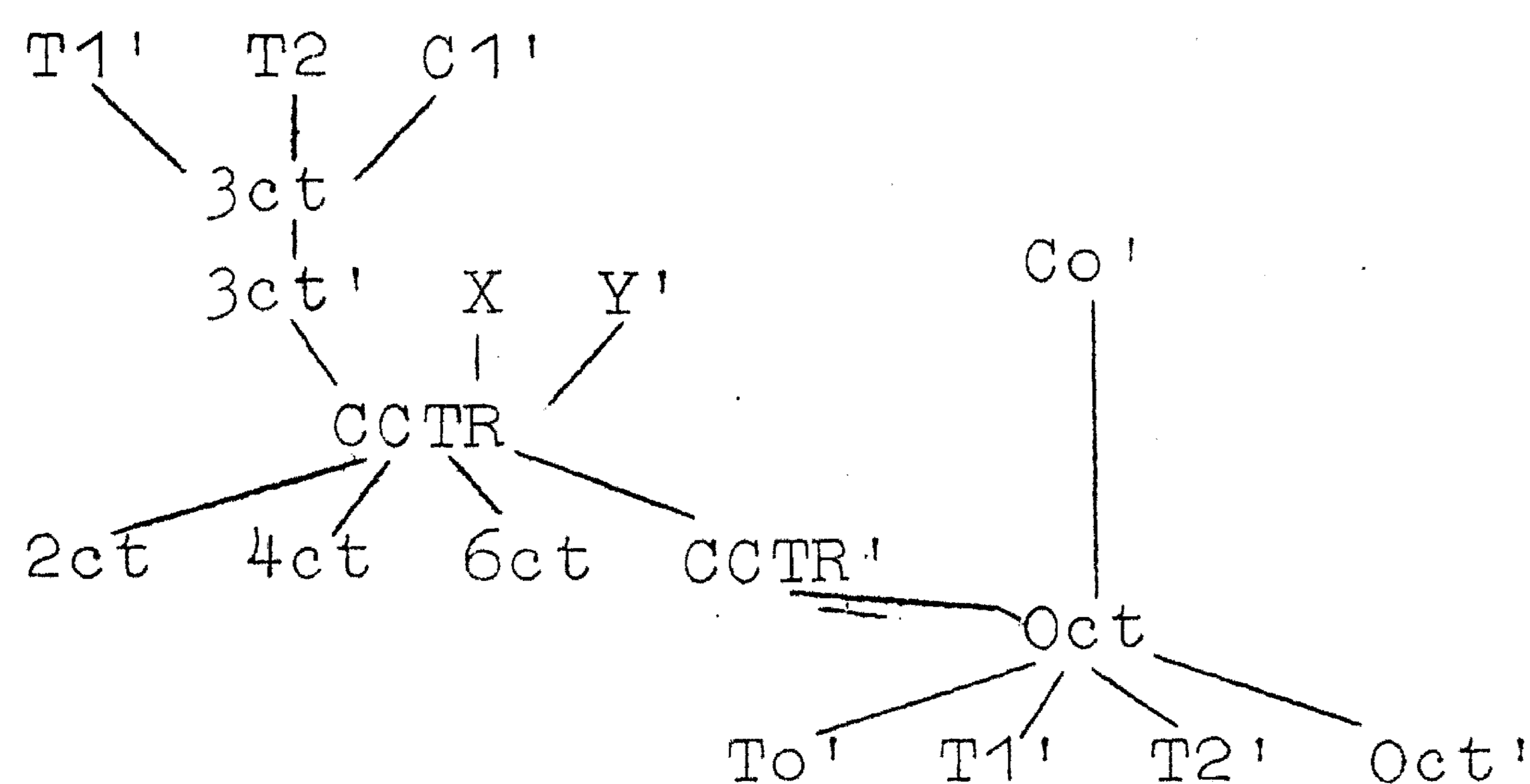
fig. 18.

signaal C1' (zie fig. 18). De signalen X en Y' geven zo'n voorwaarde weer, waarbij we veronderstellen, dat de signalen X en Y' laag zijn. Op 3 cijfertijd wordt het 2^e niveau van plaatje CCTR' laag door de signalen X, Y' en 3ct'. Zoals bekend is, ontstaat signaal 3ct' door het kloksignaal C1' en de signalen van de teller (T).

3ct'/3ct	CCTR'/CCTR	Oct'/Oct
1 T1' T2 C1'	1 Houd CCTR' Co'	1 To T1 Co'
1 3ct	2 .. X Y' 3ct'	2 CCTR' Co'
	1 Houd CCTR	1 Oct

Er vinden dus vier omkeringen plaats: nl. 2 voor het vormen van de betreffende oneven cijfertijd en twee voordat beide polariteiten van de CCTR klaar zijn. Iedere omkering duurt (iets korter dan) $\frac{1}{2}\mu$.sec. (zie blz. 8). Signaal CCTR' is dus zeker laag 2μ .sec. na het laaggaan van signaal C1', zodat gedurende het eerstvolgende signaal Co' de CCTR gegarandeerd beschikbaar is. Dit houdt in, dat dus alleen iedere even cijfertijd door toedoen van de CCTR vervangen kan worden door Oct en dit moet één cijfertijd van te voren bekend zijn in verband met het feit, dat steeds het signaal C1' de actie voorbereidt.

Het houdniveau van plaatje CCTR' heeft natuurlijk op de eerste ingang het signaal CCTR'. Op de tweede ingang staat het signaal Co', waardoor de flip-flop alleen gedurende Co' in de actiestand, d.w.z. (CCTR) = 1, wordt gehouden. Aangezien het plaatje Oct' beïnvloed moet worden door (CCTR) is op het tweede niveau van dit plaatje het signaal CCTR' aangesloten. Bovendien is op de tweede ingang van dat niveau het kloksignaal Co' aangesloten, zodat signaal Oct' ook werkelijk alleen gedurende die 2 μ .sec. laag wordt, ongeacht de potentiaal, die het eerste niveau van plaatje Oct' heeft op dat moment. Vervolgens is het signaal CCTR aangesloten op de niveau's van de plaatjes waarmee de overige even cijfertijden worden gevormd, zodat al die cijfertijden worden "tegengehouden". Bij dit tegenhouden is de pulslengte van het signaal CCTR niet belangrijk. Nu moet de teller (T) nog worden teruggesteld, ongeacht de inhoud, die hij op dat moment heeft, of m.a.w. de teller moet een Oct-sig-naal produceren, ongeacht de cijfertijd, die op dat moment door zijn inhoud wordt bepaald. Men lette er op dat in dit geval het signaal Oct in eerste instantie door de werking van het signaal CCTR' ontstaat en niet door de tellerstand. Daardoor is het in dit geval mogelijk de teller te beïnvloeden met het signaal Oct, terwijl in "normale" gevallen Oct ontstaat door de inhoud van de teller. Het laatstgenoemde signaal moet worden toegevoerd aan de houdniveau's van de plaatjes To', T1' en T2'. Op plaatje T2 was het echter al aangesloten om andere redenen (zie het voorgaande), zodat dit niet nogmaals behoeft te gebeuren. Het signaal Oct maakt dus van de drie accentplaatjes van de teller het houdniveau hoog, waardoor deze wordt teruggesteld. Een overzicht toont het ontstaan en de actie van CCTR:



In de X-1 komen verschillende tellers voor, o.a. de "gewone" cijfertijdring (CTR) en de geheugen cijfertijdring (GCTR). Ze werken allen volgens het besproken principe (T).

Geheugencijfertijden

De hierboven besproken cijfertijden dienen in hoofdzaak voor het arithmetisch orgaan. Van de "geheugencijfertijdring" (GCTR) worden de cijfertijden afgeleid om het geheugen te drijven. De grenzen tussen de taak van de geheugencijfertijden en die van de "gewone" cijfertijden zijn niet strak te trekken. Ook hier duurt de cyclus van acht cijfertijden weer $16\mu\text{sec}$. Er is slechts één belangrijk verschil met de hiervoor besproken cijfertijden. De GCTR mag niet steeds door blijven tellen, alleen afhankelijk van het kloksignaal, maar mag slechts met tellen beginnen onder controle van de geheugencontroleling (GCR). Uitgaande van het moment, dat de teller een Ogct vormt, zal de volgende stap zijn het vormen van 1gct. Daartoe zouden op het plaatje 1gct' de signalen GCTR1, GCTR2 en C1' laag moeten zijn, maar dit plaatje heeft vier niveau's met elk zes ingangen.

	<u>1gct'</u>					
1	GCTR1	GCTR2	C1'	GCRo'	GCR1	GCR2
2	GCTR1	GCTR2	C1'	GCRo	GCR1'	GCR2
3	GCTR1	GCTR2	C1'	GCRo	GCR1	GCR2'
4	GCTR1	GCTR2	C1'	GCRo'	GCR1'	GCR2'

Op de eerste drie ingangen van al deze niveau's zijn steeds de genoemde signalen aangesloten, terwijl de geheugencontroleling via de overige ingangen de geheugencijfertijdring controleert. Deze teller mag slechts beginnen te tellen respectievelijk bij de vier oneven standen van de GCR, zoals die zijn aangegeven op tek. X4. Op de overige ingangen van de vier niveau's van plaatje 1gct' zijn dan ook zodanige GCR-signalen aangesloten, dat in elk van de vier oneven standen van deze ring één niveau laag wordt (als tenminste tevens de eerste drie ingangen laag zijn). De GCTR kan dan normaal tellen op dezelfde wijze als besproken is bij de "gewone" CTR. Het gevolg van dit alles is, dat de GCTR in de nulstand blijft staan gedurende de tijd, dat de GCR in een even stand is. Dan zal het niveau van plaatje Ogct' laag worden, iedere keer als kloksignaal Co' laag is, zodat herhaaldelijk een signaal Ogct optreedt.

Elke handeling van het geheugen duurt steeds acht cijfertijden. Het moet daarom mogelijk zijn de geheugencijfertijd na iedere cyclus te laten wachten. Daartoe wordt het signaal $7gct'$ toegevoerd aan de tweede niveau's van de plaatjes GCR2' en GCR2. Deze niveau's verzorgen de overgangen van stand 3 naar 4 en van stand 7 naar 0. Ook is het signaal $7gct'$ aangesloten op het tweede niveau van plaatje GCR1' en het derde van plaatje GCR. Deze niveau's veroorzaken de overgangen van stand 1 naar 2 en van stand 5 naar 6. Gedurende $7gct$ wordt de GCR dus in een even stand gebracht, waardoor de teller (GCTR) stopt. De overgang van de even naar de oneven standen gebeurt slechts onder uitwendige controle. Door de werking van het tweede niveau van plaatje GCR1 kan de geheugencontrolling worden teruggesteld uit de derde stand in de nulstand.

Is de geheugencontrolling door uitwendige oorzaak al in de volgende oneven stand gebracht voor de telcyclus zou stoppen, dan heeft het feit, dat de GCR in een even stand verkeerde, geen invloed op de teller; deze loopt dan gewoon door.

De uitbreiding van de cijfertijden

Aangezien we in sommige gevallen niet genoeg hebben aan acht cijfertijden, kan dit aantal nog effectief worden uitgebreid door combinatie met signalen van een teller, die uit de flip-flops a en b bestaat. Deze teller werkt op dezelfde wijze als beschreven is op blz. 23. Flip-flop a gaat steeds om, wanneer signaal Oct' laag gaat en blijft dus gedurende een cyclus van acht cijfertijden in die ene stand staan en gedurende de volgende cyclus in de andere stand.

a' / a			b' / b		
1 Houd	a'		1 Houd	b'	
2 Lees	b	Oct'	2 Lees	a'	$4ct'$
<hr/>					
1 Houd	a		1 Houd	b	
2 Lees	b'	Oct'	2 Lees	a	$4ct'$
			3 Clear	$CCTR'$	Co'

Hetzelfde geldt voor flip-flop b met dien verstande, dat deze flip-flop steeds omgaat, wanneer signaal $4ct'$ laag wordt, Als signaal a' laag is bij de ene cyclus cijfertijden, dan is signaal a laag bij de volgende, enz.

Wanneer nu een niveau laag moet zijn om de $32\mu\text{sec.}$ afgezien van verdere signalen aangesloten op dat niveau, dan sluiten we b.v. signaal $3ct'$ en signaal a' aan elk op een ingang. Nu zal de ene keer dat $3ct'$ laag gaat a' ook laag zijn en de volgende keer niet. Zodoende kunnen we nu 16 achtereenvolgende cijfertijden van elkaar onderscheiden.

Deze teller kan zodanig worden beïnvloed door signaal $CCTR'$, dat signaal a' laag gaat bij Oct en b' hoog. Dit is een aangenomen nulstand van deze teller. Om dit te bereiken, wordt het derde niveau van plaatje b laag door de signalen $CCTR'$ en Co' . Was signaal b al laag, dan zou het leesniveau van plaatje a' laag worden op de eerstvolgende Oct . Was signaal b hoog, dan wordt het laag ten gevolge van de signalen $CCTR'$ en Co' , waarop het signaal a' evenals hiervoor laag wordt op de eerstvolgende Oct .

Het coderend circuit opteller-in (OI)

Zoals reeds bij de bespreking van het blokschema is opgemerkt, bepaalt dit coderend circuit uit welk register de logische opteller (LO) een woord selecteert. Er kan worden gekozen uit vier registers, nl. het A-register (27 bits), het S-register (27 bits), het B-register (16 bits) en de opdrachtteller (15 bits). De laatste twee registers worden in het minst significante gedeelte van OI overgenomen.

Om te bepalen welk register overgenomen moet worden, maakt men gebruik van een zgn. "multiplet". Dit multiplet bestaat in eerste instantie uit vier plaatjes nl. lees-A (LA'), lees-S (LS'), lees-B (LB') en lees-opdrachtteller (LOT'). Het uitgangssignaal van het ene plaatje (het on-accent van de genoemde signalen dus) wordt toegevoerd aan de houdniveau's van de andere drie plaatjes. Een multiplet kan worden beschouwd als een meerzijdige flip-flop. Wordt van een plaatje b.v. LA' één niveau laag gemaakt omdat (A)

moet worden overgenomen (X' en Z laag), dan wordt signaal LA hoog en maakt van de overige drie accentplaatjes de houdniveau's hoog. De signalen LS , LB en LOT worden dus laag, zodat het houdniveau LA' laag wordt. Het multiplet blijft dan in deze stand

		LA'/LA			
1	Houd	LS	LB	LOT	staan tot een niveau van de andere
2		X'	Z		accentplaatjes laag gaat, b.v. van
<hr/>					
1		LA			plaatje LS' . Nu is het signaal LS hoog
					en maakt het houdniveau van plaatje LA'
					weer hoog. De conclusie uit dit betoog

is, dat slechts één van de vier signalen laag kan zijn. Deze plaatjes geven onaccent signalen af, maar ook de accentsignalen moeten beschikbaar komen. Om deze omkering te krijgen is voor elk signaal nog een plaatje nodig. Met behulp van de aldus verkregen signalen kan het circuit OI zijn functie uitoefenen.

Het circuit OI bestaat uit 27 plaatjes, dat is evenveel als het aantal bits, dat de grootste registers bevatten. De registers B en OT bevatten minder cijfers. Daarom worden, wanneer de opdracht teller moet worden geselecteerd, op de plaatjes aan de meest significante zijde van OI , voor welke dit register geen cijfers heeft, de betreffende signalen niet aangesloten. Omdat het getal in het B -register ook negatief moet kunnen zijn, is hiervoor een speciale voorziening getroffen, die later zal worden behandeld. Overigens zijn alle plaatjes per bit analoog. Eén plaatje zal als voorbeeld worden besproken.

		OI7'/OI7			
1	Lees A	$A7'$	LA'		Wanneer A geselecteerd moet worden,
2	Lees S	$S7'$	LS'		zal een niveau van plaatje LA' laag
3	Lees OT	$OT7'$	LOT'		worden en daardoor signaal LA' eveneens.
4	Lees B	$B7'$	LB'		Dit signaal is aangesloten op het eerste
<hr/>					
1		OI7			niveau van plaatje $OI7'$. Indien sig-
					naal $A7'$ laag was, d.w.z. $(A7) = 1$,
					dan wordt signaal $OI7'$ ook laag. Was
					signaal $A7'$ hoog, d.w.z. $(A7) = 0$, dan is ook $OI7'$ hoog. Immers
					alle overige niveau's van plaatje $OI7'$ zijn hoog door de overige
					signalen van het multiplet. Bij het overnemen uit de twee kleine
					registers worden de meest significante signalen door het multiplet
					hoog (0) gemaakt behalve als (B) negatief is, doordat er alleen
					de signalen van het multiplet op zijn aangesloten, die betrekking
					hebben op niet geselecteerde registers.

Het coderend circuit logische opteller (L0)

Op tekening X4 is een tabel getekend, genaamd "opteller". De eerste kolom in deze tabel noemt de standen van de opteller. Men kan deze standen verdelen in groepjes van twee, waarbij steeds dezelfde handeling wordt verricht, echter de ene keer met het getal positief en de andere maal met het getal negatief. In dit systeem zou stand 4 hetzelfde moeten doen als stand 0 en kan daarom achterwege worden gelaten.

De R in de tweede kolom is een algemene aanduiding voor één van de vier interne registers, waarbij in het midden wordt gelaten, welk register door 0I is gekozen. De overige kolommen geven de standen van de daarbij genoemde flip-flops weer, zoals die moeten zijn om de opteller de juiste handeling te laten verrichten. De plaatjes "controle opteller in" (COI) zorgen ervoor dat deze flip-flops in de juiste stand gezet worden. Er zijn 13 standen en daarom moeten er ook 13 plaatjes COI' zijn (Stand 4 van de tabel wordt niet gebruikt). De structuur van de signalen COI wordt later besproken. De inhoud van flip-flop COA moet één worden in de standen 1 - 3 - 5 - 7 - 9 - 13 van de tabel, m.a.w. de COA moet in de actiestand worden gezet. Dan is signaal COA hoog. Op het houdniveau van plaatje COA zijn de signalen COI1, COI 3, COI5, COI7, COI9 en COI13 aangesloten. Op deze wijze worden al deze flip-flops in de juiste stand gezet, waarbij plaatje CCC' geen flip-flop is. Het signaal CCC wordt hoog, wanneer de flip-flops CCA en CCB tegelijkertijd een nul bevatten.

Toen het ontwerp van de X-1 klaar was en men dus alle mogelijkheden kon overzien, kon men bepalen, welke wijzigingen in de stand der flip-flops nimmer voor zouden komen in enige opdracht. Als gevolg hiervan behoeft dan die wijziging ook niet teweeg gebracht te worden, zodat het betreffende COI-signaal dan niet nodig is. Dit spaart dus per keer een diode uit.

Beschouwen we de tijdsdiagrammen, dan blijkt daaruit, dat b.v. stand 2 nooit direct volgt op enige actiestand van flip-flop COA. Signaal COI2 behoeft dus nimmer flip-flop COA terug te stellen en is daarom niet aangesloten op het houdniveau van plaatje COA'.

Bekijken we de signalen van de logische opteller, b.v. op plaatje LO7', dan blijkt dat deze de logische som geeft van een cijfer uit

LO7'/LO7			
1	OI7	M7	COA'
2	OI7'	M7'	COB'
3	OI7	M7'	COC'
4	OI7'	M7	COD'
<hr/>			
1	LO7		

het M-register en het overeenkomstige cijfer uit OI.

Aan de hand van de tabel "opteller" (tek. X4) zal nu worden verklaard, dat de LO inderdaad de daargenoemde handelingen verricht. We zullen alleen de gevallen beschouwen, waarin het signaal LO7' laag wordt, omdat in alle

andere gevallen het signaal een nul vertegenwoordigt.

Stand 0

Nu zijn de signalen COA' en COB' hoog, terwijl de signalen COC' en COD' laag zijn (zie tabel). De eerste twee niveau's van plaatje LO7' zijn dus hoog en hebben geen invloed.

Van de niveau's 3 en 4 zijn de signalen COC' en COD' laag. Deze niveau's kunnen dus laag worden. Bij de logische optelling zijn er vier mogelijkheden:

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 0 &= 1 \\ 1 + 1 &= 0 \end{aligned}$$

In twee gevallen moet het signaal LO7' laag worden, dus een één vertegenwoordigen, nl. wanneer de op te tellen cijfers ongelijk zijn. Dit gebeurt inderdaad op deze niveau's, want op het 3e niveau is het signaal OI7 (laag bij 0) en het signaal M7' (laag bij 1) aangesloten. Op het 4e niveau zijn de signalen OI7' (laag bij 1) en M7 (laag bij 0) aangesloten. Hiermee is het gestelde doel bereikt.

Stand 1

In dit geval zijn de signalen COA' en COB' laag en de signalen COC' en COD' hoog. Nu hebben het 3e en 4e niveau dus geen invloed en kunnen de niveau's 1 en 2 wel laag gaan. De inhoud van M moet worden afgetrokken van het getal uit R.

Volgens definitie geldt het volgende: het + teken wordt voorgesteld door een nul en het -teken door een één. Deze cijfers worden voor het getal geplaatst. Bovendien worden bij een negatief getal alle cijfers "omgekeerd", d.w.z. overal wordt 0 door 1 en 1 door 0 vervangen.

Het aldus verkregen getal kan "gewoon" worden opgeteld bij het getal uit R, hetgeen dus inhoudt, dat het oorspronkelijke getal wordt afgetrokken van dat getal uit R. Een voor de hand liggende gedachte is dan ook om het getal uit M "om te keren" en daarna aan L0 toe te voeren. Dit kost echter veel meer tijd (een extra omkering) en meer materiaal (een eenheid T1 per bit). Een eenvoudiger oplossing is om het "verkeerde" uitgangssignaal van de flip-flop te benutten. Immers als de inhoud van een flip-flop nul is, dan is het accentsignaal hoog en het onaccentsignaal laag. Nu moet het signaal L07' laag worden, d.w.z. een één vertegenwoordigen, door een uitgangssignaal van een flip-flop met inhoud nul. Daarvoor gebruiken we dus het onaccentsignaal.

Ook bij deze stand zijn er twee gevallen, waarin het signaal L07' een één moet vertegenwoordigen. Hiertoe is op het 1e niveau aangesloten het signaal OI7 (laag bij 0) en het signaal M7 (laag bij 1, in verband met de omkering!) en op het 2e niveau signaal OI7' (laag bij 1) en het signaal M7' (laag bij 0!). Ook voor deze stand is dus het gestelde doel bereikt.

Stand 2

Het signaal L07' moet laag worden, wanneer $(M) = 1$, ongeacht de polariteit van de signalen OI7' en OI7. Volgens de tabel van tek. X4 zijn de signalen COB' en COC' laag. Indien signaal OI7' laag is, wordt L07' laag door M7' (2e niveau). En als OI7 laag is, heeft M7' hetzelfde gevolg (3e niveau). Weer is de gewenste handeling verricht.

Stand 3

De logische opteller moet nu de inhoud van het M-register negatief overnemen, dus L07' moet laag worden, wanneer signaal M7' hoog is en M7 laag. Volgens de tabel zijn de signalen COA' en COD' laag, m.a.w. alleen de niveau's 1 en 4 kunnen laag worden. De inhoud van M7 wordt overgenomen, maar nu omgekeerd, ongeacht de polariteit van de signalen OI7' en OI7.

Stand 4

Deze wordt niet gebruikt, omdat nu in het systeem van de tabel hetzelfde gedaan zou moeten worden, als bij stand 0.

Stand 5

Voor de logische opteller is dit hetzelfde geval als beschreven bij stand 1, alleen de flip-flops CCA en CCB hebben hier een andere stand. De gevolgen daarvan zullen we bij de behandeling van de carryvormer zien.

Stand 6

De inhoud van R moet in de logische opteller worden gezet. De signalen COB' en COD' zijn laag. Het signaal LO7' wordt laag tengevolge van OI7', ongeacht welk signaal laag is: M7' of M7.

Stand 7

Nu moet de inhoud van R negatief in de logische opteller worden gezet. Het signaal LO7' wordt laag tengevolge van signaal OI7 (onaccent!), ongeacht de polariteit van M7' en M7.

Stand 8

In deze stand wordt een logische optelling uitgevoerd. Hierbij wordt geen overdracht of "carry" gevormd. De flip-flops "Controle opteller" staan in dezelfde stand als bij stand 0, terwijl de flip-flops "Controle Carry" nu niet in de actie-stand staan

Stand 9

Ook hier wordt een logische optelling uitgevoerd, maar nu met het getal uit het M-register negatief. Vergelijk de standen van de flip-flops in dit geval met die van stand 1.

Stand 10

Onder controle van de flip-flops volgens deze stand wordt een logische vermenigvuldiging uitgevoerd. Ook hierbij wordt geen carry gevormd. Bij een logische vermenigvuldiging zijn er vier mogelijkheden:

$$\begin{aligned} 0 \times 0 &= 0 \\ 0 \times 1 &= 0 \\ 1 \times 0 &= 0 \\ 1 \times 1 &= 1. \end{aligned}$$

Er is dus slechts één geval, waarin signaal LO7' laag moet worden, nl. wanneer OI7' en M7' laag zijn. Volgens de tabel op tek. X4 is nu signaal COB' laag, zodat alleen het 2e niveau kan werken en hierop zijn inderdaad de signalen OI7' en M7' aangesloten.

Stand 11

Nu wordt eveneens een logische vermenigvuldiging uitgevoerd, maar de inhoud van het M-register wordt negatief beschouwd. Signaal LO7' moet nu laag worden, wanneer de signalen OI7' en M7 (onaccent!) laag zijn. Dit is het geval met het 4e niveau, hetgeen overeenkomt met het gegeven uit de tabel op tek. X4.

Standen 12 en 13

Deze standen zullen later worden verklaard.

Het coderend circuit carryvormer(CV)

De logische optelling van twee getallen geschiedt op onderstaande wijze:

6	5	4	3	2	1	0	bitnummer
0	1	1	0	0	1	0	(OI)
0	0	1	1	1	1	1	(M)
							+
0	1	0	1	1	0	1	(LO)

Bij deze optelling worden de carry's (overdrachten) niet doorberekend, maar verwaarloosd. Vandaar dat we in de tot nu toe behandelde schakelingen van LO, de carry slechts zijdelings behoeften te noemen.

Een "gewone" optelling verloopt als volgt:

6	5	4	3	2	1	0	bitnummer
1	1	1	1	1	0	0	Carry's
0	1	1	0	0	1	0	(OI)
0	0	1	1	1	1	1	(M)
							+
1	0	1	0	0	0	1	(SV)

Een carry moet worden gemaakt en de carryvormer moet dus functioneren bij een "gewone" optelling. Uit het bovenstaande optelling kan worden afgeleid, dat in vier gevallen een carry moet worden gevormd, nl.:

OI	M	LO	CV	
			van voor-	
			gaand bit	
1	1	0	0	bit 1
0	1	1	1	bit 2
1	1	0	1	bit 4
1	0	1	1	bit 5

Deze vier gevallen zijn terug te brengen tot twee voorwaarden, nl.:

- 1e. Wanneer het signaal $OI7'$ een één vertegenwoordigt en het signaal $LO7'$ een nul, want dit houdt in, dat $(M7) = 1$. Dit geval is verwezenlijkt op het 2e niveau van plaatje $CV7'$.
- 2e. Wanneer het signaal $LO7'$ een één vertegenwoordigt en het signaal $CV6'$ eveneens, want dit houdt in, dat of $(M7) = 1$ of $(OI7) = 1$.

CV7'/CV7				Dit is verwezenlijkt op het 1e niveau.
1	<u>CV6'</u>	$LO7'$		Met de signalen "Controle Carry"
2	$OI7'$	$LO7$	CCA'	(CCA' - CCB' - CCC) wordt de carryvormer
3	$OI7$	$LO7$	CCB'	bediend, volgens de standen, die zijn
<hr/>				aangegeven in de tabel op tek. X4.
1	$CV7$			Het 1e niveau kan steeds werken onaf-
				hankelijk van de controle carry.

Het 2e niveau werkt alleen wanneer flip-flop CCA in de actie-stand staat, d.w.z. CCA' is laag. Als signaal CCB' laag is, kan het 3e niveau werken. Zoals bekend is (zie blz. 36), kan een getal gewoon worden opgeteld, als het negatief is, wanneer men maar alle cijfers "omkeert". M.a.w. als men de accent- en de onaccentsignalen vervangt door resp. de onaccent- en accentsignalen. In tegenstelling tot het 2e niveau wordt het 3e niveau laag, wanneer signaal $OI7$ (onaccent) laag is. Signaal $OI7$ stelt nu -1 voor.

Bij optelling van b.v. een getal, dat bestaat uit allemaal énen en een getal, waarvan alleen het minst significante cijfer een één is, moeten achtereenvolgens alle plaatjes van de carryvormer een carry doorgeven. Een analoog voorbeeld vindt men bij de optelling van het 1e, 2e en 3e bit in de besproken optelling. Dit kost een groot aantal omkeringen, dus veel tijd.

Daarom heeft men de gebruikelijke schakeling iets gewijzigd met behulp van de uitwendige bedrading (zie fig. 19). Het contact,

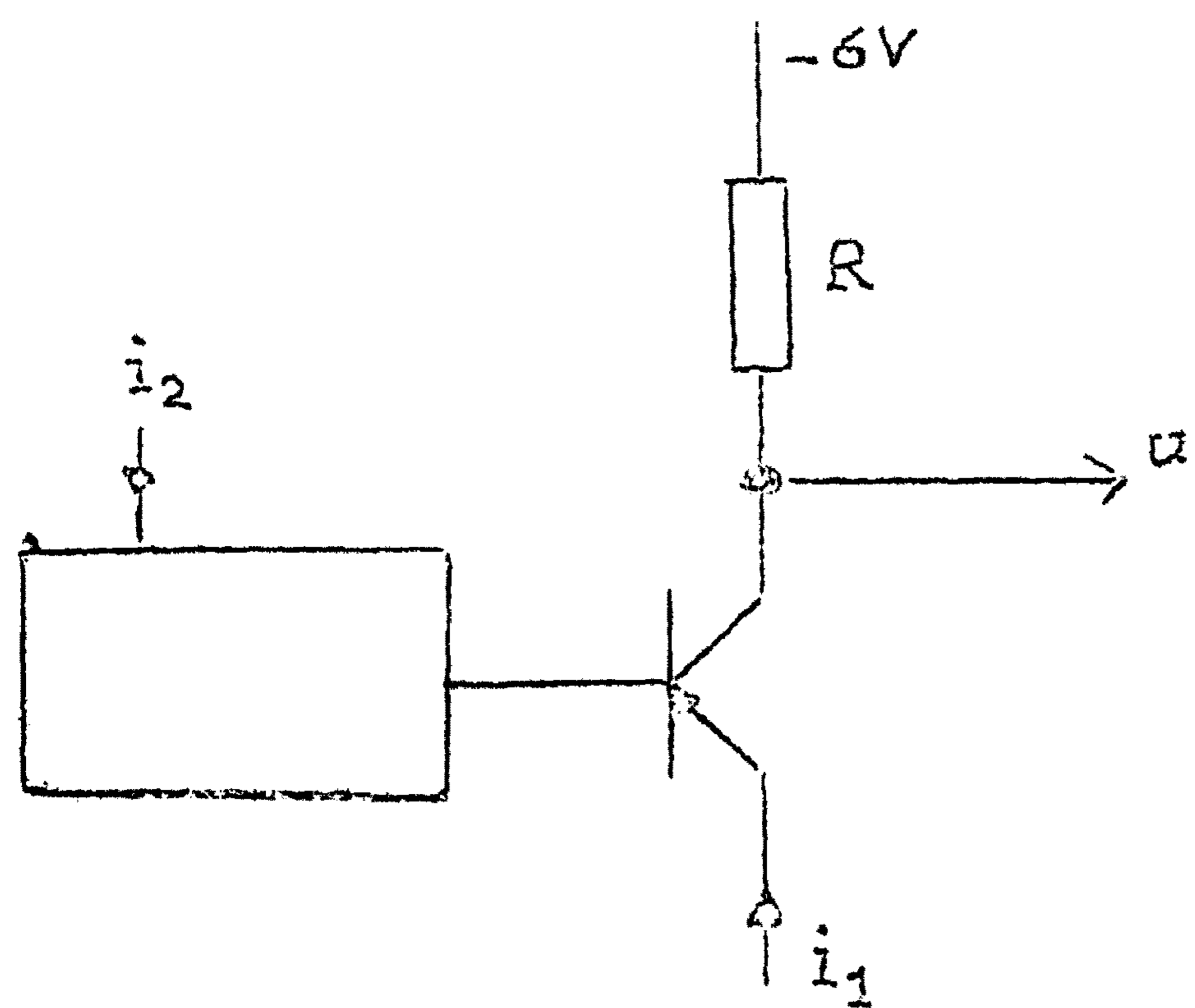


fig. 19

waarop de emitter is aangesloten, wordt niet, zoals bij voorgaande gevallen, met de aarde verbonden (zie blz. 4), maar als ingangcontact gebezigd. Wanneer op dit contact (i_1) een signaal wordt aangesloten, dat hoog (0V) is, dan geeft dit de transistor de indruk of zijn emitter "normaal" aan aarde ligt. De gedragingen van deze schakeling zijn dan

ook daarmee in overeenstemming. Signaal u wordt laag, als signaal i_2 hoog wordt en omgekeerd. Wanneer signaal i_1 laag is (-6V), dan ligt zowel de collector als de emitter aan -6V, zodat er geen stroom loopt. Bovendien ligt de basispotentialiaal nu immers altijd aanzienlijk boven die van de emitter. Signaal u is dan ook laag (-6V). De signaalinjectie op de emitter heeft hetzelfde gevolg, als wanneer het signaal i_1 invers was aangesloten op een "normale" ingang. Het voordeel is dat nu "de omkering" sneller plaats vindt, hoewel in feite het signaal niet wordt omgekeerd. Opgemerkt dient nog te worden, dat de emitter in het eerste geval niet direct aan aarde ligt, maar via de (te verwaarlozen) weerstand van één geleidende transistor.

Waren er meer geleidende transistoren in de emitterleiding van de eerste, dan werd daardoor de emitterweerstand vergroot. Dit heeft dan tot gevolg, dat de stroomtegenkoppeling groter wordt. Ook stuurt de basisspanning de transistor dan moeilijker uit, doordat de emitterspanning te groot wordt. Eveneens in verband hiermee, mag de belasting van de transistoren niet te groot worden. Elke transistor moet uiteraard stroom sturen door een aantal ingangen. De transistor, welke met zijn emitter werkelijk aan aarde ligt, heeft ten eerste zijn eigen belasting, maar bovendien de stroomdoorgang van elke transistor met de daarbij behorende belasting. Daarom wordt steeds slechts één transistor op genoemde wijze aan een andere gekoppeld.

Voor een duidelijk overzicht en een goed begrip wordt in het bedradingsboek een speciale notatie gebezigd. Bij de ingangsignalen is het signaal genoteerd, dat schijnbaar op een ingang is aangesloten, met vermelding van het juiste contactnummer (aan de emitter). Bovendien is het gemerkt met een streeplijntje om de aandacht erop te vestigen, dat het geen "gewone" ingang is. Ten overvloede wordt er de aandacht op gevestigd, dat het aldus gemerkte signaal, alleen invloed heeft op het niveau, waarbij het vermeld staat.

Alle plaatjes van de CV moeten achtereenvolgens een laag accent-signaal afgeven bij het bovengenoemde geval. Om nu alle plaatjes aan elkaar te koppelen met een minimum aan omkeringen, worden de niveau's twee aan twee op de boven beschreven wijze gekoppeld, terwijl de overige koppelingen op "gewone" wijze geschieden. Hieruit blijkt, dat dit systeem alleen is door te voeren, als het minst significante plaatje (CV0) een accentsignaal afgeeft, het tweede plaatje eveneens, terwijl het derde plaatje (CV2') een onaccent signaal af moet geven, evenals het vierde plaatje, enz. Om de beschikking te hebben over het accentsignaal, zowel als het onaccentsignaal moet dus van elk plaatje het uitgangssignaal nog eenmaal worden omgekeerd.

Bij de uitcodering van de signalen op de onaccentplaatjes moeten andere signalen worden aangesloten, als bij uitcodering op de accentplaatjes, omdat de niveau's van deze plaatjes de gevallen specificeren, waarin geen carry gevormd wordt of wel (CV) = 0

	CV8 /CV8'	OI	M	LO	CV van voor- gaand bit
1 CV7	LO8'				
2 OI8	LO8 CCB	0	1	1	0
3 OI8'	LO8 CCA	1	0	1	0
<hr/>					
1 CV8'		0	0	0	0
		0	0	0	1

Ook deze vier gevallen zijn terug te brengen tot twee voorwaarden nl.: 1e. Wanneer signaal OI8 een nul vertegenwoordigt en het signaal LO8 eveneens, want dit houdt in dat (M8) = 0. Dit geval is uitgecodeerd op het 2e niveau van plaatje CV8

2e. Wanneer het signaal CV $\underline{7}$ een nul vertegenwoordigt en het signaal LO8' een één, want dat houdt in, dat of (M8) = 1 of (OI8) = 1. Dit is uitgecodeerd op het 1e niveau.

Ook nu kan het 1e niveau steeds werken onafhankelijk van de controle carry. Het 2e niveau werkt alleen, wanneer geen carry moet worden gevormd (CCB is laag), dat zijn dus ook de gevallen, waarin de carry nul moet zijn. Het 3e niveau kan laag worden, omdat signaal CCA laag is, o.a. in stand 5 van de opteller. Dan wordt het getal in OI negatief beschouwd.

In de gevallen, dat wel een carry moet worden gevormd, moeten de drie niveau's van dit plaatje tegelijkertijd hoog zijn. Hiervan zullen we één voorbeeld geven, de andere gevallen zijn analoog. Bij bit 1 in de tabel op blz. 40 is signaal LO8' hoog en dit maakt niveau 1 van plaatje CV8 hoog, signaal OI8 maakt niveau 2 hoog en signaal CCA niveau 3. Hierdoor is signaal CV8' laag en vertegenwoordigt een één. Om deze schakelwijze te verduidelijken is in fig. 19a op blz. 42b een voorbeeld hiervan gegeven.

Alleen in de standen 0 - 1 - 5 - 12 - 13 van de opteller (tabel op tek. X4) moeten carry's worden gevormd, in alle overige standen niet. In het laatste geval staan de flip-flops CCA en CCB beide in de nulstand en zijn de signalen CCA en CCB laag. Deze signalen zijn de enige signalen, die op het niveau van plaatje CCC' zijn aangesloten. Het signaal CCC is dus hoog, wanneer er geen carry mag worden gevormd. Het is aangesloten op de eerste niveau's van de plaatjes CV2', CV6', CV10', CV14', CV18', CV22' en CV26'.

Het is mogelijk, dat van alle CV-plaatjes het 1e niveau laag is en snel hoog moet worden, omdat bij de eerstvolgende handeling geen carry behoeft te worden gevormd. Dan zou dit 26 omkeringen kosten. Met signaal CCC wordt dit aantal omkeringen beperkt. Plaatje CVo kan het carrysignaal doorgeven aan plaatje CV1. Maar plaatje CV1 kan het niet doorgeven aan plaatje CV2', want het niveau, waarop dit signaal binnenkomt, wordt hoog gehouden door

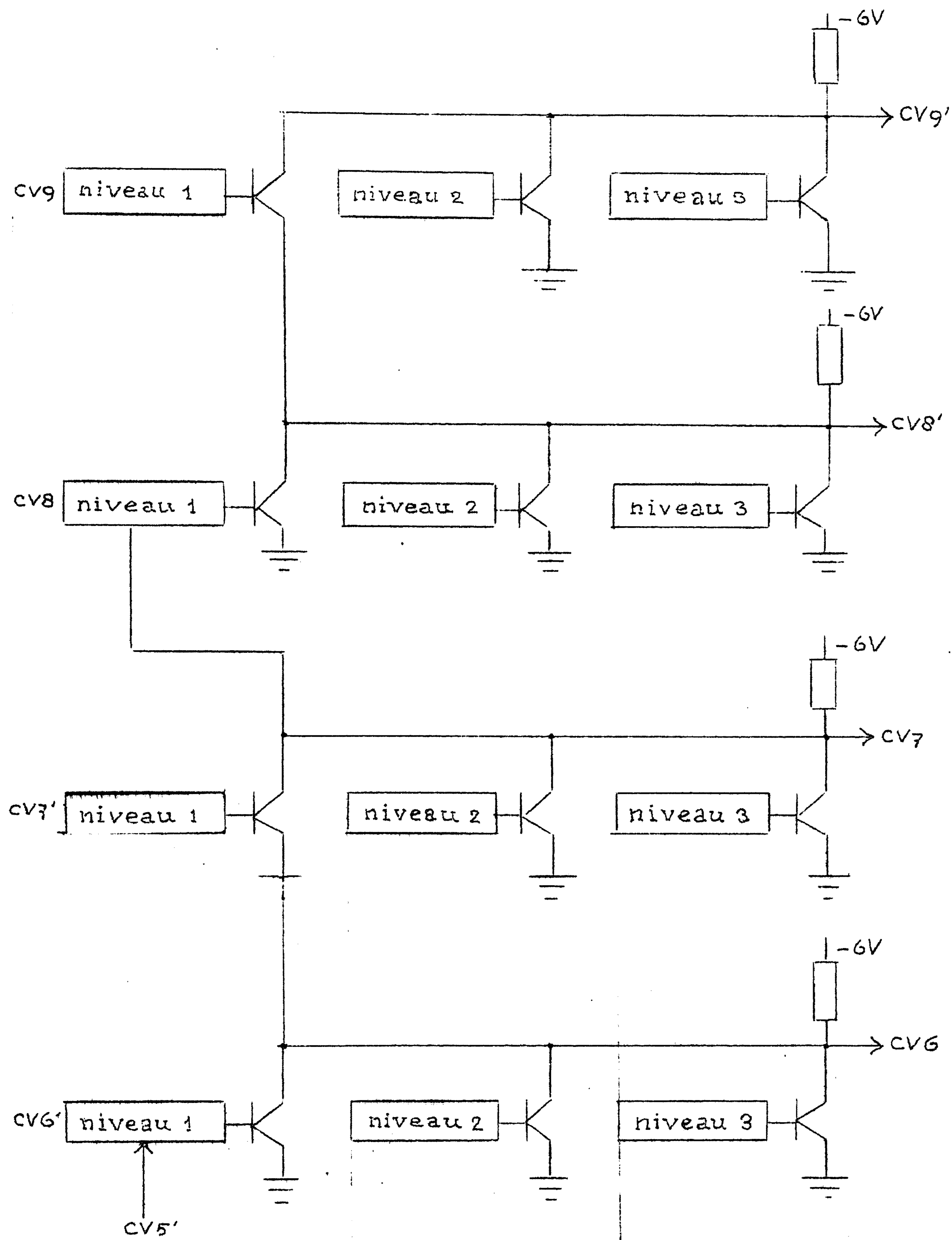


fig. 19a

het signaal CCC. Plaatje CV3' kan het carry-sigitaal, dat daarin is gevormd, doorgeven aan plaatje CV4, dit aan plaatje CV5, terwijl dit laatste plaatje geen invloed kan uitoefenen op plaatje CV6', omdat hiervan het niveau weer hoog wordt gehouden door signaal CCC. Om de vier plaatjes is het signaal CCC aangesloten op een niveau, zodat de carry na drie omkeringen is "doodgelopen".

Het coderend circuit somvormer (SV)

Van de somvormer bestaan alleen de onaccentplaatjes, aangezien slechts de accentsignalen beschikbaar behoeven te zijn.

De somvormer moet het resultaat geven van de gehele optelling, dus van het getal uit OI en het getal uit M met de carry's. Zoals is besproken in het voorgaande, wordt de logische som van de eerste twee getallen gegeven door LO. Wanneer nu daarbij de carry's worden geteld, dan ontstaat het resultaat van de werkelijke optelling. De laatstgenoemde signalen zijn aangesloten op de plaatjes van de somvormer.

Bij beschouwing van het voorbeeld van een optelling (blz. 39) blijkt, dat er vier mogelijkheden zijn voor de somvormer, nl.

LO	CV van voor- gaand bit	SV	
0	0	0	bit 1
1	1	0	bit 2
1	0	1	bit 0
0	1	1	bit 4 en 6 .

SV7	Wanneer beide signalen, die op de ingangen van één niveau zijn aangesloten, gelijke polariteit hebben, wordt een niveau van plaatje SV7 laag en daardoor het signaal SV7' hoog, d.w.z. vertegenwoordigt een nul. Dit is tweemaal het geval, nl.
1 L07 CV6	
2 L07' CV6'	

1e. wanneer de LO een nul geeft en de carry van het voorgaande bit nul is,

2e. wanneer beide een één geven.

In twee andere gevallen moet de somvormer een één geven, nl. wanneer beide signalen ongelijke polariteit hebben. Dan is signaal SV7' laag.

Het uitregister

Het woord, dat in de somvormer is ontstaan moet door het U-register worden overgenomen. Voordat dit echter gebeurt, moeten alle flip-flops van het U-register worden teruggesteld. Daartoe maakt

U7'/U7			
1	Houd	U7'	CU
2	Lees	SV7'	UI'
3		SV6'	UIL'
4		SV8'	UIR'
<hr/>			
1	Houd	U7	

het signaal CU het houdniveau van het accentplaatje hoog. Zoals we later zullen zien, kan slechts één der overige niveau's laag worden ten gevolge van signaal CU.

Stel dat signaal UI' laag is, dan wordt niveau 2 van plaatje U7' laag, als signaal SV7' laag is. De flip-flop U7 vertegenwoordigt dan een één.

Wanneer SV7' hoog is, wordt $(U7) = 0$. De werking van de niveau's 3 en 4 wordt later behandeld. De inhoud van het uitregister kan door diverse registers worden overgenomen (zie blokschema).

Het A-register

Het overnemen van het woord uit het U-register in het A-register wordt nu als voorbeeld besproken. Het overnemen in de andere registers geschiedt op analoge wijze. De voorwaarden, waarop het A-register de inhoud van het U-register moet overnemen, zijn uitgedoceerd op een aantal niveau's van plaatje AI'. Als één van deze niveau's laag wordt, gaat signaal AI hoog. Dit signaal maakt het houdniveau A7' hoog en stelt de flip-flop A7 terug. Tevens

A7'/A7			
1	Houd	A7'	AI
2	Lees	U7'	AI'
<hr/>			
1	Houd	A7	

wordt het signaal AI omgekeerd en daardoor ontstaat signaal AI'. Dit laatste signaal scheidt de mogelijkheid voor A om de inhoud van U over te nemen. Op een zeker moment wordt signaal AI weer laag, zodat het houdniveau van plaatje A7' zijn houdende

functie weer gaat uitoefenen. Eén omkering later wordt het leesniveau van dit plaatje hoog t.g.v. signaal AI', maar dat heeft al geen invloed meer.

Handbediening

Op een verticaal paneel van de X-1 zijn o.a. enkele groepen schakelaars aangebracht. Er zijn drie groepen van tweestanden-schakelaars, waarmee een binair woord kan worden gevormd. In de eerste plaats is er de groep "getalschakelaars", hiermee kan een woord van 27 bits worden gevormd. De getalschakelaars beïnvloeden direct het dood geheugen. De X-1 kan dit woord interpreteren als opdracht of als getal. In de tweede plaats bevinden zich hier de "beginadressschakelaars". Zoals de naam al zegt, kan men met deze schakelaars een adres specificeren. Een adres bestaat uit 15 bits en deze groep uit 15 schakelaars. Dan is er nog de groep "stop-adressschakelaars". Ook hiermede kan een adres worden gespecificeerd.

In het schrijfblad van de X-1 console is een paneeltje met drukknoppen gemonteerd. Vier daarvan zijn de zgn. "operationele knopjes" (DOE, HH, BVA en BGA). Verder zijn er nog zestien "decimale knopjes". Tien voor de decimale cijfertekens (DTo t/m 9), één voor het plusteken (DT10), één voor het minteken (DT11) en één voor de punt (of komma) bij breuken (DT12). De drie resterende knopjes dienen voor doeleinden, welke later zullen worden besproken.

Op tek. XE 30 zijn de schakelaars (in ruststand) getekend met de bijbehorende bedrading. De signalen, die op deze tekening zijn vermeld, hebben weer twee polariteiten nl. hoog en laag.

Een signaal is hoog, wanneer dat signaal aardpotentiaal (0V) heeft. Dat is dus hetzelfde, als bij de reeds besproken signalen. Zoals bekend is, kan een ingang van een niveau worden beschouwd als niet aanwezig, wanneer het signaal op die ingang laag is. Nu is ook het omgekeerde het geval, wanneer er geen signaal is aangesloten op een ingang, d.w.z. de spanning op die ingang is "zwevend", dan beschouwt men dit geval, alsof er op die ingang een laag signaal is aangesloten. Dit is hier toegepast.

Bij beschouwing van signaal DKAC1 blijkt, dat dit signaal hoog wordt, wanneer één van de DT-knopjes wordt neergedrukt. Signaal KO wordt laag, als één van de zestien knopjes is neergedrukt. We zullen met deze voorbeelden volstaan, daar de andere signalen eenvoudig zijn af te leiden. Met al deze signalen is een code gevormd door de drukschakelaars te bedraden volgens een bepaald systeem, zodat aan de signalen te zien is, op welk knopje is gedrukt.

Is knopje DT10 neergedrukt, dan worden daardoor de signalen DU2', DI3', DI2', DUo', DI1', DIo' en KO laag. Als knopje DT7 wordt neergedrukt, zijn de signalen DI2', DI1', DIo' en KO laag. Deze signalen worden aan een aantal flip-flops toegevoerd, zodat nu door de flip-flops wordt onthouden op welk knopje was gedrukt. Hiervoor worden gebruikt de flip-flops OKG, DKG, HBo, HB1 en de vier minst significante flip-flops van het A-register.

We zullen nu alle gebeurtenissen nagaan, die volgen op het neerdrukken van een knopje. Dit doen we aan de hand van het tijdschema "handbediening" op tek. XE 12.

In de eerste plaats gaan we de operationele knopjes beschouwen en beginnen met het knopje "DOE".

Knopje "DOE"

We veronderstellen, dat de X-1 om de een of andere reden is gestopt en nu willen we hem alleen de opdracht, die met de getal-schakelaars in het geheugen is gezet, laten uitvoeren. Daarna moet de X-1 weer stoppen. Dan moet op het knopje "DOE" worden gedrukt.

Stel, de arithmetische en de geheugen controle ring staan in de nulstand en het knopje DOE wordt neergedrukt op b.v. 3 cijfertijd. Het eerste gevolg hiervan is het laag worden van de signalen HBU' en KO. Ook worden de signalen DU2', DI3' en DI2' laag, maar dat heeft hier geen invloed, zoals we later zullen begrijpen.

	OKAC ...	Het tweede gevolg van het neerdrukken van
1	OKAC1	de knop is, dat de signalen OKAC1 en OKAC2
2	OKAC2	hoog worden. Deze beide signalen zijn aange-
		sloten op twee niveau's van het plaatje OKAC.

Zoals bekend is, kan nu het uitgangssignaal OKAC' slechts laag worden, indien de signalen OKAC1 en OKAC2 tegelijkertijd hoog zijn. Dit is tevens het bewijs, dat het knopje voldoende is neergedrukt

		OKG'/OKG	en goed contact maakt. Op
1	Houd	OKG' 2ct	het leesniveau van plaatje
2	Lees	Ro' OKAC' 5ct' HBo HB1	OKG' komt op de eerste in-
			gang het signaal Ro'. Dit
1	Houd	OKG	signaal is de uitcodering

van de nulstand van de arithmetische controle ring en is nu laag.

Ook de signalen HBo en HB1 zijn laag (verklaring zie hieronder). Het signaal OKAC' maakt het leesniveau van plaatje OKG' laag op 5 cijfertijd. De flip-flop blijft nu in de eenstand staan tot de eerstvolgende 2 cijfertijd, want dan wordt het houdniveau van plaatje OKG' hoog door signaal 2ct.

Met de signalen van flip-flop OKG worden de flip-flops HBo en HB1 beïnvloed. De eerste bepaalt of de machine na het uitvoeren van de opdracht moet stoppen (nulstand) of moet doorlopen (eenstand). Met flip-flop HB1 wordt bepaald of de opdracht moet worden gehaald op het adres, dat de opdrachtteller aangeeft (nulstand) of uit de getalschakelaars (eenstand).

Het signaal OKG' zet de flip-flop HB1 in de eenstand op 7 cijfertijd. Op het leesniveau van plaatje HB1' zijn de signalen HBU' (laag ten gevolge van knopje DOE) OKG' en 7ct' aangesloten.

	HB1'/HB1				
1	Houd	HB1'	CHB		
5	Lees	HBU'	OKG'	7ct'	
1	Houd	HB1			

Adres 0 in het dood geheugen wordt beïnvloed door de getalschakelaars, en daar is te vinden welk woord met de getalschakelaars in de machine is gebracht. Het coderend

circuit "selecteer geheugen" (SG) maakt het mogelijk de diverse adressen in het geheugen te selecteren. Als SG14 een nul vertegenwoordigt, selecteert het circuit het levend geheugen en als het een één vertegenwoordigt, wordt het dood geheugen geselecteerd. De andere SG-plaatjes (0 t/m 13) bepalen het verdere adres in het geheugen. De niveau's 1 en 2 van de SG-plaatjes zijn hoog ten gevolge van de signalen SOR' en SOT'. Deze signalen worden later besproken. De 14 minst significante plaatjes zijn allemaal nul, zodat het adres 0 is. Van plaatje SG14' wordt het 3e niveau laag, zodat dit het dood geheugen selecteert. Daartoe zijn de signalen HB1' en GCR2 op dit niveau aangesloten. Het laatste signaal is laag gedurende de eerste vier standen van de geheugen controle ring (zie tek. X4).

Flip-flop HB1 kan alleen worden teruggesteld door het signaal CHB. Dit mag alleen gebeuren aan het einde van een opdracht, dus wanneer

	CHB'		de arithmetische controle ring in stand drie staat
1	R3'	STOP'	(R3') en de handeling gestopt moet worden (STOP').
			Uit deze beide signalen ontstaat dan ook het

signaal CHB. Dit plaatje heeft nog een niveau, maar dat zal later worden besproken.

Zolang het signaal HB1 hoog is, kan flip-flop OKG onmogelijk omgaan in de eenstand ten gevolge van enig operationeel knopje. Dit is een beveiliging tegen het ongewenst neerdrukken van knopjes tijdens de werking van de X-1. Het signaal HB1 (hoog) is ook aangesloten op

HBS'	het niveau van plaatje HBS', zodat signaal HBS
1 HBO HB1	laag blijft tot signaal CHB de flip-flop HB1
	terugstelt en de X-1 moet stoppen. Dit alles is

het gevolg van het drukken op knopje "DOE".

Wanneer nu het knopje wordt losgelaten, dus weer in de ruststand komt, dan veranderen de signalen OKAC1, OKAC2, HBU' en KO van potentiaal. Alleen het laatstgenoemde signaal veroorzaakt dan actie. Het wordt omgekeerd en daarna als signaal KO' toegevoerd aan het 5e niveau van plaatje GCRo'. De beide signalen HBS en KO'

	GCRo'/GCRo	op dit niveau garanderen, dat
1 Houd GCRo'		de X-1 pas gaat werken, als
5 0-1 ACRO' 6ct' HBS STOP' KO'		alle flip-flops in de juiste
		stand staan (HBS) en het
1 Houd GCRo		knopje is losgelaten (KO').

De signalen ACRO, HBS en

STOP' zijn reeds laag, zodat op de eerstvolgende 6 cijfertijd de geheugen controle ring door signaal KO' in de eenstand wordt gezet. Dit heeft tot gevolg, dat de geheugencijfertijdtring gaat tellen.

Op 2gct maakt signaal GCRo' het 1e niveau van plaatje CM' laag, zodat de flip-flops van het M-register worden teruggesteld en dus een woord uit het geheugen kunnen overnemen. Op 3gct wordt door signaal GCRo' niveau 8 van plaatje CCTR' laag. Hierdoor wordt de "gewone" cijfertijdtring teruggesteld met de bekende gevolgen. Op

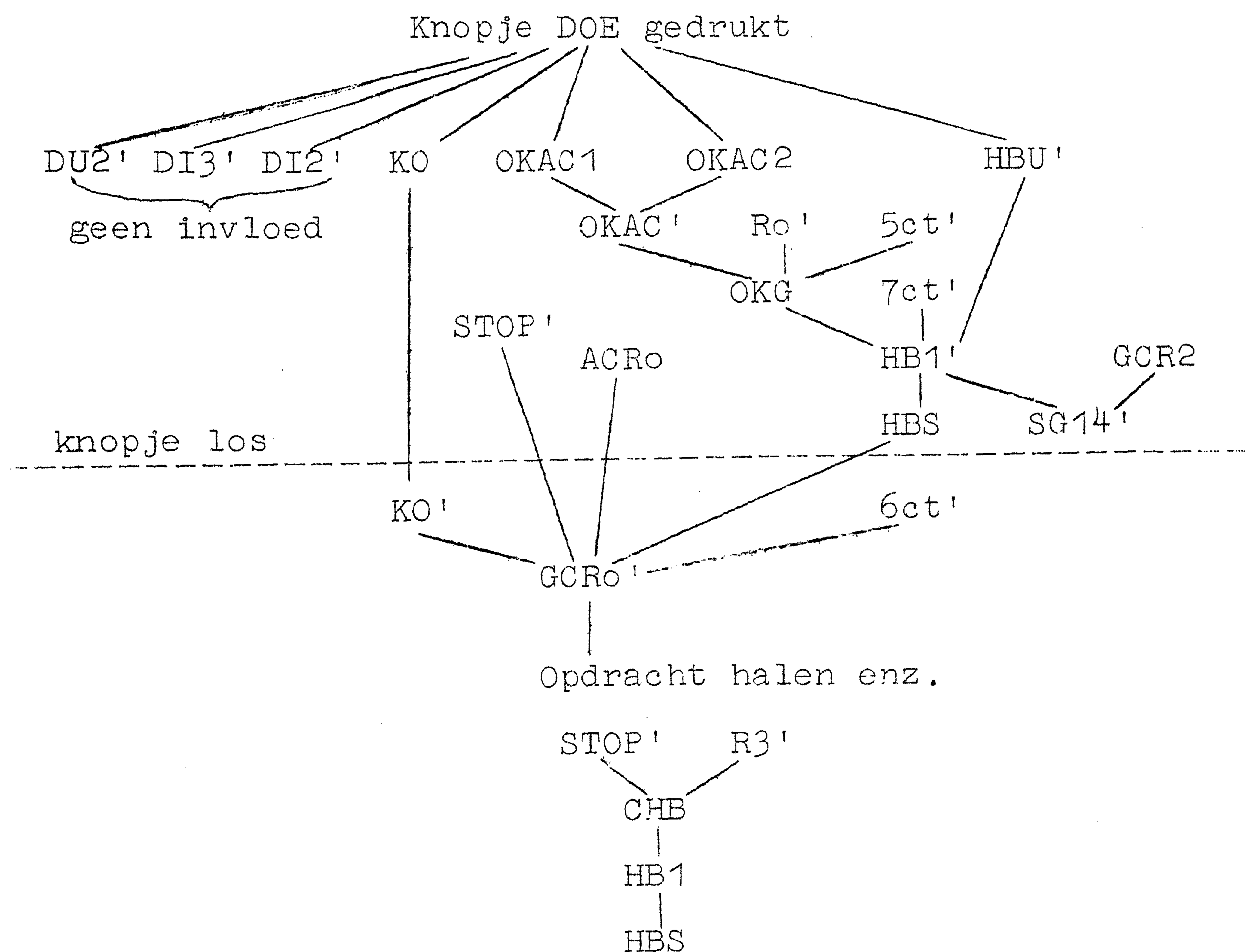
	ACRO'/ACRO	het tweede niveau van plaatje
1 Houd ACRO'		ACRO' zijn de signalen ACR1 en
2 0-1 ACR1 GCRo' GCR2 4gct'		GCR2 laag. Op 4gct wordt dit
		niveau laag ten gevolge van
1 Houd ACRO		signaal GCRo'. De arithmetische

controle ring gaat nu naar de eenstand.

Het signaal STOP' blijft laag. De X-1 gaat nu door tot de arithmetische controle ring in stand drie komt. Signaal R3' (uitcodering ACR:in3) wordt dan laag. Het signaal STOP' is nog steeds niet

veranderd. Deze beide signalen maken dan het 1e niveau van plaatje CHB' laag. Flip-flop HB1 wordt nu teruggesteld door signaal CHB. Ook flip-flop HBo staat in de nulstand. Signaal HBS is dus hoog. Ondertussen is ook de geheugen controle ring in de nulstand gekomen en kan slechts overgaan in de éénstand als signaal HBS weer laag is en signaal STOP' laag blijft. Dit laatste houdt tevens in, dat de GCR niet door de niveau's 3 en 4 van plaatje GCRo' in de éénstand kan worden gebracht, want op beide niveau's is het signaal STOP aangesloten. De X-1 stopt nu.

Hieronder volgt nu een kort overzicht van het ontstaan der besproken signalen.



Knopje "herhaal" (HH)

Wanneer op dit knopje is gedrukt, dan doet de X-1 precies hetzelfde als beschreven bij knopje DOE, alleen wordt nu de opdracht, die gegeven is door de getalschakelaars, bij herhaling uitgevoerd. Het neerdrukken van het knopje "herhaal" geeft dezelfde signalen, als genoemd zijn bij het knopje DOE en bovendien wordt signaal HBI' laag. Dit signaal maakt het leesniveau van plaatje HBo' laag op 7 cijfertijd, als signaal OKG' reeds laag is.

		HBo' / HBo		
1	Houd	HBo'	CHB	
2	Lees	HBI'	OKG'	7ct'

1	Houd	HBo
---	------	-----

		STOP' / STOP	
1	Houd	STOP'	

1	Houd	STOP
---	------	------

2		GCRo' GCR2	2gct' HBo' GCRo'
---	--	------------	------------------

signaal STOP' hoog is. Wel kan nu niveau 4 van dit plaatje laag worden. Op dit niveau zijn de signalen GCR2 en STOP laag. Wanneer

		GCRo' / GCRo			
1	Houd	GCRo'			
4	0-1	GCR2	R3'	2ct'	STOP

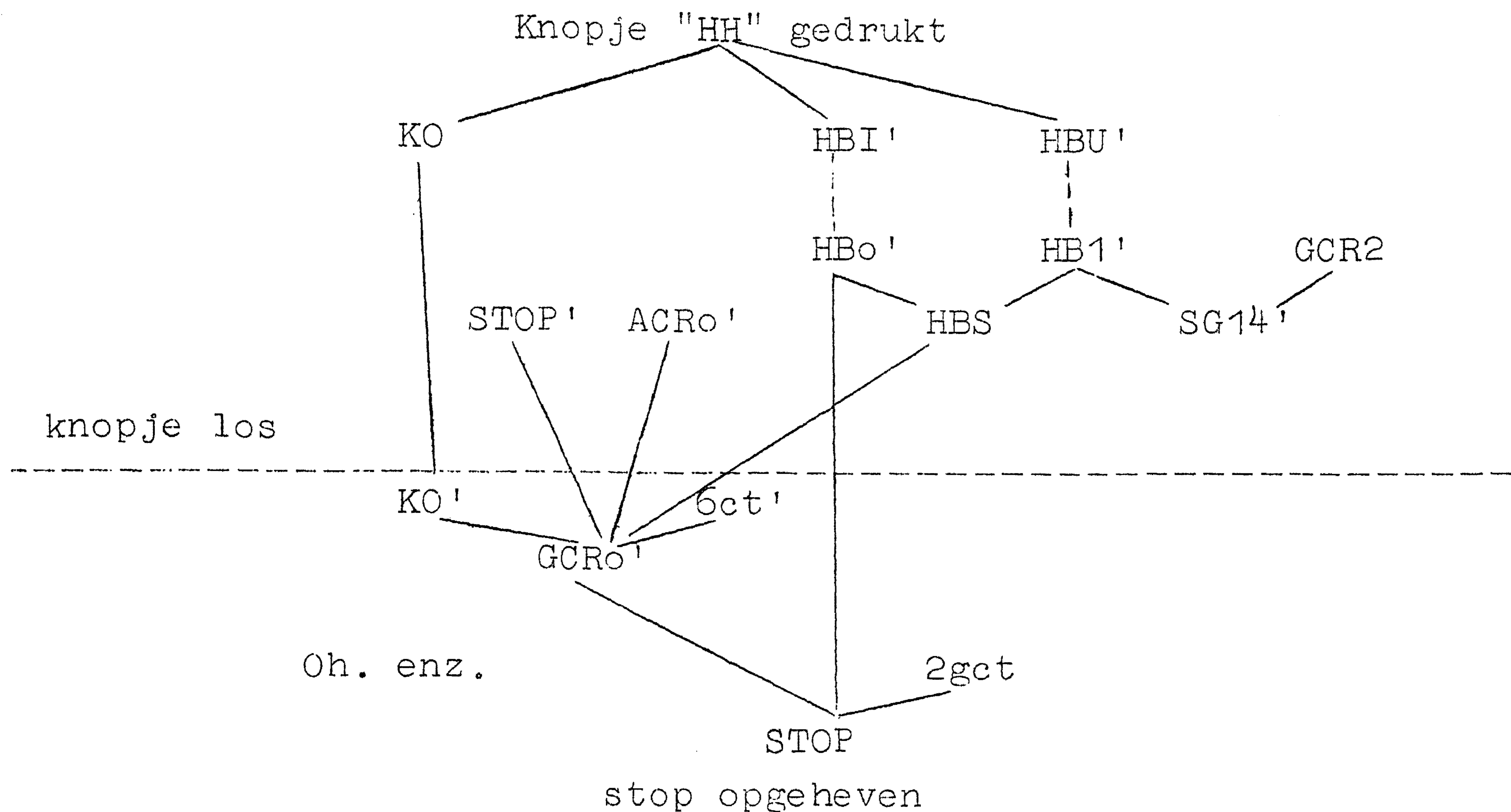
1	Houd	GCRo
---	------	------

Wanneer de geheugencontrolering in de éénstand is gezet en de geheugencijfertijdtring telt, dan wordt op 2gct het tweede niveau van plaatje STOP laag door signaal HBo'. Dit heeft tot gevolg, dat flip-flop STOP in de nulstand komt, waarmee de werking van signaal STOP' is opgeheven. Nu kunnen de flip-flops HBo en HB1 niet worden teruggesteld door signaal CHB, want dat is laag. Het niveau 5 van plaatje GCRo' kan niet laag worden, omdat

de arithmetische controlering in stand 3 komt, wordt op 2 cijfertijd de geheugencontrolering in de éénstand gezet. Nu kan de hele cyclus opnieuw worden afgewerkt. Dat gebeurt bij herhaling, want slechts door ingrijpen van buiten kan dit rondtellen van de ringen

worden gestopt.

Ook hier volgt een kort overzicht van het ontstaan der besproken signalen. De signalen, die in dit geval hetzelfde zijn als in het voorgaande geval, zijn achterwege gelaten.



Knopje "Begin volgend adres" (BVA)

Wanneer op het knopje BVA wordt gedrukt, heeft dat dezelfde handeling tot gevolg als het neerdrukken van knopje DOE, alleen wordt nu niet de opdracht uitgevoerd, die in de getalschakelaars staat, maar de opdracht, die zich bevindt op het adres, dat de opdracht-teller aangeeft. Daarom blijft flip-flop HB1 nu in de nulstand staan. Signaal HB1' is dus hoog. Ook signaal GCR2' is hoog gedurende de eerste vier standen van de GCR. Deze beide signalen maken signaal

SOT	SOT' laag. Dit laatste signaal beduidt: selecteer
1 GCR2'	het geheugen op grond van de opdrachtteller.
2 HB1	Signaal SOT', dat is aangesloten op het 2e niveau
	van de SG-plaatjes, maakt het mogelijk, dat de SG-

signalen de waarde vertegenwoordigen, die door de opdrachtteller wordt aangegeven. De SG-signalen selecteren nu het juiste adres in het geheugen.

Knopje "Begin gekozen adres" (BGA)

Nu wordt dezelfde handeling verricht als na het drukken op knopje BVA. In dit geval wordt echter begonnen met de opdracht, die zich bevindt op het adres, dat is aangegeven door de "begin adresschakelaars". Door het neerdrukken van knopje BGA wordt signaal BGA' laag.

Dit signaal is evenals signaal OKG' aangesloten op het niveau OTIC', zodat dit signaal dus laag is, als beide genoemde signalen laag zijn. Afhankelijk van de stand van de schakelaars zet signaal OTIC de betreffende flip-flops van de opdrachtteller in de éénstand of niet.

Wanneer een schakelaar contact maakt, wordt het houdniveau op het onaccent plaatje van de betreffende flip-flop hoog, zodat deze in de éénstand komt. De opdrachtteller wordt ingelezen op 5 cijfertijd,

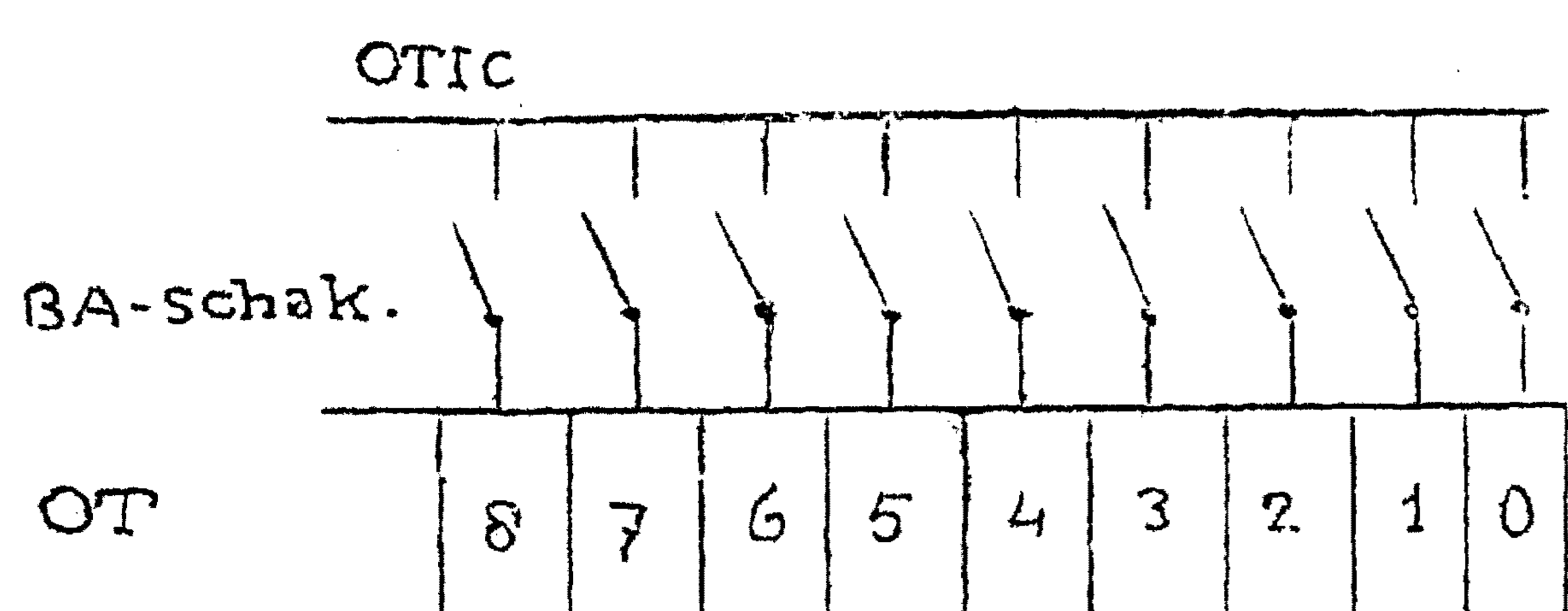


fig. 20.

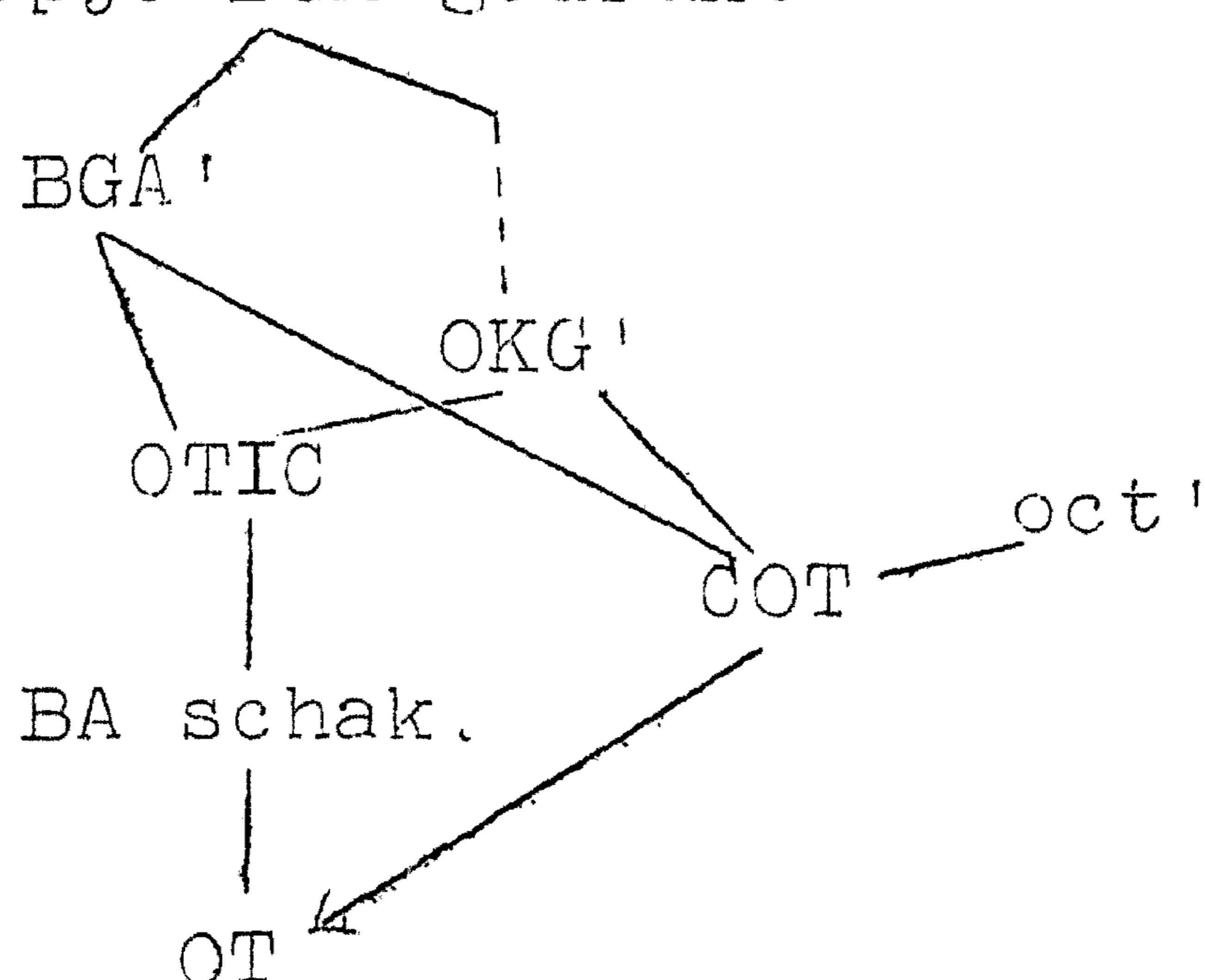
omdat signaal OKG' dan pas laag wordt. Dit is echter onbelangrijk, want de opdrachtteller is nog niet schoongemaakt. De inhoud van de opdrachtteller wordt nu vermengd et het adres uit de schakelaars. Daarom maken op 0 cijfertijd de signalen BGA' en OKG' niveau 7 van plaatje COT' laag. Zo kan signaal COT de opdrachtteller eerst terugstellen. Daarna zet signaal OTIC, dat pas eindigt op 2 cijfertijd, de

	OT7' / OT7
1 Houd OT7'	COT
1 Houd OT7	OTIC7 .

flip-flops van de opdrachtteller in de juiste standen.

Hieronder volgt een kort overzicht van het ontstaan van de besproken signalen.

knopje BGA gedrukt



Decimale knopjes

Slechts één cijferknopje zal worden besproken, omdat de andere daarna op eenvoudige wijze zijn af te leiden.

We veronderstellen dat de arithmetische en de geheugen controle ring in de nulstand staan en de cijfertijdtring telt. Het knopje DT6 wordt neergedrukt op b.v. 3 cijfertijd. Daardoor veranderen de signalen

DKAC1, DKAC2, DIO', DI1', DI2', DUO' en KO van potentiaal. De signalen DKAC1 en DKAC2, die nu hoog zijn, maken tezamen signaal DKAC' laag. Dit is weer het bewijs, dat het knopje voldoende is

	DKAC	neergedrukt en goed contact maakt. Op 5 cijfertijd
1	DKAC1	maakt signaal DKAC' het leesniveau van plaatje DKG'
2	DKAC2	laag, mits de signalen Ro', HBo en HB1 laag zijn.

De schakeling van flip-flop DKG is geheel analoog aan de besproken schakeling van flip-flop OKG. Op 7 cijfertijd maken de signalen DKG en 7ct de beide niveau's van plaatje HBO hoog, zodat signaal HBo' laag wordt. In samenwerking met het signaal KO' wordt nu de geheugencontrole ring in stand 1 gezet op dezelfde wijze als hiervoor besproken is.

Eveneens op 7 cijfertijd maakt signaal DKG' het 7e niveau van plaatje AI' laag. Signaal AI stelt nu alle flip-flops van het A-register terug. Het signaal DKG' maakt op 0 cijfertijd het 8^e niveau van plaatje COT' laag, zodat signaal COT alle flip-flops van de opdrachtteller terugstelt. Op adres 4 in het dood geheugen staat de eerste opdracht van het zgn. invoerprogramma. Dat programma verwerkt het getal, nadat het in het A-register is gezet. Signaal DKG duurt nog een cijfertijd na het terugstellen van de opdrachtteller en zet deze dan in de juiste stand. Daartoe worden de houdniveau's van de plaatjes OT2 en OT14 hoog gemaakt met signaal DKG. De opdrachtteller neemt zo het nieuwe adres over.

		Ao'/Ao	
1	Houd	Ao'	AI
3	Hand	DIO'	DKG' oct'

1	Houd	Ao
2	Hand	DUO' DKG'

		A2'/A2	
1	Houd	A2'	AI
3	Hand	DI2'	DKG' oct'

1	Houd	A2
2	Hand	DU2' DKG'

		A1'/A1	
1	Houd	A1'	AI
3	Hand	DI1'	DKG'

1	Houd	A1
---	------	----

		A3'/A3	
1	Houd	A3'	AI
3	Hand	DI3'	DKG'

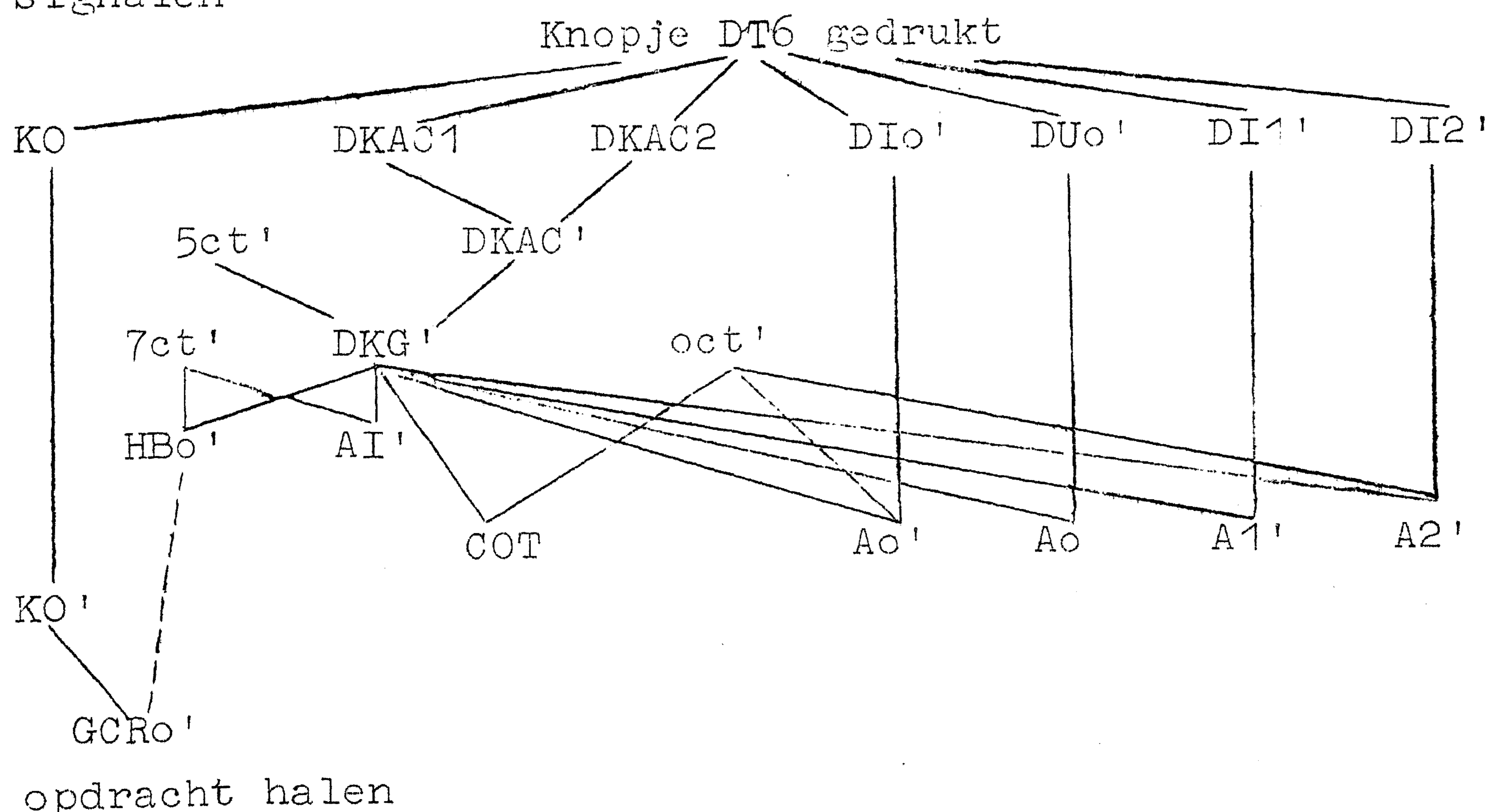
1	Houd	A3
---	------	----

Op niveau 3 van plaatje Ao' zijn de signalen DIo', DKG' en oct' aangesloten. Signaal Ao' wordt dus laag op 0 cijfertijd. Op het 2e niveau van plaatje Ao zijn de signalen DUo' en DKG' aangesloten. Deze signalen willen signaal Ao (onaccent!) laag maken. Dit laatste lukt altijd, omdat aan dit niveau geen cijfertijdpuls wordt toegevoerd en dit niveau dus langer laag blijft dan het niveau van het accentplaatje. Het resultaat is dus dat flip-flop Ao in de nulstand staat.

Het 3e niveau van plaatje A1' wordt laag ten gevolge van de signalen DI1' en DKG'. Flip-flop A1 staat nu in de éénstand. Niveau 3 van plaatje A2' wordt laag op 0 cijfertijd door de signalen DI2' en DKG'. De inhoud van flip-flop A2 is nu één. Alle andere flip-flops van het A-register worden niet door knopje DT6 beïnvloed, die blijven dus een nul bevatten. De inhoud van het A-register zegt nu welk decimaal cijfer door het knopje in de X-1 is gebracht:

$$0 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + 0 \times 2^3 = \underline{6} .$$

Dit woord kan door de X-1 verder worden verwerkt afhankelijk van het programma. Eveneens afhankelijk van het programma is het geval, wat de X-1 doet, als men b.v. eerst knopje 8 en daarna knopje 5 indrukt. Het resultaat is, dat de X-1 dit "leest" als het getal 85. Hieronder staat een kort overzicht van het ontstaan der besproken signalen



Normale in-opdracht (ongemodificeerd)

Volgens rapport EL-1-N bevatten de signalen OR15 t/m OR20 aanwijzingen omtrent enkele additionele mogelijkheden, welke in principe bij elk type opdracht voorkomen. Wanneer de opdrachten "normaal" moeten worden behandeld, is de inhoud van deze zes flip-flops nul. De gevallen waarin dit voorkomt staan vermeld in rapport EL-1-N. Bij een in-opdracht moet het arithmetisch resultaat of wat daarmee overeenkomt in een der interne registers worden geborgen. De volgorde der gebeurtenissen, die plaatsvinden bij deze soort opdrachten, staat aangegeven in het bovenste tijdschema op tek XE 10. Dat tijdschema zal nu worden besproken.

Signaal GCRo' wordt laag, doordat b.v. niveau 5 van dat plaatje laag gaat (zie handbediening). De geheugen controle ring wordt daardoor in stand 1 gebracht, d.w.z. er wordt een opdracht gehaald uit het geheugen op grond van het adres in de opdrachtteller. Het onmiddellijk gevolg hiervan is, dat de geheugen cijfertijd ring begint te tellen en dus de gct-signalen worden gevormd, zoals reeds eerder is beschreven (blz. 31). Door combinatie van signaal GCRo' en de signalen van de GCTR wordt het "drijfsignaal lees geheugen" (DL) gevormd. Nauwkeurige beschrijving van dat signaal volgt later.

CLG'	Doordat de geheugen controle ring in
1 Opdracht GCRo' GCR1 GCR2	stand 1 staat, wordt signaal CLG hoog en dat maakt het niveau van plaatje CMI' weer hoog. De andere signalen, die op dit niveau zijn aangesloten worden later besproken. Nu wordt aangenomen dat ze laag zijn. Signaal CMI is nu laag en maakt het 2e niveau van plaatje MI laag, maar dat heeft geen invloed, want signaal CM houdt het 1e niveau van dit
MI	plaatje reeds laag. Mocht signaal CM evenwel hoog worden
1 CM	dan blijft signaal MI' hoog, zodat het M-register de
2 CMI	inhoud van het U-register <u>niet</u> kan overnemen.
CM'	Het derde gevolg is, dat signaal GCRo' op
1 geheugen GCRo' 2gct'	2gct niveau 1 van plaatje CM' laag maakt, waardoor de flip-flops van het M-register worden teruggesteld. Het M-register is nu schoon en het woord uit het geheugen kan daarin worden overgenomen. Op 6gct staat het woord gega-randeerd in het M-register, daarna kan erover worden beschikt.

De flip-flops ACRO , GCR1 en GCR2 staan nog in de nulstand, zodat hiervan de onaccentsignalen laag zijn. Op 3gct maken deze signalen

CCTR'/CCTR		het 8e niveau van plaatje CCTR' laag,
1 Houd	CCTR' Co'	zodat nu het signaal Oct wordt gevormd en de CTR in de nulstand wordt gezet. Daarna telt de CTR door op de bekende wijze (blz. 28).
8 ACRO-1	ACRO GCR1 GCR2 3gct'	
CCTR		
1 Houd	CCTR	

Het vierde gevolg is, dat signaal GCRo' het 2e niveau van plaatje ACRO' laag maakt op 4gct, omdat de signalen ACR1 en GCR2 laag zijn.

ACRO'/ACRO		De arithmetische controle ring komt nu in de eenstand, hetgeen inhoudt, dat de opdracht kan worden geïnterpreteerd. Het signaal R1' is laag, omdat de ACR in de eenstand staat.
1 Houd	ACRO'	Dit signaal maakt op cijfertijd 1b het 5e niveau van plaatje COIo' laag, zodat, op de bekende wijze, de opteller in de nulstand (optelstand) wordt gezet.
2 0-1	ACR1 GCRo' GCR2 4gct'	
COIo'		
1 Houd	ACRO	
5 0. halen	R1' 1ct' B	

Op dezelfde cijfertijd wordt het 2e niveau van plaatje LB' laag ten

LB'/LB		gevolge van signaal R1'. Het coderend circuit OI selecteert nu het B-register. Uit de stand van zaken op dit moment kan de X-1 nog niet opmaken, of er B-correctie moet worden toegepast of niet. Uit tijdsoverwegingen wordt de opteller vast gereed gezet voor B-correctie. Blijkt dat het <u>niet</u> nodig is, dan is er ook minder haast en kan de opteller alsnog in stand 2 (transportstand) worden gezet.
1 Houd	LA LS LOT	
2 B-correctie	R1' 1ct' b	
LB		
1 Houd	LB	

ORFI'		Vervolgens maakt het signaal R1' op cijfertijd 2a' het niveau ORFI' laag, zodat het functiegedeelte van de opdracht op grond van de inhoud van het register M in het betreffende gedeelte van register OR wordt geschreven. Dat zijn de flip-flops OR15 t/m OR26 en dit registergedeelte wordt beïnvloed door de signalen ORFI' en ORFI. We nemen als voorbeeld flip-flop OR21. Het niveau ORFI' werd laag door de genoemde signalen. Het uitgangssignaal ORFI maakt het houdniveau van plaatje OR21' hoog en stelt deze flip-flop terug. Het signaal ORFI wordt ook omgekeerd en
1 R1' 2ct' a'		

dat maakt het mogelijk gelijktijdig de inhoud van flip-flop M21 over

	OR21'/OR21		te nemen op het leesniveau van plaatje OR21'
1 Houd	OR21'	ORFI	op de bekende wijze.
2 Lees	M21'	ORFI'	Nu is het functiegedeelte van de opdracht
1 Houd	OR21		beschikbaar en kan de X-1 bekijken, hoe volgens deze opdracht moet worden gehandeld.

	GCR1'/GCR1		Op 7gct' wordt de geheugen controle ring in stand 2 gezet, d.w.z. wacht tot de opdracht
1 Houd	GCR1'		terug kan worden gezonden naar het geheugen,
2 1-2	GCRo'	GCR2 7gct'	opdat eerst kan worden gecontroleerd of het woord goed is overgenomen (deze controle wordt later besproken) en daarna beslist
1 Houd	GCR1		kan worden, wat er met deze opdracht moet gebeuren. Zoals reeds werd opgemerkt, nemen we aan, dat de signalen OR15 t/m OR20 alle een nul voorstellen (EL-1-N). De signalen OR19' en OR20' zijn hoog. Hieruit trekt de X-1 de conclusie dat de opdracht ongemodificeerd naar het geheugen moet worden teruggezonden. Deze beide signalen maken niveau MB' hoog en signaal MB dus laag. De overige signalen op dit niveau zijn momenteel niet belangrijk. Signaal MB is aangesloten op

niveau 3 van plaatje GCRo (onaccent). Hierop is ook aangesloten signaal Ogct'. Op Ogct wordt dit niveau laag, want signaal MB is laag, zoals juist is besproken en de signalen GCR1' en GCR2 zijn laag in de standen 2 en 3 van de GCR. Het gevolg van dit alles is, dat signaal GCRo laag wordt en de geheugen controle ring nu in stand 3 komt. De opdracht kan nu zonder meer worden teruggezonden naar het adres in het geheugen, dat nog steeds in de opdrachtteller staat. Hiervoor is gelegenheid tot 6gct, zodat dan het woord gegarandeerd is opgeborgen. Pas daarna mag de inhoud van de opdrachtteller veranderen. Het "drijfsignaal schrijf geheugen" (DS) wordt gevormd door combinatie van signaal GCRo en de signalen van de GCTR. Beschrijving van dat signaal volgt later.

	GCRo'/GCRo		niveau 3 van plaatje GCRo (onaccent). Hierop is ook aangesloten signaal Ogct'. Op Ogct wordt dit niveau laag, want signaal MB is laag, zoals juist is besproken en de signalen GCR1' en GCR2 zijn laag in de standen 2 en 3 van de GCR. Het gevolg van dit alles is, dat signaal GCRo laag wordt en de geheugen controle ring nu in stand 3 komt. De opdracht kan nu zonder meer worden teruggezonden naar het adres in het geheugen, dat nog steeds in de opdrachtteller staat. Hiervoor is gelegenheid tot 6gct, zodat dan het woord gegarandeerd is opgeborgen. Pas daarna mag de inhoud van de opdrachtteller veranderen. Het "drijfsignaal schrijf geheugen" (DS) wordt gevormd door combinatie van signaal GCRo en de signalen van de GCTR. Beschrijving van dat signaal volgt later.
1 Houd	GCRo'		
1 Houd	GCRo		
3 2-3	GCR1'	GCR2 Ogct' MB	

Doordat de inhoud van flip-flop OR20 nul is, is signaal OR20' hoog. De X-1 concludeert hieruit, dat geen B-correctie moet worden toegepast. Signaal OR20' maakt het niveau van plaatje BC' hoog en signaal BC laag. De overige signalen op dit niveau zijn op dit moment niet belangrijk.

De arithmetische controle ring staat nog steeds in stand 1, d.w.z. signaal R1' is laag. Dit signaal maakt op cijfertijd 4a' het 1e niveau van plaatje COI2' laag, mits signaal BC laag is en dat is

COI2'	nu het geval. De opteller wordt door
1 Opdr. in R1' 4ct' a' BC	signaal COI2 in stand 2 gebracht.

Dat houdt in, dat de inhoud van het M-register, ongeacht OI, via het arithmetisch orgaan in het U-register wordt gezet (transport). Op cijfertijd 5a' wordt het 1e niveau

CU'	van plaatje CU' laag, als signaal R1'
1 Opdr. 5ct' a' R1'	nog laag is. Hierdoor wordt het U-

register schoongemaakt. Tevens maakt het signaal CU het 1e niveau van plaatje UI hoog.

UI	Het 2e niveau van dit plaatje is hoog door signaal CUI.
1 CU	Het ontstaan van dit laatste signaal wordt later be-
2 CUI	sproken. Het gevolg van deze beide signalen is, dat

signaal UI' laag wordt. Het U-register neemt nu het woord uit de somvormer over. Als voorbeeld nemen we flip-flop U7 (zie blz. 44). Het 2e niveau van plaatje U7' wordt laag als (SV7) = 1. Signaal CU wordt een omkering eerder laag dan signaal UI' hoog, zodat de flip-flop juist de waarde van signaal SV7' kan overnemen.

ORAI'	Als het woord, dat de opdracht voorstelt, eenmaal
1 R1' 6ct' b'	in het U-register staat, kan het adresgedeelte in

het betreffende gedeelte van het opdracht-register worden overgenomen. Daartoe worden de signalen ORAI' en ORAI gevormd op cijfertijd 6b', terwijl signaal R1' nog laag moet zijn. Met behulp

OR7'/OR7	van deze signalen wordt het adresgedeelte over-
1 Houd OR7' ORAI	genomen uit het U-register op dezelfde wijze
2 Lees U7' ORAI'	als het functiegedeelte werd overgenomen uit

het M-register. Eveneens op cijfertijd 6b' gaat het 2e niveau van plaatje LOT' laag, omdat signaal R1' nog laag is. Het uitgangssignaal van dit plaatje maakt het houdniveau van plaatje LB' hoog en schept tevens de mogelijkheid, na omgekeerd te zijn, voor het circuit OI

LOT'/LOT	om de opdrachtteller te selecteren. Ten derde
1 Houd LA LS LB	wordt op cijfertijd 6b' de opteller in stand 12
2 +1 R1' 6ct' b'	geplaatst. Zoals bekend is moet signaal COI12'

dat doen. Op niveau 3 van plaatje COI12' zijn dan ook de signalen 6ct', b' en R1' aangesloten.

1 Houd LOT

COI12' Het laatste signaal geeft aan, dat de X-1 nog
 3 +1 R1' 6ct' b' steeds bezig is de opdracht te interpreteren.
 Stand 12 van de opteller dient om het nummer
 van het adres, dat in de opdrachtteller staat met één te verhogen,
 zodat het adres voor de volgende opdracht in de opdrachtteller komt
 te staan en daarmee bepaald is, waar die opdracht moet worden gehaald.
 De logische opteller neemt de inhoud van de opdrachtteller via het
 circuit OI over, ongeacht de inhoud van het M-register (zie blz. 36
 en tek. X 4). Zoals bij de carryvormer is besproken (blz. 39) moet
 een carry eerst worden gemaakt om daarna bij de logische som te
 worden opgeteld. De logische opteller vertegenwoordigt een getal
 (adres uit OT), dat met één moet worden verhoogd. Wanneer nu het
 minst significante CV signaal een één voorstelt geeft de somvormer
 het nieuwe adres.

CV26'A/CV26A				SVo		CVo/CVo'						
2	+1	R1'	COB'	COD'	CCA'	1	LOo	CV26A	1	CV26A	LOo'	
<hr/>						2	LOo'	CV26'A	2	OIo	LOo	CCB
1	CV26A					3	OIo'	LOo	CCA			

Er zijn twee mogelijkheden. Als het minst significante cijfer van het getal in LO een nul is, moet het minst significante cijfer in de somvormer een één worden. Op plaatje CV26'A wordt het 2e niveau laag gemaakt door de signalen R1' (reden bekend), COB', COD' en CCA' (opteller stand 12). Het uitgangssignaal van dit plaatje maakt het eerste niveau van plaatje SVo (onaccent!) hoog, terwijl signaal LOo' het 2e niveau hoog houdt. Immers (LOo) = 0. Signaal SVo' is dus laag en (SVo) = 1.

In het tweede geval maakt signaal CV26'A het 2e niveau van plaatje SVo laag, als (LOo) = 1. Signaal SVo' wordt hoog en (SVo) = 0. Bovendien moet er nu een carry worden gevormd met plaatje CVo. Niveau 1 van dit plaatje wordt hoog door signaal CV26A, niveau 2 door signaal LOo en niveau 3 door signaal CCA. Signaal CVo' is nu dus laag en (CVo) = 1.

Op cijfertijd 2a wordt het 2e niveau van plaatje CU' laag (R1' is laag). Daardoor kan het U-register het woord uit de somvormer overnemen op de bekende wijze, na eerst te zijn teruggesteld.

COT'				Op 7gct mag de inhoud van de op-			
1	+1	GCRo	GCR1'	7gct'	FO	HB1	drachtteller veranderen in verband

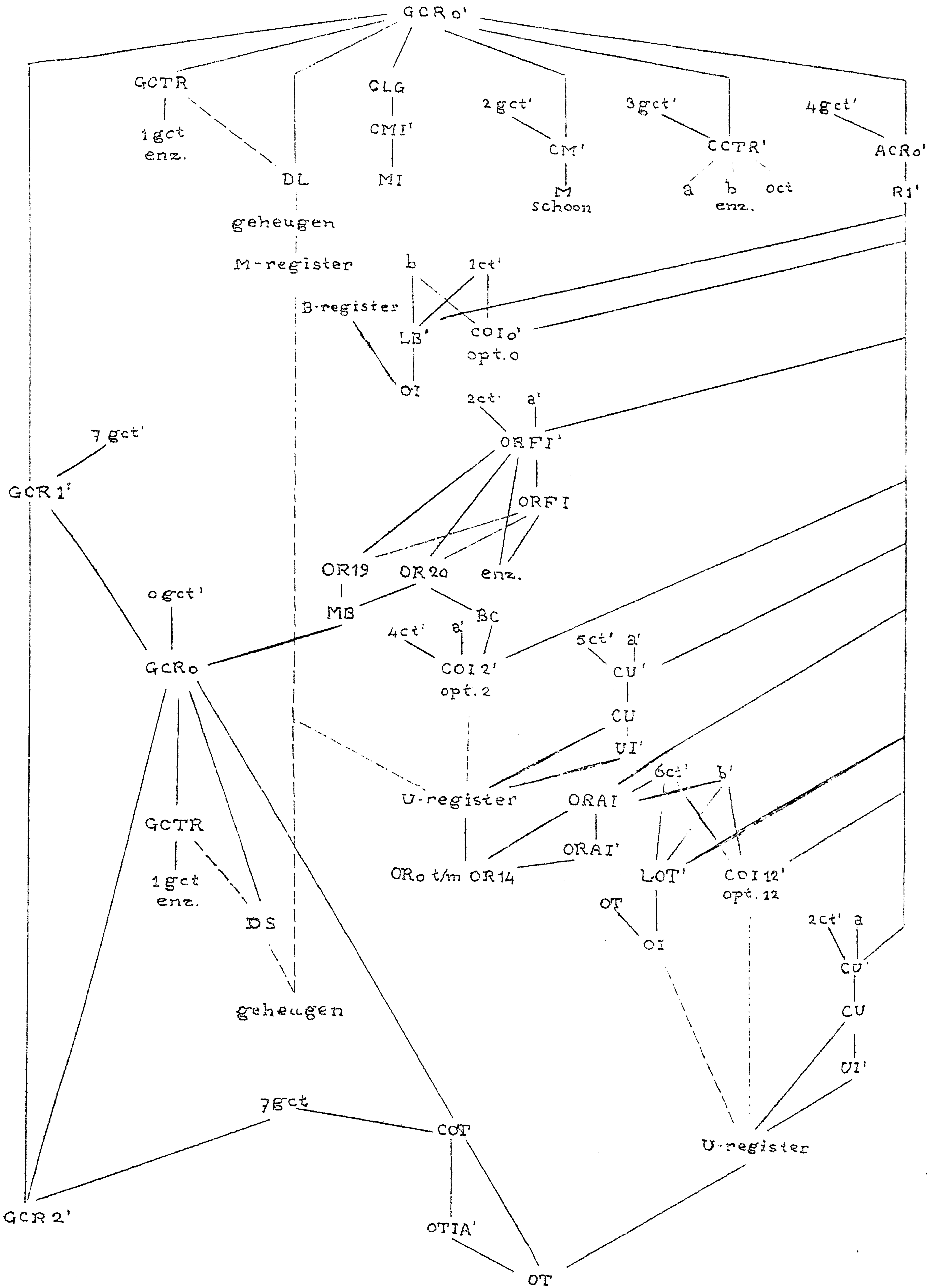
met het terugschrijven van de opdracht in het geheugen. Op deze cijfertijd wordt niveau 1 van plaatje COT' laag, doordat de geheugen controle ring in stand 3 staat (bij stand 4 is er geen signaal 7gct'), signaal FO laag is (opdracht niet fout, wordt later behandeld) en signaal HB1 laag is (opdracht uit geheugen, zie handbediening).

Signaal COT stelt de flip-flops van de opdrachtteller terug en maakt uitgangssignaal OTIA' laag, omdat de andere niveau's van dit plaatje hoog zijn. Het 2e niveau is hoog, omdat signaal ACRO hoog is (ACR in stand 1). De niveau's 3 en 4 zijn hoog, omdat de signalen, die hierop zijn aangesloten, slechts korte tijd laag zijn (zie handbediening). Met signaal OTIA' kan de opdrachtteller worden ingelezen.

Eveneens op 7gct wordt de geheugen controle ring in stand 4 gezet, d.w.z. wacht met het halen van het getal tot alle organen in de juiste standen staan, zoals de opdracht aangeeft.

Wat tot nu toe is gebeurd zullen we in het kort samenvatten. Op een zeker moment gaf de geheugen controle ring aan, dat een opdracht moest worden gehaald uit het geheugen op een door de opdrachtteller gegeven adres. Daartoe moest eerst het M-register worden "schoongemaakt" (CM). Het woord, dat op het genoemde adres stond, werd in het M-register gezet en het functiegedeelte werd daarna direct door het betreffende gedeelte van het opdrachtregister overgenomen. Vervolgens ging het hele woord uit M naar het circuit L0. In het arithmetische orgaan werd bekeken of de inhoud van het B-register erbij moest worden geteld. Dat was niet het geval. Daarop nam het U-register, dat gelijktijdig werd schoongemaakt, het woord ongewijzigd over en vanuit het U-register werd het adresgedeelte in het betreffende gedeelte van het opdrachtregister gezet. Zo stond de opdracht naar wens in het opdrachtregister. Omdat de opdracht, ongecorrigeerd, weer op hetzelfde adres in het geheugen moest worden gezet, waar het vandaan kwam, kon de inhoud van het M-register onveranderd (ongemodificeerd) in het geheugen worden geborgen. Toen moest nog worden bepaald, waar de volgende opdracht gehaald moest worden. Daartoe werd de inhoud van de opdrachtteller in het arithmetisch orgaan gebracht met één opgehoogd en weer in de opdrachtteller geborgen.

Op blz. 61 staat nog een overzicht van het ontstaan der verschillende signalen.



Verklaring van de signalen vermeld op tek. X-1 en tek. X-2

Het functiegedeelte van de opdracht kan men zich in twee stukken verdeeld denken: OR15 t/m OR20 en OR21 t/m OR26. Het laatste stuk specificceert het karakter van de opdracht (b.v. optelling, vermenigvuldiging, enz.). Combinaties van de signalen OR21 t/m OR26 moeten diverse niveau's in de X-1 beïnvloeden. Wanneer één van deze combinaties een groot aantal niveau's beïnvloedt, is het eenvoudiger zo'n combinatie uit te coderen op een plaatje en het aldus verkregen signaal op die betreffende niveau's aan te sluiten. Tevens heeft men dan op eenvoudige wijze de beschikking over het inverse signaal.

Tengevolge van de signalen OR21 t/m OR26 wordt b.v. signaal X' laag. De naam X specificceert de handeling, die moet worden verricht onder invloed van dit signaal. Op niveau's, die laag moeten worden bij handeling X kan signaal X' zijn aangesloten. Signaal X kan worden gebruikt voor niveau's, die bij deze handeling juist niet laag mogen worden.

Een voorbeeld is te vinden in het voorgaande bij de bespreking van de signalen geen B-modificatie (MB is laag) en geen B-correctie (BC is laag) op blz. 57.

De uitcodering van een aantal combinaties zal nu worden besproken. Alleen als het nodig is, wordt het signaal omgekeerd. De wijze, waarop dat gebeurt, is bekend en wordt hier niet verklaard.

AO'	Signaal AO' wordt laag, wanneer een
1 OR22 OR25	optelling moet worden verricht, dus bij
2 OR22 OR26'	een additieve opdracht. Deze gevallen

zijn op de beide niveau's uitgecodeerd. Vergelijken we, op tek. X-1 een tek. X-2, de opdrachten, waarachter in kolom AO een 1 is ingevuld, met de cijfers, die de signalen OR21 t/m OR26 voorstellen, dan blijkt $(AO) = 1$, wanneer $(OR22) = 0$ en $(OR25) = 0$ en ook wanneer $(OR22) = 0$ en $(OR26) = 1$. Met deze drie signalen zijn de additieve opdrachten bepaald.

AUO'	Signaal AUO' wordt laag, als het
1 OR22 OR23' OR25	arithmetisch resultaat niet in een intern register, maar in het geheugen moet

worden opgeborgen. Het zijn de additieve opdrachten, waarbij de pijl naar "n" wijst. (Het symbool n stelt een adres in het geheugen voor.)

Het blijkt, dat $(AU0) = 1$, als $(OR22) = 0$, $(OR23) = 1$ en $(OR25) = 0$. De signalen van deze drie flip-flops, die hierbij laag zijn, worden aangesloten op dit niveau.

BRO' 1 OR24 OR25 OR26'	Signaal BRO' wordt laag, wanneer er van het B-register wordt gebruik gemaakt. Dat is alleen het geval bij de opdrachten 32 t/m 39. Dan is $(BRO) = 1$, omdat $(OR24) = 0$, $(OR25) = 0$ en $(OR26) = 1$.
---------------------------	---

CO' 1 OR25' OR26'	De opdrachten 48 t/m 63 zijn communicatieopdrachten en alleen bij deze opdrachten mag $(CO) = 1$ zijn. Tevens moet gelden: $(OR25) = 1$ en $(OR26) = 1$.
----------------------	---

TSS' 1 OR23' OR24' OR25 OR26'	De waarde van signaal TSS' is 1 bij de opdrachten 44 t/m 47. Deze opdrachten zijn bepaald door de signalen van de flip-flops OR23, OR24, OR25 en OR26. Van deze flip-flops is de inhoud dan resp. 1, 1, 0 en 1.
----------------------------------	---

DRO' 1 SKIP OR23 OR25' OR26	De registers A en S kunnen tezamen één woord bevatten, zodat ze dan als het ware in elkaars verlengde liggen. Dat komt voor bij de opdrachten 16 t/m 19 en 24 t/m 27. Zo'n opdracht wordt een "dubbel register opdracht" genoemd en is bepaald door $(OR23) = 0$ $(OR25) = 1$ en $(OR26) = 0$. Bovendien is op het niveau van plaatje DRO' signaal SKIP aangesloten. De reden hiervoor ordt later behandeld.
--------------------------------	---

IO' 1 DRO OR23 2 OR23' OR25' OR26	Het signaal IO' is de uitcodering van de gevallen, waarbij het arithmetisch resultaat of wat daarmee overeenkomt in een intern register moet worden geborgen. Dat is het geval, wanneer het geen dubbel-register-opdracht is, $(DRO) = 0$ en $(OR23) = 0$ of wanneer geldt $(OR23) = 1$ $(OR25) = 1$ en $(OR26) = 0$.
---	--

U0 1 OR23 2 CO OR25'	De niveau's van plaatje U0 (onaccent!) zijn beide hoog als $(OR23) = 1$ en $(OR25) = 0$ of als $(OR23) = 1$ en $(CO) = 1$. Dit laatste geeft aan dat het een communicatieopdracht betreft. Signaal U0' is dan
----------------------------	--

laag. Dit signaal ontstaat bij de opdrachten, waarbij het resultaat op een bepaald adres in het geheugen moet worden geborgen.

NAIO'	
1 DRO OR22' OR23	De waarde van signaal NAI0' is 1 bij alle niet additieve opdrachten, als het woord uit het arithmetisch orgaan in een intern register moet worden geborgen met uitzondering van de dubbel-register-opdrachten. Dan moet gelden $(DRO) = 0$, $(OR22) = 1$ en $(OR23) = 0$. Het signaal OR23 is hier nogmaals aangesloten, hoewel dit reeds is verwerkt in het signaal DRO. Dat is nodig, omdat $(DRO) = 0$ kan zijn in gevallen, waarin $(OR23) = 1$. Signaal NAI0' mag echter alleen laag worden, als geldt $(OR23) = 0$.

NAUO'	
1 IO OR22' OR23'	Wanneer $(NAUO) = 1$ moet het woord uit het arithmetisch orgaan niet in een intern register worden geborgen. Het is <u>geen</u> in-opdracht, dus $(IO) = 0$. Verder geldt $(OR22) = 1$ en $(OR23) = 1$. Het signaal OR23 is hier aangesloten om analoge redenen, als besproken bij signaal NAI0'.

Om alle opdrachten uit te coderen worden de aldus verkregen signalen nog weer onderling en ook met signalen van het functiegedeelte der opdracht (OR-signalen) gecombineerd. Hiervan zullen we nog vele voorbeelden tegenkomen.

Gelijkheidscontrole (Parity check)

Bij de behandeling van het blokschema werd reeds opgemerkt, dat voor het woord in het M-register een nul of een één wordt geplaatst, zodanig dat het totaal aantal enen steeds oneven is (blz. 21). Het aantal enen moet dus worden geteld. Alle cijfers van het M-register worden verdeeld in negen groepjes van drie. Van elk groepje wordt bekeken of het aantal enen, dat daarin voorkomt, oneven is. Er zijn daarbij vier mogelijkheden, nl.:

1	0	0
0	1	0
0	0	1
1	1	1

Voor elke groep zijn deze vier gevallen op een plaatje uitgedeed.

PCoo'/PCoo				Signaal PCoo' wordt dus laag in elk van
1	Mo'	M1	M2	de genoemde gevallen. Op geheel analoge
2	Mo	M1'	M2	wijze worden de overige M-signalen op
3	Mo	M1	M2'	de plaatjes PCo1' t/m PCo8' aangesloten.
4	Mo'	M1'	M2'	De aldus verkregen signalen worden weer

verdeeld in drie groepjes van drie en ook weer op dezelfde wijze aangesloten respectievelijk op de plaatjes PC10' t/m PC12'. Van elk groepje wordt dan weer bekeken of het aantal enen hierin oneven is. Tenslotte worden deze drie signalen op dezelfde wijze aangesloten op het plaatje PC20'.

Wanneer nu het woord van 27 cijfers een oneven aantal enen bevat, wordt een van de niveau's van plaatje PC20' laag en signaal PC20 dus hoog.

In twee gevallen wordt een nieuw woord van 27 cijfers gevormd en in het M-register gezet; namelijk, als een opdracht gemodificeerd naar het geheugen moet worden gezonden (signaal MB' is

LPC'/LPC				laag) en als een geheel nieuw woord in
1	R1'	Oct'	b' MB'	het arithmetisch orgaan is gemaakt, dat
2	R3'	4ct'	UO'	in het geheugen moet worden gezet
				(signaal UO' is laag). Deze signalen
1	LPC			zijn elk op een niveau van plaatje LPC' aangesloten. De overige signalen op deze

niveau's bepalen het moment, waarop het M-register het betreffende woord moet overnemen.

Het aantal enen in het M-register wordt dan bepaald op de besproken wijze. Stel signaal PC20 wordt hoog. Dit signaal is aangesloten op het 3e niveau van plaatje M27'. (Dat is het plaatje voor het 28e cijfer!) Op het 1e en 3e niveau van dit

M27'/M27				plaatje zijn respectievelijk de sig-
1	Houd	M27'	CM LPC	nalen LPC en LPC' aangesloten, zodat
3	Lees	PC20	LPC'	flip-flop M27 op de bekende wijze de
				waarde van het signaal PC20 (omgekeerd!)
1	Houd	M27	kan overnemen. In dit geval wordt	
				(M27) = 0. Het resultaat is, dat het

totaal aantal enen oneven blijft.

Was het aantal enen even, dan wordt signaal PC20 laag en daardoor wordt $(M27) = 1$. Nu is het resultaat weer een oneven aantal enen.

Men lette erop, dat de waarde van het signaal PC20 altijd het omgekeerde is van de inhoud van flip-flop M27. Hiervan wordt gebruik gemaakt bij het controleren van het woord, dat uit het geheugen door het M-register wordt overgenomen. Ook flip-flop M27 neemt dan een cijfer uit het geheugen over.

Van de overige cijfers wordt het aantal enen geteld. Het resultaat wordt vergeleken met het cijfer, dat flip-flop M27 bevat. Daartoe worden op niveau 1 van plaatje FOUT (onaccent!)

	FOUT/FOUT'		de signalen M27' en PC20 aangesloten en op
1	M27'	PC20	niveau 2 de signalen M27 en PC20'. Wanneer
2	M27	PC20'	de waarden van deze signalen <u>ongelijk</u> zijn,
			zal één dezer niveau's laag zijn. Signaal
1	FOUT'		FOUT' is dan hoog. Dat beduidt, dat het

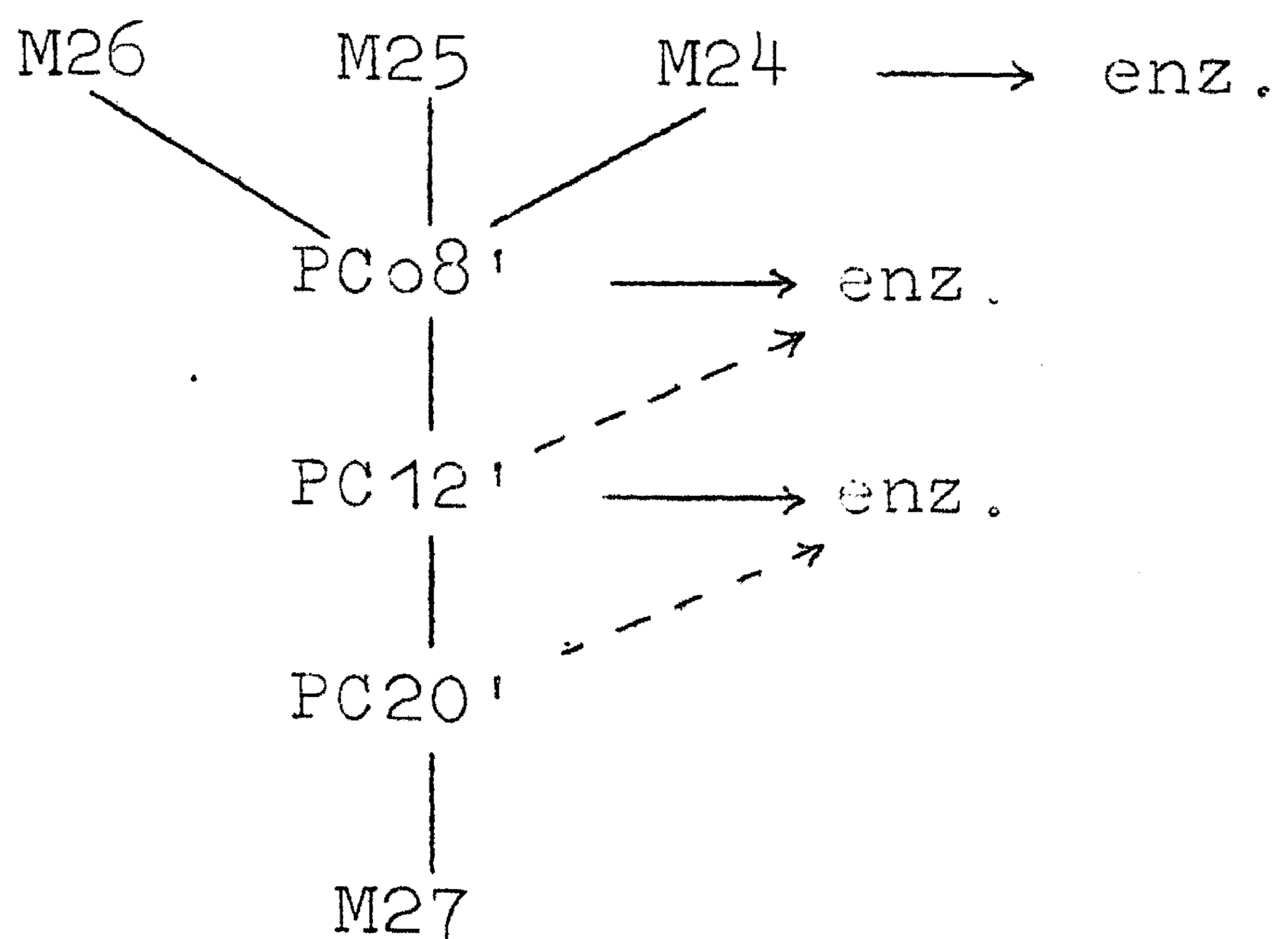
woord goed is overgenomen.

Zijn daarentegen de waarden van genoemde signalen gelijk, dan worden beide niveau's hoog door de onaccentsignalen als van deze signalen de waarde nul is en door de accentsignalen als de waarde één is. Nu wordt signaal FOUT' laag en dat beduidt, dat het woord verkeerd is overgenomen. Met de signalen FOUT' en FOUT worden diverse niveau's in de X-1 beïnvloed en wel zodanig, dat de X-1 de handeling gaat verrichten, die in dat geval noodzakelijk is.

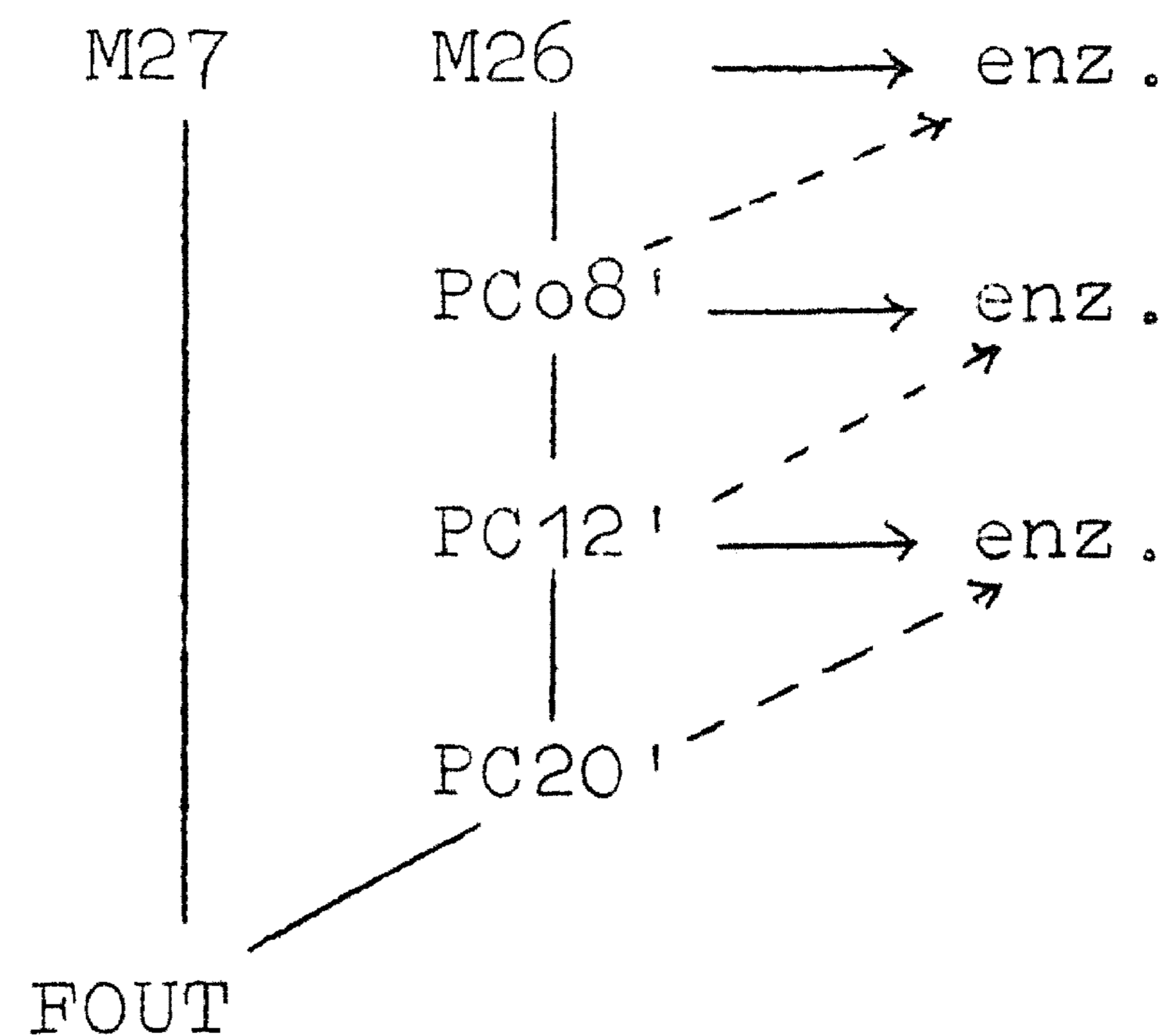
Op 6gct is het halen van een woord uit het geheugen voltooid en staat het gegarandeerd in het M-register. Er zijn acht omkeringen nodig om signaal FOUT te maken van de cijfers uit het M-register. Dat komt overeen met 4μ sec. Twee cijfertijden na 6gct is de controle dus bekend en kan over het woord worden beschikt.

Het overzicht hieronder laat het ontstaan van de signalen zien.

Het maken van het controle cijfer:



De controle:



Als de geheugen controle ring in stand 5 staat, wordt een getal uit het geheugen gehaald. Op de eerstvolgende Ogct is de de controle bekend, maar dan staat tevens de geheugen controle

<p style="text-align: center;">FG'/FG</p> <p>1 Houd FG' OKG DKG</p> <p>2 Lees ACRo' GCR1 GCR2' Ogct' FOUT'</p> <hr style="border: 0; border-top: 1px solid black; margin: 5px 0;"/> <p>1 Houd FG</p>	<p>ring in stand 6.</p> <p>Was het getal <u>niet</u> goed overgenomen, dan wordt signaal FOUT' laag. Op</p>
<p style="text-align: center;">STOP'/STOP</p> <p>1 Houd STOP'</p> <hr style="border: 0; border-top: 1px solid black; margin: 5px 0;"/> <p>1 Houd STOP FG FO</p>	<p>niveau van plaatje STOP (onaccent!) hoog. De X-1 stopt nu, omdat (STOP)=1. Flip-flop FG wordt teruggesteld, wanneer op een van de operationele knopjes is gedrukt: (OKG) = 1, of op één van de decimale knopjes: (DKG) = 1. De X-1 kan dus weer worden gestart door op één van de genoemde knopjes te drukken.</p>

Ook kan een opdracht uit het geheugen worden gehaald. De controle vindt plaats als de arithmetische controle ring in stand 1 staat (zie tijdschema's) Stel: de opdracht is niet goed overgenomen. Signaal FOUT' wordt dan laag. Op cijfertijd 4a' maakt signaal FOUT' het 2e niveau van plaatje FO' laag, mits signaal R1' (ACR in stand 1) laag is. Signaal FO maakt

	FO' / FO				het houdniveau van plaatje STOP
1	Houd	FO'	OKG	DKG	hoog en beïnvloedt nog diverse
2	Lees	R1'	4ct'	a' FOUT'	andere niveau's. De gevolgen
<hr/>					
1	Houd	FO			hiervan worden later besproken.

Instelling opteller bij een normale in-opdracht.

We zullen nu de bespreking van het tijdschema op tek. XE-10 vervolgen. Het is de bedoeling, de opteller in de juiste stand te zetten en het circuit OI het juiste register te laten selecteren, op grond van een opdracht, voordat een getal uit het geheugen wordt gehaald. Op dit moment kan de arithmetische operatie beginnen. Het niveau van plaatje BAO' wordt laag op cijfertijsd 3a, mits de arithmetische controle ring in stand 1 staat (signaal R1' is laag). Signaal BAO' is aangesloten op alle COI'-plaatjes, waarvan de uitgangssignalen de opteller in één van de standen kunnen zetten, die op het tijdschema zijn aangegeven. Signaal BAO' is eveneens aangesloten op niveau's van de plaatjes LA', LB' en LS', zodat circuit OI een van de in het tijdschema vermelde registers kan selecteren (zie blz. 33-39).

Het 2e niveau van plaatje LA' gaat laag door signaal BAO', mits deze opdracht er niet één is, waarbij gebruik wordt gemaakt van het B-register: (BR0) = 0. Verder moet bij overigens gelijklopende

opdrachten verschil worden gemaakt tussen het A- en het S-register. Daarom is signaal OR24 laag: (OR24) = 0. De voorwaarde, uitgecodeerd op dit niveau, komt overeen met de gegevens welke zijn vermeld op tek. X-1 en tek. X-2, bij de opdrachten, die het resultaat in het A-register doen opbergen.

	LB'				Bij andere in-opdrachten moet het
1	Houd	LA	LS	LOT	B-register worden geselecteerd. Nu
3	Normaal	BAO'		BRO'	doet signaal BAO' het 3e niveau van

plaatje LB' laag gaan. Signaal BRO' behoeft hier geen nadere verklaring.

LS' Op het 2e niveau van plaatje LS' is het
 1 Houd LA LB LOT signaal BAO' eveneens aangesloten en maakt
 2 Norm. BAO' SO OR24' dit niveau laag, als $(OR24) = 1$ en als
 $(SO) = 0$. Signaal SO is laag bij alle
 opdrachten, behalve als $(OR24) = 1$, $(OR25) = 0$ en $(OR26) = 1$.
 SO' Dit komt voor bij de zgn. sprongopdrachten.
 1 OR24' OR25 OR26' Uit het voorgaande blijkt, dat één van de
 registers A, B of S op grond van een
 opdracht kan worden uitgekozen door signaal BAO'.

De COI-niveau's, die betrekking hebben op een in-opdracht, worden eveneens laag door signaal BAO'. Op ieder van de hieronder besproken COI-plaatjes is een signaal van de flip-flop OR21 aangesloten. Deze signalen bepalen of beide getallen positief moeten worden beschouwd: $(OR21) = 0$, of één van beide getallen negatief moet worden geïnterpreteerd: $(OR21) = 1$. Hierbij dient te worden opgemerkt, dat bij in-opdrachten het getal uit het interne register altijd positief wordt beschouwd, zodat dan het M-register negatief wordt geïnterpreteerd, als $(OR21) = 1$.

COIo' Signaal COIo zet de flip-flops
 6 Optelling BAO' TSS AO' OR21 "Controle Opteller" in de optel-
 stand. De opdracht moet dan
 additief zijn: $(AO) = 1$ en geen "tellende en subroutine sprong":
 $(TSS) = 0$. Bovendien moet het M-register positief worden
 beschouwd: $(OR21) = 0$. Op deze voorwaarden maakt signaal BAO'
 het signaal COIo hoog, met de bekende gevolgen.

COI1' Signaal COI1 zet de
 5 Aftrekking BAO' DRO OR21' OR22 OR23. opteller in de optel-
 stand, waarbij de inhoud
 van het M-register negatief wordt geïnterpreteerd. De opdracht
 moet dan geen dubbel register opdracht: $(DRO) = 0$, additief:
 $(OR22) = 0$ en een in-opdracht zijn: $(OR23) = 0$, tevens moet het
 getal uit het M-register negatief worden beschouwd: $(OR21) = 1$.
 Signaal BAO' maakt dit niveau laag op deze voorwaarden.

COI2' Signaal COI2 zet de opteller in
 2 Schoon in BAO' NAI0' OR21 de transportstand, waarbij de
 inhoud van het M-register zonder
 meer wordt overgenomen. Dit is een niet additieve in-opdracht:

(NAIO) = 1, terwijl het M-register positief wordt beschouwd: (OR21) = 0. Signaal BAO' maakt dit niveau laag.

COI3'	Door signaal COI3 wordt de op-
1 Schoon in BAO' NAIO' OR21'	teller in de transportstand gezet, waarbij de inhoud van het M-register negatief wordt geïnterpreteerd. Dus: (OR21) = 1. Overigens is dit niveau gelijk aan het voorgaande.

COI8'	De opteller komt in de
1 Logische optelling BAO' AO DRO OR21 OR22	stand "logische optelling" door signaal COI8.

Signaal BAO' doet dit niveau laag gaan, mits de opdracht geen additieve opdracht (in de normale zin van het woord) is: (AO) = 0 en geen dubbel-register opdracht: (DRO) = 0. Het getal uit het M-register moet positief worden beschouwd: (OR21) = 0. Tenslotte geldt voor nadere bepaling van de opdracht: (OR22) = 0.

COI9'	De opteller komt in de
1 Logische optelling BAO' AO DRO OR21' OR22	stand "logische optelling" door signaal COI9, waarbij

het getal uit het M-register negatief wordt geïnterpreteerd: (OR21) = 1. Verder is dit niveau gelijk aan het niveau van plaatje COI8'.

COI10'	Signaal COI10 zet de
1 Collatie BAO' IO' OR21 OR22' OR23'	opteller in de stand "logische vermenigvuldiging", terwijl het getal uit het M-register positief wordt beschouwd: (OR21) = 0 Dit is een in-opdracht: (IO) = 1. Deze opdracht wordt nog nader gespecificeerd door: (OR22) = 1 en (OR23) = 1.

COI11'	Het signaal COI11 zet
1 Collatie BAO' IO' OR21' OR22' OR23'	de opteller eveneens in de stand "logische vermenigvuldiging". Hierbij wordt de inhoud van het M-register negatief beschouwd: (OR21) = 1. Verder zijn op dit niveau dezelfde signalen aangesloten als op dat van plaatje COI10'.

Getal bij een normale in-opdracht

Als de opdracht tot zover is uitgevoerd, kan het getal worden gehaald op het adres, dat in het opdrachtregister staat aangegeven (adresgedeelte). Dat getal ondergaat dan, afhankelijk van de opdracht, een van bovengenoemde bewerkingen. Daartoe wordt de arithmetische controle ring in stand 2 gezet. Op cijfertijd 4a

ACR1'/ACR1		
1	Houd	ACR1'
2	1-2	ACRo' 4ct' a FOUT
<hr/>		
1	Houd	ACR1

wordt niveau 2 van plaatje ACR1' laag, want signaal ACRO' is laag in de standen 1 en 2 van de arithmetische controle ring. Het signaal FOUT stelt de voorwaarde, dat de opdracht nog goed in het M-register staat.

De overige signalen op dit niveau (hier niet vermeld) worden later besproken en zijn laag. Op Ogct wordt de geheugen controle ring in stand 5 gezet, doordat de signalen GCR1', GCR2' (GCR in standen 4 en 5) en Ogct' op het 2e niveau van het plaatje GCRo'

GCRo'/GCRo		
1	Houd	GCRo'
2	4-5	GCR1' GCR2' Ogct'
<hr/>		
1	Houd	GCRo
<hr/>		
GCR1'/GCR1		
1	Houd	GCR1'
<hr/>		
1	Houd	GCR1
3	5-6	GCRo' GCR2' 7gct'

zijn aangesloten. Op 2gct ontstaat signaal CM, omdat signaal GCRo' weer laag is. Het M-register wordt nu schoongemaakt, zodat het getal uit het geheugen daarin kan worden overgenomen. Op 7gct moet het halen van het getal zijn voltooid, want dan wordt de geheugen controle ring in stand 6 gezet, door het 2e niveau van plaatje GCR1 laag te maken met de signalen GCRo', GCR2' (GCR in de standen 5 en 6) en 7gct'. Ook nu kan

eerst op de bekende wijze worden gecontroleerd of het woord (getal) goed is overgenomen. Daarna kan het woord weer in het geheugen worden geborgen, want het M-register onthoudt nu het getal.

GCRo'/GCRo		
1	Houd	GCRo
<hr/>		
1	Houd	GCRo
4	6-7	GCR1 GCR2' Ogct' AUO

De geheugen controle ring moet nu in stand 7 worden gezet, om het het getal terug te zenden. Niveau 4 van plaatje GCRo wordt laag op Ogct, als de signalen GCR1 en GCR2' laag zijn (GCR in standen 6 en 7) en stelt deze flip-flop terug. Het signaal AUO stelt de

voorwaarde, dat dit niveau alleen laag mag worden als deze opdracht geen additieve uit-opdracht is.

Het getal kan nu in het arithmetisch orgaan de bewerking ondergaan, die door de opdracht is bepaald.

		CCTR' / CCTR	
1	Houd	CCTR' Co'	Op cijfertijd 1b' gaat niveau 3 van plaatje CCTR' laag, omdat signaal R2' nog laag is (ACR in stand 2).
3	ACR 2-3	R2' 1ct' b' DRO CO	
<hr/>			
1	Houd	CCTR	De signalen aangesloten op de 4e en 5e ingang van dit

niveau stellen de voorwaarde, dat deze opdracht geen dubbelregister- en geen communicatieopdracht mag zijn. Signaal CCTR' stelt de "gewone" cijfertijd ring terug op de bekende wijze, zodat deze opnieuw begint te tellen. Bovendien maakt dit signaal

		ACRo' / ACRo	
1	Houd	ACRo'	het 2e niveau van plaatje ACRo laag, als de signalen ACR1' (ACR in standen 2 en 3) en Co' (kloksignaal) laag zijn. De arithmetische controle ring staat nu in stand 3, d.w.z. "einde van de opdracht". Nu kan de geheugen
1	Houd	ACRo	
2	2-3	ACR1' CCTR' Co'	controle ring in stand 0 worden gebracht en het arithmetisch resultaat op de bestemde plaats worden geborgen.

Op 7gct wordt het 2e niveau van plaatje GCR2 laag, doordat de signalen GCRo en GCR1 laag zijn (GCR in standen 7 en 0). De

		GCR2' / GCR2	
1	Houd	GCR2'	geheugen controle ring staat nu in de nulstand en wacht met het halen van een nieuwe opdracht tot het sein daarvoor wordt gegeven.
1	Houd	GCR2	
2	7-0	GCRo GCR1 7gct'	Op de eerstvolgende 1 cijfertijd wordt het 3e niveau van plaatje CU' laag, als signaal R3' laag is (ACR stand 3) en als deze opdracht

geen dubbelregister- en ook geen schuifopdracht is. Op de bekende wijze neemt het U-register het

		CU'	
3	resultaat	R3' 1ct' DRO AS	woord uit de somvormer over Afhankelijk van de opdracht wordt de inhoud van het U-register door een van de interne registers overgenomen. Daartoe wordt op de eerstvolgende 2 cijfertijd het betreffende "inlees- en terugstelsignaal" gemaakt, mits de

wordt de inhoud van het U-register door een van de interne registers overgenomen. Daartoe wordt op de eerstvolgende 2 cijfertijd het betreffende "inlees- en terugstelsignaal" gemaakt, mits de

arithmetische controle ring in stand 3 staat. Maar daar komen we nog op terug. Onder dezelfde voorwaarden ($R3'$, $2ct'$) wordt

	$GCRo'/GCRo$				de geheugen controle ring in
1 Houd	$GCRo'$				stand 1 gebracht, want niveau 4
4 0-1	$GCR2$	$R3'$	$2ct'$	STOP	van plaatje $GCRo'$ wordt laag
<hr/>					
1 Houd	$GCRo$				tengevolge van de signalen $GCR2$,
					$R3'$, $2ct'$ en STOP. Het laatste
					signaal beduidt, dat het programma

de X-1 wil laten doorgaan. De geheugen controle ring laat nu vast de nieuwe opdracht halen, die beschikbaar is op het adres, dat in de opdrachtteller staat. Daartoe wordt op $2gct$ het M-

	$ACR1'/ACR1$				register schoongemaakt met behulp van
1 Houd	$ACR1'$				signaal CM. Op 5 cijfertijd wordt de
<hr/>					
1 Houd	$ACR1$				arithmetische controle ring in stand 0
2 3-0	$ACRo$	$5ct'$			gezet door flip-flop $ACR1$ terug te stellen
					met niveau 2 van plaatje $ACR1$, want
					signaal $ACRo$ is laag (ACR stand 3 en 0).

Alles is nu gereed voor de volgende opdracht.

Tenslotte zullen we het ontstaan van de genoemde inleessignalen nader onderzoeken. Zoals reeds is opgemerkt, ontstaan deze signalen op 2 cijfertijd, als de arithmetische controle ring in stand 3 staat (signaal $R3'$ is laag). De hier behandelde op-

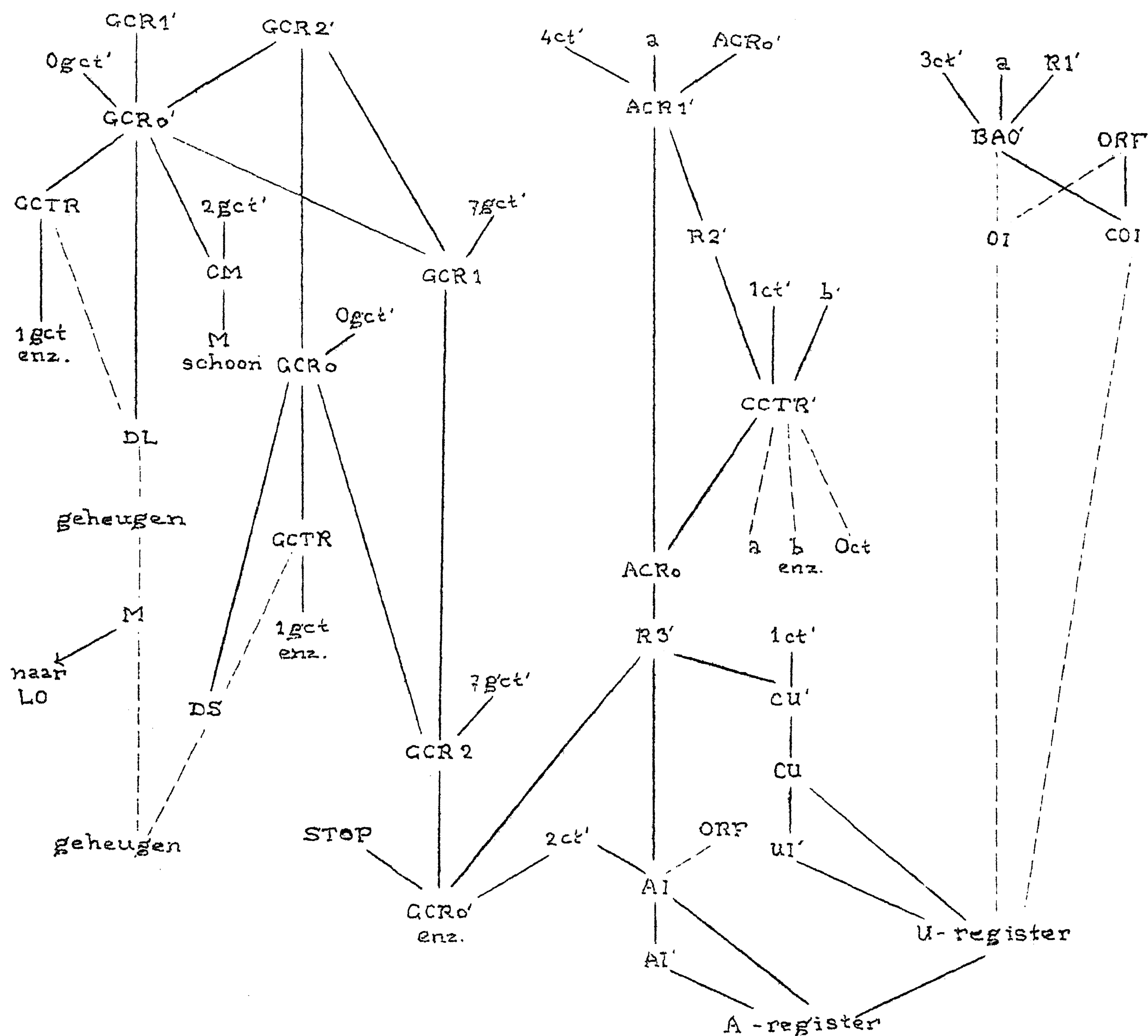
	AI'				drachten zijn <u>in</u> -opdrach-		
1 In-opdr.	$R3'$	$2ct'$	IO'	BRO	NRW	$OR24$	ten: $(IO) = 1$, deze
							opdracht is geen B-regis-
							ter opdracht: $(BRO) = 0$, tussen het A- en het S-register wordt
							het eerste gekozen: $(OR24) = 0$ en de inhoud van dit register
							mag <u>wel</u> worden gewijzigd: $(NRW) = 0$. De signalen, die bij de
							genoemde waarden laag zijn, maken niveau 1 van plaatje AI' laag.

	SI'				De eerste vier signalen		
1 In-opdr.	$R3'$	$2ct'$	IO'	NRW	SO	$OR24'$	op niveau 1 van plaatje
							SI' zijn nu zonder meer duidelijk. De signalen SO en $OR24'$ be-
							palen het S-register.

	BI'				Signaal BRO' op niveau 1 van	
1 In-opdr.	$R3'$	$2ct'$	IO'	NRW	BRO'	plaatje BI' geeft aan, dat
						dit niveau alleen laag mag
						worden bij B-register opdrachten. Dit zowel als de overige sig-
						nalen op dit niveau behoeven geen nadere verklaring.

Van deze drie signalen wordt ook het inverse signaal gemaakt, waarna de betreffende registers op de reeds besproken wijze worden ingelezen.

Hieronder volgt een overzicht van het ontstaan der signalen.



Tenslotte wordt signaal COT hoog bij de opdrachten 40 t/m 42 door

3 gewone sprong COT' R3' 2ct' OR23 OR24' OR25 OR26' specificatie van het functiegedeelte der opdracht. Deze opdrachten zijn de zgn. sprongopdrachten, waarbij de inhoud van de opdracht teller zodanig wordt gewijzigd, dat de eerstvolgende opdracht niet wordt gehaald op het adres, dat ontstaat na de ophoging van het nummer van het adres, maar op het nieuwe adres, dat de opdracht teller aangeeft na de wijziging.

Deze wijziging ontstaat door de inhoud van adres n positief of negatief op te tellen bij de inhoud van de opdrachtteller op dat moment, d.w.z. na de ophoging. Bij opdracht 42 neemt de opdrachtteller de inhoud van adres n zonder meer over, waarbij dat getal uiteraard niet negatief kan zijn, aangezien er geen negatieve adressen bestaan. Om dezelfde reden staat opdracht 43 wel in de rij der opdrachten, maar is onmogelijk.

Signaal COT maakt signaal OTIA' laag, zoals op blz. 60 reeds is besproken met dien verstande, dat bij deze opdrachten signaal TSS' gegarandeerd hoog is op niveau 2, terwijl op dit moment signaal ACRO laag is.

De inhoud van adres n gaat natuurlijk via het arithmetisch orgaan en komt zodoende in het U-register. Met behulp van signaal OTIA'

OT7'/OT7			neemt de opdrachtteller de inhoud van het U-register over. De aandacht wordt erop gevestigd, dat bij deze opdrachten niet een niveau van plaatje LOT' laag behoeft te gaan door signaal BAO', want signaal LOT'
1 Houd	OT7'	COT	
2 Lees	U7'	OTIA'	
<hr/>			
1 Houd	OT7		

was op cijfertijsd 6b' reeds laag geworden om andere redenen en blijft dan eenvoudig laag.

In-opdracht met B-correctie (ongemodificeerd)

Bij dit soort opdrachten is het de bedoeling, het adresgedeelte der opdracht te wijzigen. Het cijfer OR19 is dan nul en OR20 is één (zie rapport EL-1-N). De volgende der gebeurtenissen, die plaatsvinden bij deze opdrachten staat aangegeven in hetzelfde tijdschema op tek. XE 10, dat is besproken bij de normale in-opdracht. Ook alle signalen ontstaan op dezelfde wijze met één uitzondering. Signaal OR20' is laag, zodat het niveau van plaatje BC' laag is en signaal BC dus hoog, want deze opdracht

BC'			is geen tellende en subroutinesprong: (TSS) = 0, de opdracht mag niet fout zijn overgenomen: (FOUT) = 0 en
1 TSS FOUT SKIP	OR20'		

signaal SKIP moet laag zijn. Op de laatste signaal komen we later nog terug.

Doordat signaal BC hoog is, kan niveau 1 van plaatje COI2' nu niet laag worden, de opteller blijft dus in de optelstand (stand 0) staan. De inhoud van het B-register wordt nu dus wel bij het woord uit het M-register geteld. Register B bevat evenwel 15 cijfers en een tekencijfer (TB). De signalen van de flip-flops Bo t/m B14 worden naar de plaatjes OIo' t/m OI14' gezonden. Signaal TB is aangesloten op niveau 2 van plaatje TBS.

TBS	Op het 1e niveau van dit plaatje komt signaal LB,
1 LB	terwijl het uitgangssignaal TBS' is aangesloten op
2 TB	de plaatjes OI15' t/m OI26'. Er zijn in principe
	vier gevallen mogelijk:

LB	TB	TBS
0	0	0
0	1	0
1	0	0
1	1	1

In de eerste twee gevallen mag geen van de niveau's van de OI plaatjes, waarop de signalen van het geselecteerde register binnenkomen laag kunnen worden ten gevolge van het multiplet-sig-naal (zie blz. 34). In deze beide gevallen is het B-register niet geselecteerd, dus dan mag signaal TBS' ook niet laag worden. In het 3e geval is wel het B-register geselecteerd, maar (TB) = 0. Cok nu moet signaal TBS' dus hoog zijn.

Alleen in het 4e geval moet signaal TBS' laag worden, want het B-register is geselecteerd en het teken van het getal in het B-register is negatief.

In het 3e geval is signaal TBS' dus hoog en verder zijn volgens het multipletprincipe alle niveau's hoog van de OI-plaatjes, waarop dit signaal is aangesloten. Circuit OI leest het positieve getal uit het B-register, alsof het aan de meest significante zijde is aangevuld met nullen.

Indien het getal in het B-register negatief is, wordt het door OI beschouwd als een getal, dat aan de meest significante zijde met 12 enen is aangevuld, met inbegrip dus van teken B. Immers nu is signaal TBS' laag en maakt het 3e niveau van de betreffende OI-plaatjes laag, zodat deze OI-signalen een één vertegenwoordigen. De 15 cijfers van het B-register worden bij de 15 minst significante cijfers van het M-register geteld, terwijl met het tekencijfer op de beschreven wijze wordt gehandeld.

Het functiegedeelte van de opdracht bestaat uit de 12 meest significante cijfers van het M-register en die mogen niet worden gewijzigd. Het tekencijfer van M is in dit geval een onderdeel van het functiegedeelte der opdracht en kan zowel nul als één zijn. De inhoud van het B-register moet als positief of als negatief getal bij M kunnen worden geteld. Hierbij heeft het tekencijfer van B dus de oorspronkelijke betekenis.

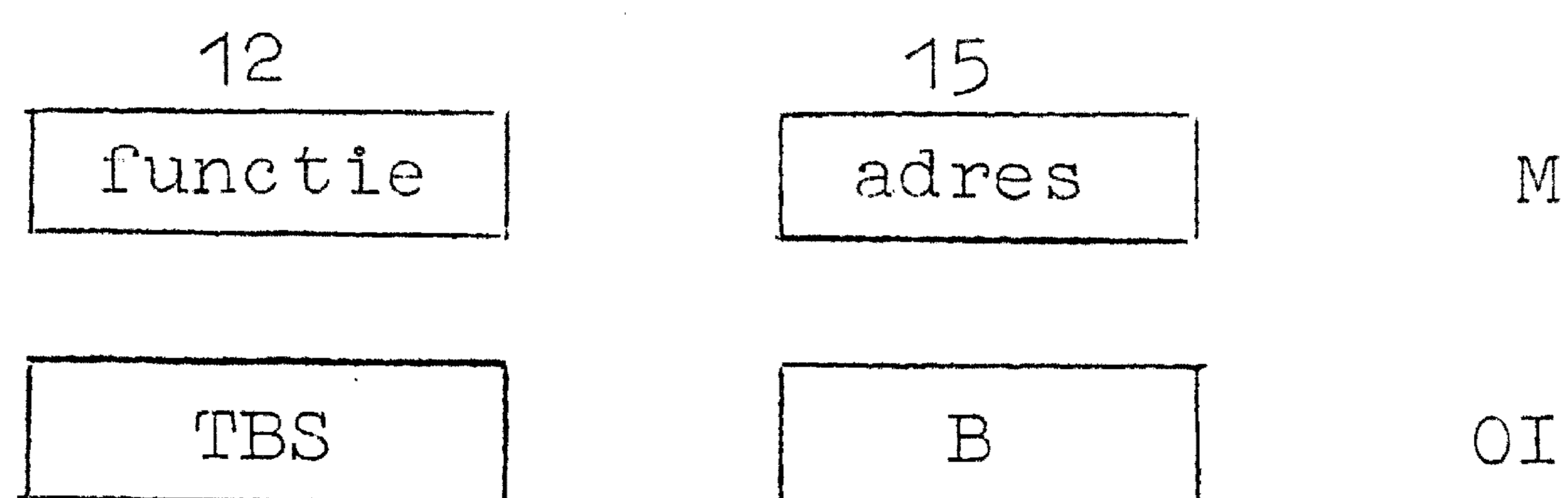


fig. 21

De meest significante cijfers van M kunnen zowel één als nul zijn, daarom stellen we ze gelijk aan x . Dan moeten de overeenkomstige cijfers van het U-register eveneens x worden, ongeacht TB. Het functiegedeelte in het U-register is bij opdrachten met B-correctie niet belangrijk, maar bij opdrachten met B-modificatie wel, zoals we bij de behandeling daarvan zullen zien. Voor beide soorten opdrachten wordt dezelfde arithmetische bewerking gebezigd. Daarom wordt ook hier het functiegedeelte beschermd. Als het teken B positief is, geeft het geen moeilijkheden, want $(TBS) + x = 0 + x = x$. Bovendien is het onmogelijk, dat een carry wordt gevormd door de 15e cijfers van beide registers, want dat zou inhouden, dat er naar een niet bestaand adres wordt verwezen. In het programma dient hiermede rekening te worden gehouden. Het functiegedeelte blijft dus onveranderd.

In het algemeen geldt bij optellingen, waarbij negatieve getallen voorkomen, dat men deze kan uitvoeren door beide getallen, met inbegrip van de tekens, gewoon bij elkaar te tellen en een overdracht, welke eventueel uit de beide tekencijfers ontstaat, aan de minst significante zijde er weer bij te tellen:

1	1	1	0	0	-3
0	0	1	0	0	+4
1	0	0	0	0	0
					1
0	0	0	0	1	+1

De absolute waarde van het negatieve getal uit B moet altijd kleiner zijn dan de waarde van het adres uit M (dus ook niet gelijk), want anders wordt het nummer van het adres negatief. Negatieve adressen bestaan niet en dienen te worden voorkomen door juiste programmering.

Onder deze voorwaarde ontstaat door optelling van het negatieve getal uit B bij het adres uit M altijd een overdracht in het adresgedeelte. Deze overdracht loopt door tot en met het meest significante CV-plaatje, omdat de betreffende OI-plaatjes een één vertegenwoordigen door signaal TBS'. Het resultaat is, dat bij het functiegedeelte der opdracht alleen nullen worden opgeteld: $(CV) + (TBS) = 1 + 1 = 0 +$ volgende carry. Het functiegedeelte blijft dus wederom onvervormd: $x + 0 = x$.

Op de besproken wijze verloopt alles goed, alleen vragen we ons af of we het een en ander niet iets kunnen bespoedigen. Dat is inderdaad mogelijk. Immers als het getal in het B-register negatief is, dan weten we zeker, dat de plaatjes CV14 t/m CV26 overdrachten zullen moeten vormen. Om wel de gewenste overdrachten beschikbaar te krijgen, maken we vast een niveau van plaatje CV14' en een niveau van plaatje CV26' laag met de signalen TBS' en ACR1. Daardoor worden de overdrachten in het functiegedeelte vast gevormd en tevens de overdracht, die bij de minst significante cijfers moet worden geteld. Waarom het signaal TBS' hier is aangesloten, is gezien het voorgaande duidelijk. Het signaal ACR1 bepaalt de ringstand, waarbij de snelle overdracht moet worden gevormd, want dat mag alleen gebeuren met de arithmetische controle ring in stand 1. Stand 0 is onbelangrijk, daar dan $(LB) = 0$ is. (Alleen bij het inschakelen van de X-1 is het niet zeker, dat signaal LB' hoog zal zijn, maar dan kan het ook geen kwaad.) Bij een B-register opdracht wordt het woord in B als een gewoon getal beschouwd en moet de snelle overdracht worden voorkomen. Dat gebeurt ook inderdaad, want gedurende de getalbewerking is $(ACR1) = 1$.

Op deze wijze komt de gecorrigeerde opdracht in het U-register te staan. Het adresgedeelte van het opdrachtregister neemt dan het betreffende deel van de opdracht uit het U-register over, waarna op grond van deze opdracht en op het aldus verkregen adres een woord uit het geheugen wordt gehaald. Verder verlopen de gebeurtenissen als bij de normale in-opdracht.

Niet additieve uit-opdracht (ongemodificeerd).

Bij een "niet additieve uit-opdracht" moet de inhoud van één van de interne registers in het geheugen worden geborgen, waarbij de oorspronkelijke inhoud van dat adres wordt vernietigd.

(Schoon-uit). De volgorde der gebeurtenissen, die plaatsvinden bij dit soort opdrachten, staat aangegeven in het middelste tijdschema op tek. XE10. Dat tijdschema zullen we nu bespreken.

Tot en met stand 3 van de geheugencontrole ring vinden dezelfde gebeurtenissen plaats als bij de normale in-opdracht of dezelfde als bij de in-opdracht met B-correctie, zodat deze niet weer behoeven te worden besproken.

Op 7gct wordt de geheugen controle ring in stand 4 gezet en nu kunnen alle organen in de juiste standen worden gezet op grond van de opdracht. De interne registers worden op de bekende wijze geselecteerd met behulp van signaal BAO'. Stel dat het A-register wordt gekozen. Bij een niet-additieve uitopdracht moet de opteller in stand 6 of 7 staan, omdat dan de inhoud van het (nu gekozen) A-register in register U wordt gezet via het arithmetische orgaan, ongeacht de inhoud van het M-register. De betreffende COI-niveau's worden laag door signaal BAO'. Ook hier bepaalt de inhoud van flip-flop OR21 of het getal positief of negatief moet worden beschouwd. Het betreffende A-register wordt negatief geïnterpreteerd, als $(OR21) = 1$.

COI6'	Signaal COI6 zet de flip-flops
1 Schoon uit BAO' NAUO' OR21	"Controle opteller" en "Controle carry" in de transportstand, waarbij de inhoud van het A-register zpnder meer wordt overgenomen. Dit is een niet-additieve-uit-opdracht: $(NAUO) = 1$. De inhoud van het A-register moet positief worden beschouwd: $(OR21) = 0$. Signaal BAO' maakt dit niveau laag op deze voorwaarden.

2 cijfertijden gereed is. Op 4 cijfertijd wordt niveau 2 van plaatje LPC' laag (zie blz. 65), zodat het gelijkheidscontrolecijfer door flip-flop M27 kan worden overgenomen en het complete woord in het geheugen kan worden geborgen als de geheugencontroleling in stand 7 staat. Deze ring komt op dezelfde wijze in stand 7 als bij de in-opdracht. Ook het verdere verloop is daaraan gelijk.

Bij de opdrachten 46 en 47 - (TSS) = 1 - wordt op 2 cijfertijd, als de arithmetische controleling in stand 3 staat (R3' is laag), niveau 9

	COT'	van plaatje COT' laag, waar-
9	Subr. R3' 2ct' OR22' OR24' OR25 OR26'	door de opdrachtsteller wordt
	sprong	- schoongemaakt en tevens wordt
4	tellende R3' 3ct' TSS' OR17 OR18	signaal OTIB' laag, zodat
	sprong	

de opdrachtsteller het adres uit het opdrachtregister overneemt en op dit adres de volgende opdracht laat halen. De oorspronkelijke

OTIB	inhoud van de opdrachtsteller wordt naar één van de acht
1 COT	speciale adressen (λ) in het geheugen gezonden om bewaard
2 TSS	te worden. De opdrachtsteller blijft dan geselecteerd
3 ACR1	(geen wijziging van het multiplet door signaal BAO'),
	zodat het U-register de inhoud van de opdrachtsteller

overneemt via het arithmetisch orgaan. Niveau 2 van plaatje COI6'

	OT7'/OT7	wordt daartoe laag gemaakt met de signalen
1	Houd OT7' COT	BAO', NAUO' en TSS', welke in dit geval op
3	Lees OR7' OTIB'	zichzelf duidelijk zijn. Dat speciale adres λ_m
		(m ter onderscheiding van de algemene adres-
		aanduidingen) wordt geselecteerd door de SG-
1	Houd OT7	plaatjes. Op de betreffende plaatjes wordt een

	COI6'	niveau laag gemaakt met behulp van de
2	Sprong BAO' NAUO' TSS'	signalen TSS', GCR2', OR19', OR20',
		OR21' en OR22'.

Met deze vier cijfers uit het functiegedeelte der opdracht worden de eerste 16 adressen (0 t/m 15) in het levend geheugen geselecteerd.

	SG0'		SG1'
4	TSS' GCR2' OR19'	4	TSS' GCR2' OR20'
	SG2'		SG3'
3	TSS' GCR2' OR21'	3	TSS' GCR2' OR22'

Bij opdracht 46 worden de adressen 8 t/m 11 geselecteerd en bij opdracht 47 de adressen 12 t/m 15 (zie rapport EL-1-N).

Het geheugen wordt nu dus niet geselecteerd op grond van het adresgedeelte der opdracht of de opdrachtteller, maar door een aantal functiecijfers. De adrescijfers uit het opdrachtregister of uit de opdrachtteller mogen de SG-plaatjes niet beïnvloeden. Daartoe zijn de signalen SOR' en SOT' nu hoog. Signaal SOT' is hoog, omdat signaal GCR2' nu laag is (zie blz. 51) en signaal

SOR	SOR' is nu hoog, omdat signaal TSS' laag is.
1 GCR2	Dit alles gebeurde na het laag gaan van niveau 9 van
2 TSS'	plaatje COT'. Echter als geldt: $(OR18) = (OR17) = 0$ en $(TSS) = 1$, kan ook niveau 4 laag worden op 3 cijfer-

tijd bij ringstand 3. Dit heeft geen ander gevolg, dan dat de opdrachtteller één cijfertijd langer gelegenheid krijgt om het adres uit het opdrachtregister over te nemen.

Additieve uit-opdracht (ongemodificeerd)

Bij dit soort opdrachten wordt het getal uit het geheugen vermeerderd of verminderd met een getal uit een intern register. De volgorde der gebeurtenissen, die bij deze opdrachten plaatsvinden, staat aangegeven in het onderste tijdschema op tek.XE10. Tot en met de overgang van stand 5 naar 6 van de geheugen controle ring verlopen de gebeurtenissen op dezelfde wijze als bij een in-opdracht. Ook de opteller komt op dezelfde wijze in stand 0 door signaal COIo (zie blz. 69). Signaal COI5 zet de opteller

COI5'	in stand 5. Niveau 1 van plaatje
1 Add.uit BAO' AUO' OR21'	COI5' wordt laag door signaal BAO' als deze opdracht een additieve-

uitopdracht is: $(AUO) = 1$ en het interne register negatief moet worden beschouwd: $(OR21) = 1$.

Aangezien nu het arithmetisch resultaat moet worden opgeborgen in het geheugen, zijn we genoodzaakt te wachten tot dat resultaat bekend is. Op 6gct is het getal in het M-register ter beschikking, daarna kan met de arithmetische bewerking worden begonnen. Vijf cijfertijden mag deze bewerking duren, want gedurende de 6e cijfertijd wordt het U-register ingelezen door de signalen CU en UI' (ct1b). Het inlezen van het U-register gebeurt op hetzelfde tijdstip ten opzichte van het signaal CCTR'

	CCTR'/CCTR					als bij de niet-additieve uit-
1 Houd	CCTR' Co'					opdracht. Nu echter gaat niveau 4
4 ACR 2-3	R2' 5ct' a' AUO' NBO'					van plaatje CCTR' laag op cijfer-
1 Houd	CCTR					tijd 5a', als de arithmetische
						controlering in stand 2 staat

(R2' is laag), deze opdracht een additieve-uit-opdracht is: (AUO) = 1 en (NBO) = 1. Dit laatste signaal is laag, als (OR19) = 0, d.w.z. geen B-modificatie, of als deze opdracht een tellende en subroutinesprong is: (TSS) = 1.

	NBO			Nadat het gelijkheidscontrolecijfer gereed is, kan
1 OR19'	TSS			het woord in het geheugen worden geborgen, dat
				gebeurt op dezelfde wijze als bij de niet-additieve-uit-opdrachten.

In deze verhandeling zijn drie niveau's van plaatje CCTR' aan de orde gekomen, waardoor de arithmetische controlering van stand 2 naar stand 3 gaat, nl. de niveau's 2,3 en 4. Op deze wijze van uitcoderen voor één handeling willen we hier de aandacht vestigen. Met de arithmetische controlering in stand 2 wordt eerst bekeken of er een CCTR-sigitaal moet komen op cijfertijd 7b, gaat dat niet door de verder gestelde voorwaarden, dan probeert men het op cijfertijd 5a', lukt ook dat niet, dan kan het signaal "altijd" ontstaan op cijfertijd 1b'. De voorwaarden op de eerste twee niveau's zijn vrij nauwkeurig bepaald, terwijl het laatste alleen stelt: "als niet...enz" en dus vrij algemeen is. Bij dit systeem behoeven dus niet alle voorwaarden op elk niveau opnieuw nauwkeurig te worden uitgedeerd, hetgeen "dure" niveau's vraagt. Alleen de voorwaarden voor het eerste niveau zijn precies bepaald. Waren de termen voor die voorwaarden niet aanwezig, dan kunnen op een latere cijfertijd andere voorwaarden worden gesteld, waarbij de eerste voorwaarden niet behoeven te worden uitgesloten en zo voort.

Bij de opdrachten 44 en 45 - (TSS) = 1 - kan op 3 cijfertijd, als de arithmetische controlering in stand 3 staat (R3' is laag) niveau 4,5 of 6 van plaatje COT laag gaan afhankelijk van de voorwaarden gesteld door de cijfers (OR17) en (OR18) (zie rapport EL-1-N). Signaal COT laat dan in samenwerking met signaal OTIB' de opdracht-teller het adresgedeelte uit het opdrachtregister overnemen. Door dezelfde signalen als bij de opdrachten 46 en 47 worden door de SG-plaatjes de adressen 0 t/m 7 in het levende geheugen (τ_m) geselecteerd.

Het getal, dat zich op het geselecteerde adres bevindt, wordt in het arithmetisch orgaan gebracht. De opteller wordt door

COI5'	signaal COI5 in de optelstand gezet,
2 Sprong BAO' AUO' TSS'	waarbij het interne register negatief wordt beschouwd. Daartoe wordt

niveau 2 van plaatje COI5' laag gemaakt met de signalen BAO', AUO' en TSS', welke op zichzelf reeds duidelijk zijn. De opdrachtteller blijft geselecteerd, doordat signaal BAO' geen wijziging veroorzaakt in het multipliet. Door een speciaal niveau op plaatje LOT (onaccent!) wordt signaal LOT' eveneens hoog gemaakt, zodat nu de vier multipliet-signalen hoog zijn en

2 tellende sprong	LOT ACR1' TSS' OR22	<u>geen</u> intern register door het circuit OI wordt gekozen. De signalen op dit niveau zijn de uit-
-------------------	---------------------	---

codering van de arithmetische controlering in de standen 2 en 3:

(ACR1) = 1 en verder van deze beide opdrachten: (TSS) = 1 en (OR22) = 1. Deze zelfde signalen maken niveau5 van plaatje OIo' laag, zodat het schijnt of circuit OI de opdrachtteller met

5	OIo' ACR1' TSS' OR22	inhoud 1 selecteert. Dit getal wordt negatief bij het getal van geheugenplaats τ geteld. Daarna wordt dan weer het gelijk-
---	----------------------	---

heidscontrolecijfer gemaakt en het gehele woord in het geheugen geschreven op de bestemde plaats.

B-gemodificeerde opdrachten

De getalcyclus bij dit soort opdrachten verloopt overeenkomstig die bij de ongemodificeerde opdrachten, alleen komt de arithmetische controlering op andere cijfertijden in stand 3. Dat komt doordat de opdrachtcyclus hier langer duurt, dan bij de ongemodificeerde opdrachten. Het is evenwel niet ingrijpend en daarom niet belangrijk. Het signaal CCTR' kiest op de bekende wijze het juiste moment daarvoor.

Bij de opdrachten 44 t/m 47 is modificatie onmogelijk, daar de cijfers (OR19) en (OR20) de B-modificatie specificeren en dus niet tegelijkertijd benut kunnen worden om het adres τ of λ te selecteren. De opdrachtcycli in de drie tijdschema's op tek. XE11 zijn gelijk, afgezien van de instelling van de opteller.

In de eerste plaats moet nu vanzelfsprekend signaal MB' laag worden. Dat gebeurt onder voorwaarde dat: (TSS) = 0, reeds

	MB'	besproken, (SKIP) = 0 wordt later
1	TSS SKIP FOUT OR19' OR20'	behandeld, (FOUT) = 0 vanzelfsprekend en (OR19) = (OR20) = 1.

De laatste twee signalen zijn zoals gezegd de uitcodering van de B-modificatie. De B-gemodificeerde opdracht is een bijzondere vorm van de opdracht met B-correctie. Ook signaal BC' zal dus laag worden op de wijze zoals is besproken op blz. 75. De opdracht gaat op dezelfde wijze door het arithmetisch orgaan als bij de B-correctie. Daar kon echter de opdracht al worden teruggezonden naar het geheugen, terwijl het gecorrigeerde woord nog maar net in het U-register stond. Nu evenwel moet eerst het M-register dat woord uit U overnemen, daarna moet het controlecijfer nog worden gevormd en ingelezen (LPC). Signaal CM' wordt laag op cijfertijd 6b' als de arithmetische controle-ring in stand 1 staat (R1' is laag) en signaal MB' laag is.

	CM'	Het uitgangssignaal CM maakt het
2	Gemod.B. R1' 6ct' b' MB'	M-register schoon, terwijl signaal MI' laag wordt om het woord in dit register in te lezen op de bekende wijze.

Na dit alles mag de opdracht pas terug naar het geheugen.

	GCRo'/GCRo	Daarom mag de geheugencontrole-ring
1	Houd GCRo'	pas in stand 3 worden gezet als het controlecijfer gereed is. Signaal MB is hoog en daarom kan niveau 3 van plaatje GCRo (blz. 57) nu <u>niet</u> laag worden op 0gct, maar wordt
1	Houd GCRo	
2	2-3 GCR1' GCR1 Oct'	

niveau 2 laag op Oct als de signalen GCR1' en GCR2 laag zijn (GCR in de standen 2 en 3). Deze beide niveau's zijn een voorbeeld van het besprokene op blz. 83, gaat niveau 3 niet laag om de gestelde voorwaarde, dan gaat niveau 2 enige cijfertijden later.

Verder behoeven deze tijdschema's geen nadere verklaring.

Fout opdracht

De gebeurtenissen, die plaatsvinden, wanneer een opdracht fout wordt gelezen uit het geheugen zijn grotendeels reeds beschreven (blz. 68). Nu willen we daarop nader ingaan aan de hand van het tijdschema op tek.XE12 (rechtsboven).

Op cijfertijsd 4a' wordt het leesniveau van plaatje FO' laag tengevolge van het gestelde nl. dat signaal FOUT' laag is, zodat de flip-flop STOP in stand 1 wordt gezet en de X-1 stopt. Ondertussen is de geheugencontroleling reeds in stand 3 gekomen op de normale wijze. In het normale geval zou de geheugencontroleling op 7get in stand 4 worden gezet. Dat moet worden voorkomen. Daartoe wordt niveau 2 van plaatje GCR1 laag gemaakt op 6get

	GCR1'/GCR1				door de signalen GCRo, GCR2 en
1 Houd	GCR1'				GG om de flip-flop GCR1 terug
1 Houd	GCR1				te stellen en daarmee de geheu-
2 3-0	GCRo	GCR2	6get'	GG	gen controleling in stand 0 te
					zetten. Immers in stand 3 geldt
					reeds (GCRo) = (GCR2) = 0.

Signaal GG is laag, omdat het niveau van plaatje GG' hoog wordt gemaakt door signaal FOUT, want er mag nu geen getal uit het geheugen worden gehaald.

Op cijfertijsd 5a wordt niveau 2 van plaatje CHB laag, omdat signaal R1' laag is en (STOP) = 1. Daardoor worden de HB-flip-flops teruggesteld, waardoor weer signaal HBS wordt hoog gemaakt. Daarmee is voorkomen dat met niveau 5 van plaatje GCRo' de geheugencontroleling in stand 1 kan worden gezet (zie blz. 48) door signaal KO'. De niveau's 3 en 4 zijn eveneens hoog door signaal STOP. Ook hiermee kan stand 1 dus niet worden veroorzaakt.

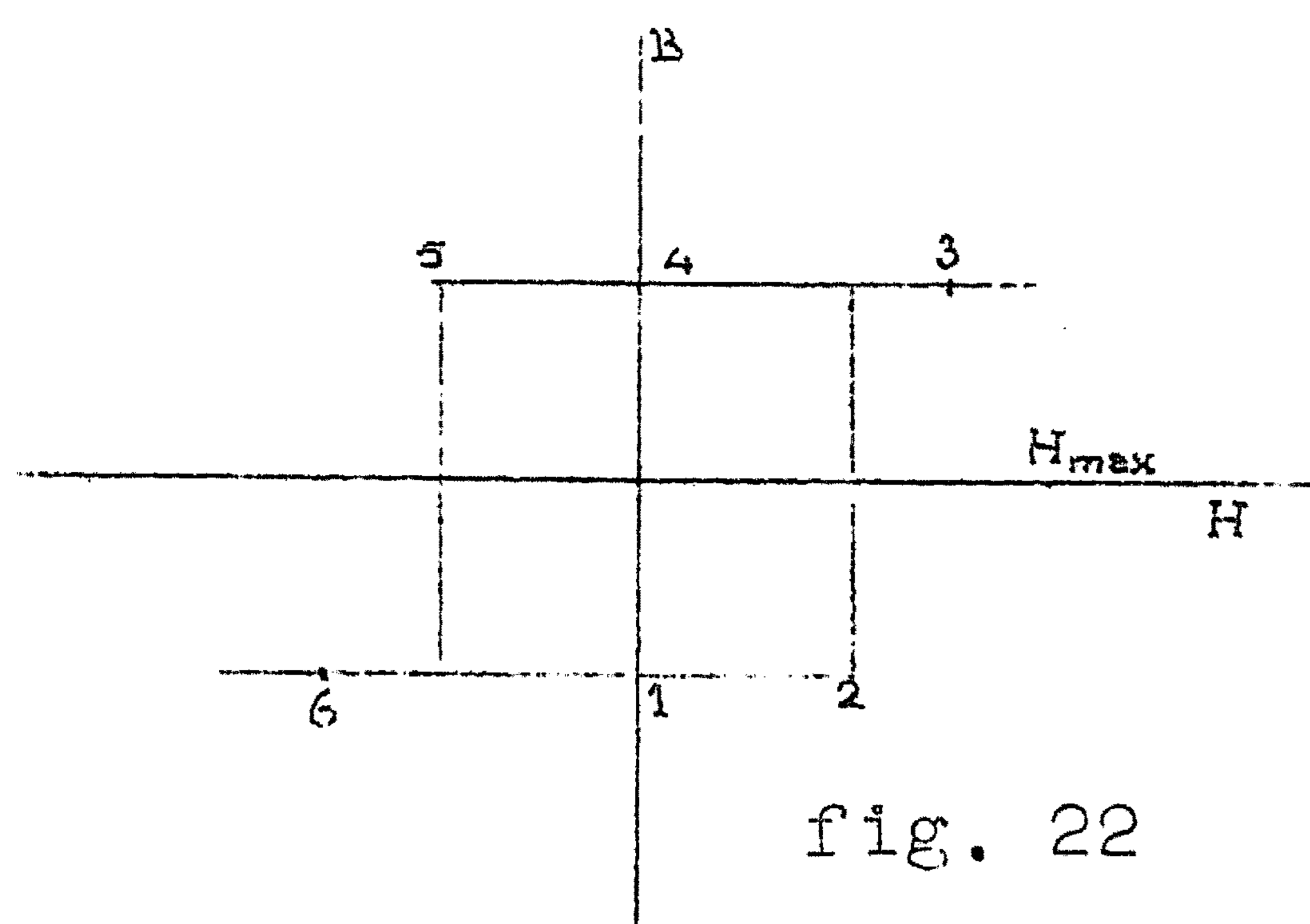
Op cijfertijsd 6b wordt de arithmetische controleling in stand 0 gezet (ACR1 is laag). Zowel de geheugencontroleling als de arithmetische controleling staan nu in stand 0, terwijl ook de HB-flip-flops zijn teruggesteld. De X-1 kan nu slechts op gang worden gebracht door op één van de operationele of decimaalknopjes te drukken, zoals is beschreven bij de handbediening.

	ACRo'/ACRo				gezet (ACR1 is laag). Zowel de geheugen-
1 Houd	ACRo'				controleling als de arithmetische
1 Houd	ACRo				controleling staan nu in stand 0,
3 1-0	ACR1	6ct'	b		terwijl ook de HB-flip-flops zijn
					teruggesteld. De X-1 kan nu slechts
					op gang worden gebracht door op één

De opbouw van het levend geheugen

Al eerder is er gesproken over het levend en het dood geheugen. Dit onderscheid wordt gemaakt om te doen uitkomen, dat de gegevens in het levend geheugen door de X-1 zijn te veranderen, terwijl de gegevens in het dood geheugen vastliggen en niet door de X-1 zijn te veranderen.

De werking van het geheugen is gebaseerd op de eigenschappen van magnetische kernen (ferriet). Deze eigenschappen zullen we in het kort herhalen. In fig. 22 is de geïdealiseerde hystereselus getekend. In rust kunnen de kernen zich in toestand 1 of 4



bevinden. Na het aanleggen van een veld H_{max} , dat de grootte heeft van het stukje 3-4 (fig.22) is de toestand van de kern veranderd van 1 via 2 en 3 naar 4, of van 4 via 5 en 6 naar 1, afhankelijk van de richting van H_{max} .

Bevindt de kern zich in toestand 1 en wordt een veld $-H_{max}$ toegepast, dan blijft de kern in toestand 1. De spanning, die daarbij op de uitgangswikkeling te meten is, noemt men wel shuttle-voltage of storing. Bevindt de kern zich in toestand 4, dan treedt een grote fluxverandering op en geeft de uitgangswikkeling een grote spanningspuls te zien. In het eerste geval bevatte de kern een nul en in het tweede geval een 1.

Nu zullen we de toepassing in het geheugen bespreken. Om één woord van 28 cijfers in het geheugen te bewaren zijn dus 28 kernen nodig. Om technische redenen worden de kernen voor de overeenkomstige cijfers van 512 woorden in één zgn. "cijfervlak" gevlochten, zoals in fig. 23 is aangegeven. Er zijn 64 verticale lijnen en 8 horizontale. Voor 512 volledige woorden zijn dus 28 cijfervlakken nodig. Deze cijfervlakken zijn op één raam gemonteerd. Zo'n raam wordt een plaat genoemd en heeft aan elke zijde 14 cijfervlakken. Een geheugenkast bevat 8 van deze platen, dat zijn dus 4096 adressen. Voor vier kasten bestaat de mogelijkheid om op de basismachine te

worden aangesloten. In totaal zijn er dus 16384 adressen in het levend geheugen.

De leeslijn (uitgangswikkeling) wordt door de kernen heengevlochten op de wijze, zoals in fig. 23 is aangegeven.

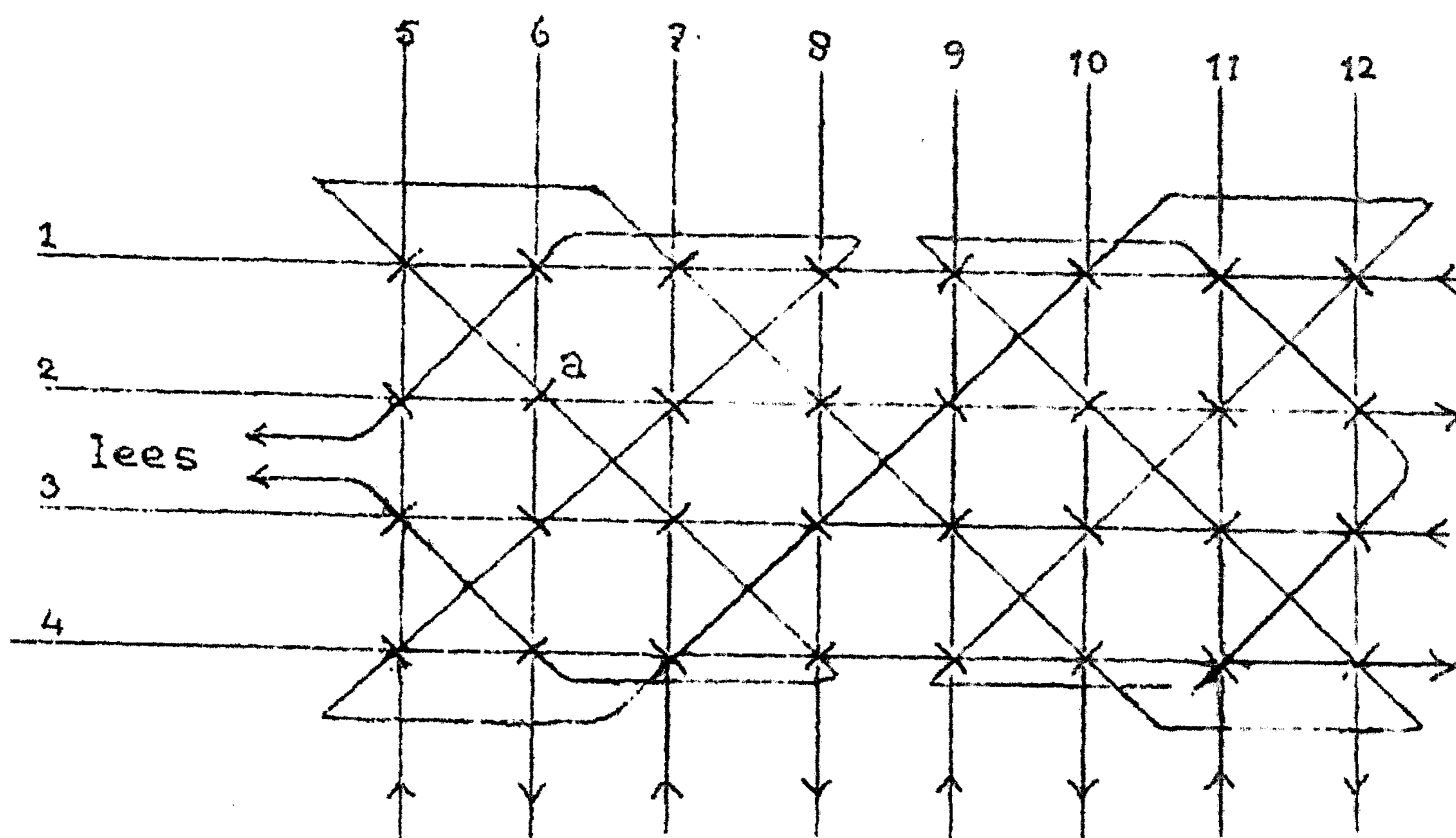


fig. 23

De uiteinden van de leeslijn geven het signaal voor de leesversterker. Om b.v. kern a uit te lezen wordt door het horizontale lijntje 2 een stroom $\frac{1}{2}i_{\max}$ gestuurd, waarbij $\frac{1}{2}i_{\max}$ in staat is een veld $\frac{1}{2}H_{\max}$ op te wekken. Dit veld is te klein om één van de kernen, waardoor deze stroom loopt te doen omgaan. Wordt ook door lijntje 6 een stroom $\frac{1}{2}i_{\max}$ gestuurd, dan is voor alle kernen waar lijntje 6 door loopt het veld te zwak, behalve voor kern a. Bevatte de kern a een nul, dan verschijnt er geen puls op de leeslijn, afgezien van de storing. Bevatte de kern een één, dan ontstaat er een duidelijke puls door de fluxverandering in de kern. Door het lezen wordt de kern in de toestand gebracht, waarbij hij een nul bevat. Vandaar dat een gelezen woord ook weer moet worden teruggeschreven, wanneer het bewaard moet blijven (destructief lezen). De benodigde stroom i_{\max} hangt van verschillende factoren af en is iets minder dan 400mA.

Op een horizontale lijn zitten 64 kernen en als elke kern een klein stoerpulsje geeft, wordt dat bij elkaar een grote stoerpuls. Daarom wordt door de horizontale drijflijn 4μ sec eerder stroom gestuurd dan door de verticale drijflijn. Bovendien

wordt het gelezen signaal gedrempeld en gepoort, zodat de stoorsignalen, die niet boven het drempelniveau uitkomen ook niet aan de uitgang van de versterker verschijnen en de stoorsignalen, die buiten de tijdsduur van de poortspanning vallen, er ook niet doorkomen. Het uitgangssignaal ligt in de grootorde van 40mV.

De stroom door de lijnen wordt gestuurd door zg. versterkers, die de functie van een schakelaar hebben. Afhankelijk van de signalen, die op de ingang worden aangesloten, gaat de eindtransistor geleiden of niet. Dat is dus hetzelfde principe als bij de T-eenheden (fig. 24).

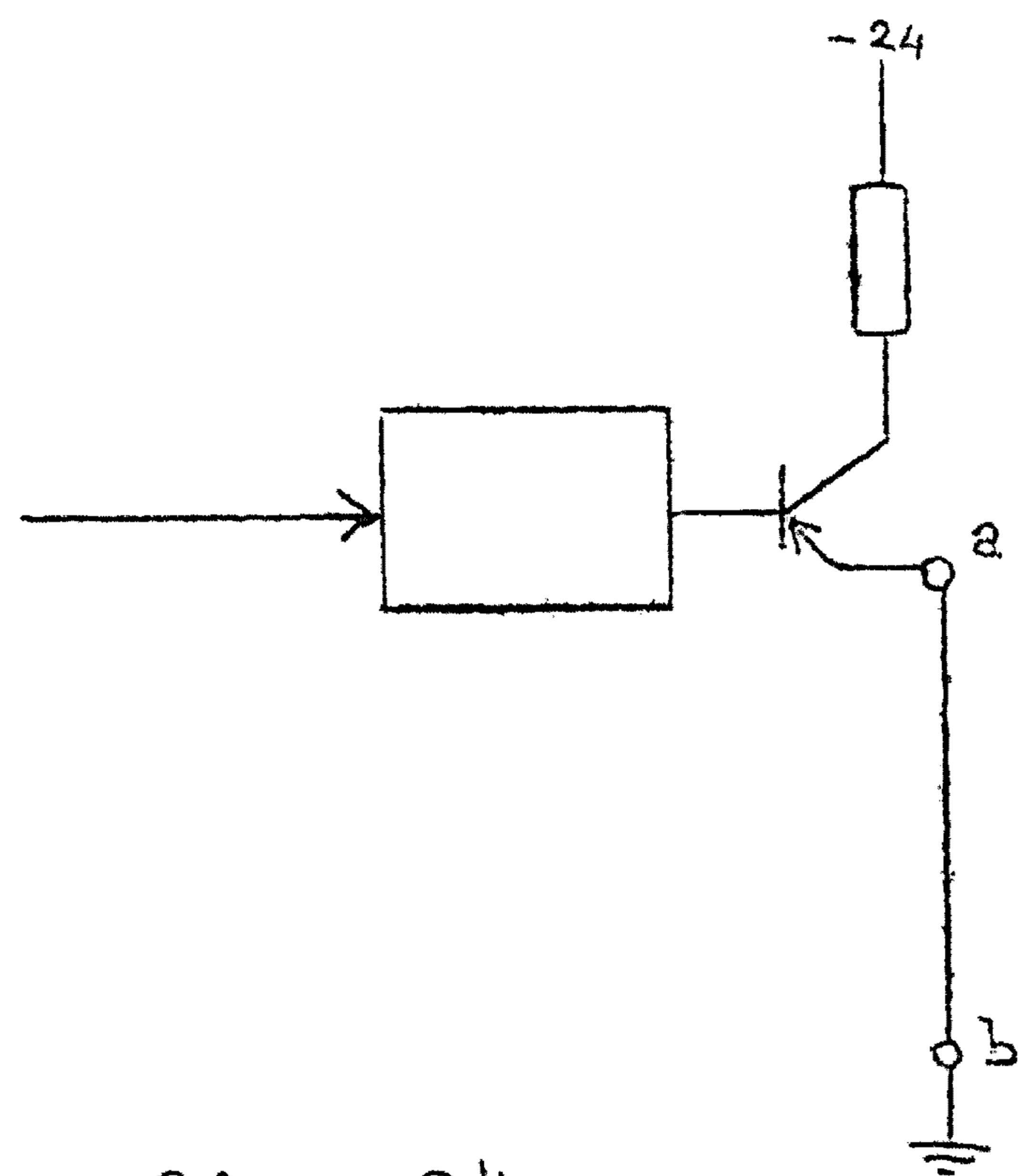


fig. 24

transistor deze doet geleiden, loopt er stroom door de leeslijn ab, en als de transistor niet geleidt, loopt er natuurlijk geen stroom. Op deze wijze geschakeld zou er voor elke lijn een versterker nodig zijn. Daarom wordt een ander systeem toegepast, dat in fig. 25 is getekend. Met drie versterkers onder en drie boven komen negen lijnen ter beschikking. Met acht versterkers onder en acht boven komen

er 64 beschikbaar. Dat wordt voor de verticale lijnen toegepast. De eindtransistoren zijn hierbij geschakeld volgens hetzelfde principe, als bij de carryvormers (blz. 41).

Wanneer de transistors van de versterkers a en b gaan geleiden, zal er een stroom lopen van b naar a, maar eveneens van b, via c en d naar a en zo via alle verdere versterkers, wanneer dit niet met dioden wordt voorkomen. De dioden worden geschakeld, zoals in fig. 25 is aangegeven. Nu zou de stroom kunnen lopen van b naar c, maar van c naar d is onmogelijk, want dan moet de stroom de diode in sperrichting doorlopen. Door de juiste versterkers te laten werken, wordt een gewenste lijn geselecteerd.

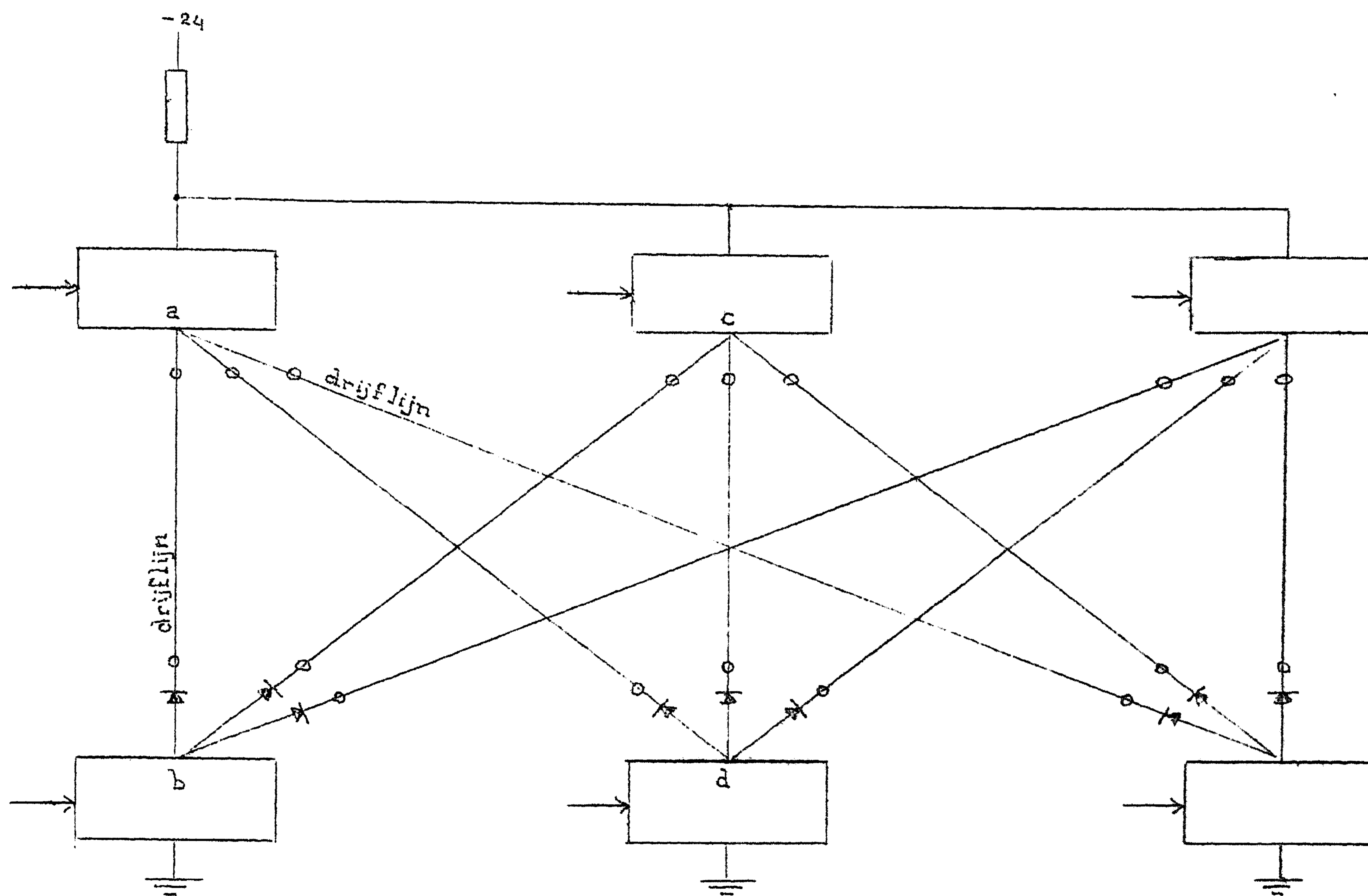


fig. 25

De horizontale lijnen worden op dezelfde manier geselecteerd. Aan de ene zijde van een cijfervlak bevinden zich acht versterkers, aan de andere zijde is er één. Dat schijnt niet economisch te zijn, maar tot nu toe spraken we alleen over de drijfstroom om te lezen. Echter ook om te schrijven is drijfstroom nodig. Deze laatste stroom moet evenwel tegengesteld gericht zijn t.o.v. de eerstgenoemde. Ook om te schrijven zijn dus 64 verticale en 8 horizontale lijnen nodig. Om de horizontale drijflijnen stroom te laten voeren respectievelijk bij lezen en schrijven, worden de acht versterkers gemeenschappelijk gebruikt. Eén is er verder nog nodig om te lezen en één om te schrijven. Totaal zijn er dus 16 horizontale lijnen. Ook het aantal onderste versterkers wordt verdubbeld. Deze twee groepen van acht versterkers worden weer geselecteerd door twee versterkers respectievelijk voor selectie van de leesgroep en de schrijfgroep (zie tek. X1-E19).

Terwille van het overzicht hebben we het doen voorkomen, alsof de besproken serie versterkers bij één cijfervlak behoorden. Dat is echter niet waar. Met behulp van één serie versterkers wordt een heel woord gelezen of geschreven, zodat elke plaat een complete serie versterkers heeft. De drijfliijnen lopen dus door alle cijfervlakken van één plaat. De horizontale drijfliijn loopt dus door $64 \times 28 = 1792$ kernen. De verticale drijfliijn door $8 \times 28 = 224$ kernen.

Per kast is er één serie lees- en één serie schrijfversterkers. De leesliijn zowel als de schrijfliijn doorlopen alle overeenkomstige cijfervlakken van de acht platen in een kast. De leesliijn doorloopt dus $8 \times 512 = 4096$ kernen.

De schrijfliijn loopt in horizontale richting door alle kernen, dus b.v. in fig. 23 door de bovenste rij kernen naar rechts en door de rij daaronder naar links en zo voort (niet getekend). De schrijfliijn voert alleen stroom als een nul in de geselecteerde kern moet worden geschreven, d.w.z. de kern mag niet omgaan, want bij het lezen werden alle geselecteerde kernen in stand 0 gezet. De stroom door de schrijfliijn is zo gericht, dat deze het veld opheft, dat door een horizontale drijfliijn wordt veroorzaakt. Het gevolg is, dat het veld in de geselecteerde kern slechts $\frac{1}{2}H_{\max}$ is (van de verticale drijfliijn), zodat de kern niet omgaat en dus "een nul blijft bevatten". Was de stroom door de schrijfliijn nul, dan was de geselecteerde kern wel omgegaan door de horizontale en verticale drijfstromen en bevatte dan een één. De schrijfliijn doorloopt $8 \times 512 = 4096$ kernen. Wanneer er een stroom door de schrijfliijn loopt, omvat de geselecteerde kern dus een veld $\frac{1}{2}H_{\max}$. Alle kernen van dit cijfervlak op deze geselecteerde horizontale en verticale drijfliijnen omvatten een veld 0, aangezien de velden van deze drijfliijnen daar ter plaatse door het veld van de schrijfliijn worden opgeheven. Alle overige kernen, waar één van de drie liijnen doorgaan, omvatten dus een veld $\frac{1}{2}H_{\max}$.

Versterkers

Er zijn drie verschillende soorten selectieversterkers, die worden onderscheiden door cijfers en kleuren. Het meest significante cijfer duidt op het type en het minst significante cijfer geeft het aantal ingangen van de versterker aan.

1. a. de versterker Vo1 oranje
- b. de versterker Vo2 oranje
- c. de versterker Vo3 oranje
2. de versterker V11 violet
3. de versterker V21 helder rood.

Verder zijn er nog versterkers, die de gelezen signalen verwerken. Deze zullen later worden behandeld.

Zoals reeds werd opgemerkt komt de werking van genoemde selectieversterkers overeen met de werking van de T-eenheden. Er zijn weer een of meer ingangsdioden, waarop signalen worden aangesloten, die op de bekende wijze het spanningsniveau van de ingang hoog of laag maken. Daar de eindtransistoren een grote stroom moeten leveren (200mA) en niet elke emitter aan aarde ligt, moeten hier bepaalde voorzieningen worden getroffen, die bij de T-eenheden niet nodig waren.

Van de Vo-versterkers liggen de emittoren aan aarde of hebben slechts een klein spanningsverschil ten opzichte van aarde, wanneer ze zich in geleidende toestand bevinden. Ze worden gebruikt als versterkers, die geleiden afhankelijk van de tijdsignalen, of als versterkers, die stroom moeten sturen door de schrijflijnen, of als versterkers voor de selectie van de drijflijnen, waarbij ze aan de "aardzijde" (onderzijde) van deze lijnen worden aangesloten.

De V11-versterkers zijn aangesloten aan de "spanningszijde" (bovenzijde) van de verticale en horizontale drijflijnen door het levend geheugen en aan de "spanningszijde" van de horizontale drijflijnen door het dood geheugen.

De V21-versterkers bevinden zich aan de "spanningszijde" van de gevlochten (verticale) drijflijnen door het dood geheugen. Deze drijflijnen worden later behandeld.

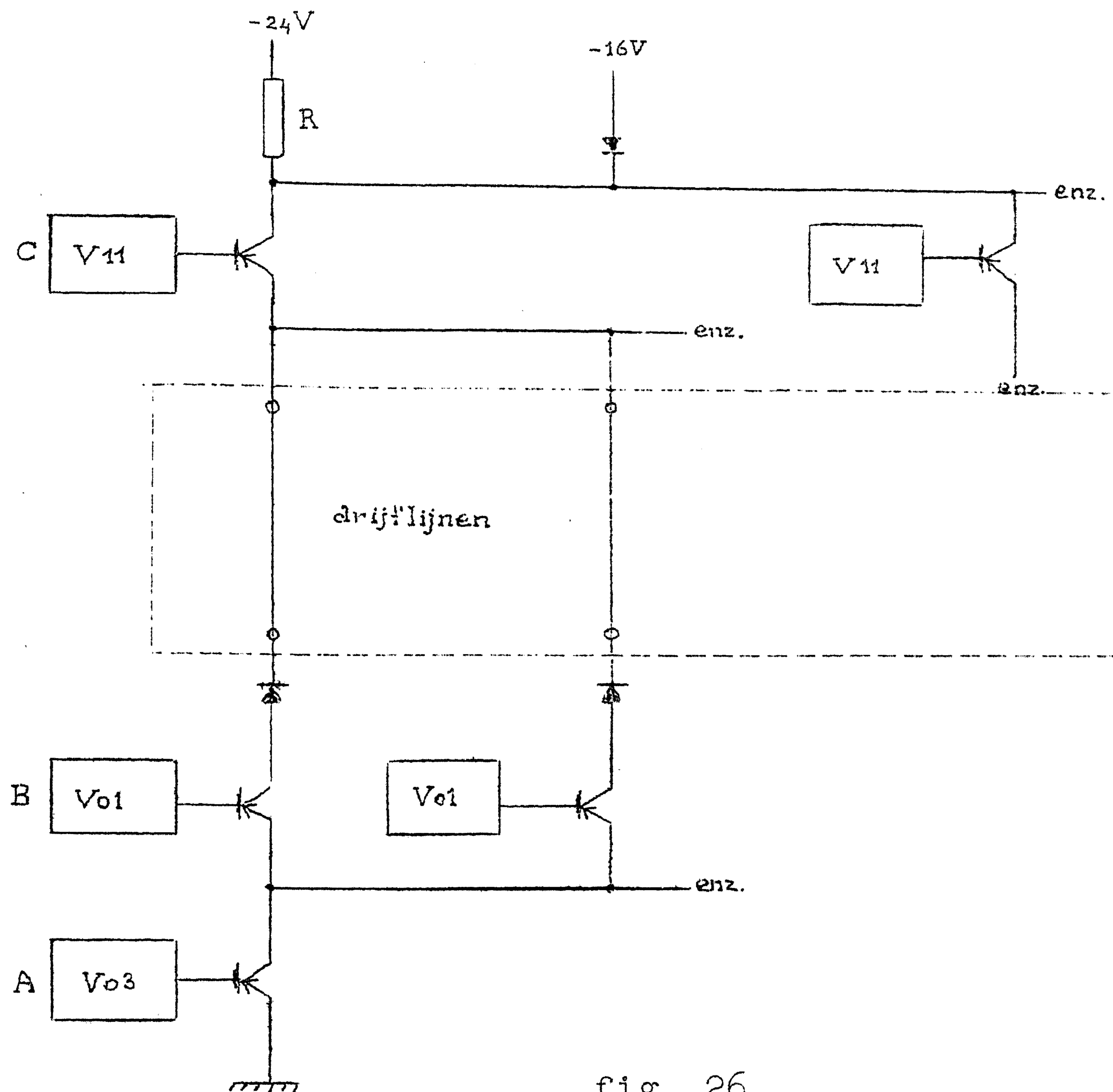


fig. 26

In fig. 26 is een detail getekend van de bedrading der versterkers. Stel dat de versterkers A, B en C zich in geleidende toestand bevinden. De eindtransistor van versterker A ligt met de emitter aan aarde, terwijl de eindtransistor van versterker B de kleine weerstand van de geleidende transistor uit versterker A in de emitterleiding heeft. Deze weerstand is te klein om speciale voorzieningen voor de basisspanning van de transistor uit versterker B te treffen. De eindtransistor van versterker C evenwel heeft de volgende weerstanden in serie in zijn emitterleiding: de weerstand van de "lange" drijflijn, de weerstand van de geleidende diode, de weerstanden van de eindtransistoren uit B en A. Al deze weerstanden tezamen worden doorlopen door een vrij grote stroom, zodat de spanning die daarover valt, reeds van invloed is. Vandaar dat versterker C er een is van een speciaal type, nl. V11 in verband met de noodzakelijke spanningsinstelling aan de basis van zijn eindtransistor.

Bovendien bestaat de mogelijkheid, dat vele malen achter elkaar dezelfde plaats in het (levend) geheugen wordt geselecteerd. De bovenste versterker wordt zowel bij het lezen als bij het schrijven gebruikt, zodat deze dan vrijwel continu in bedrijf is. Daarom zijn daarin twee transistoren parallel geschakeld.

In de collectorleiding van versterker C is een weerstand geschakeld, die dient ter bepaling van de stroom, daar anders de stroom alleen werd bepaald door de spanning en de weerstand van de diode en de transistoren. De stroom behoort 200mA te zijn, zodat de totale weerstand $120\ \Omega$ moet zijn bij de spanning van 24 Volt. Aangezien niet alle transistoren en dioden dezelfde weerstand bezitten, zou de bijgevoegde weerstand R van geval tot geval moeten worden berekend. Men heeft echter het gemiddelde genomen en de weerstand gemeenschappelijk gebruikt voor alle collectoren van de bovenste versterkers.

Nu mag de hier gebruikte transistor slechts een bepaalde maximum spanning tussen de collector en de emitter hebben; 24V is te groot en 16V valt binnen de grenzen. Aangezien de -16V beschikbaar is, leggen we die via een diode aan de collectoren (zie fig. 26). Zodra nu de spanning aan de collectoren lager (negatiever) wordt dan -16V, gaat de diode geleiden en houdt dat punt op -16V.

De eindtransistor van de versterker V21 heeft een andere instelling nodig, dan die van versterker V11 in verband met de zeer lange (gevlochten) drijflijn.

Selectie signalen voor het geheugen

Zoals reeds werd opgemerkt kan het geheugen worden geselecteerd,

- a) op grond van het adres, dat door de opdrachtsteller wordt gegeven,
- b) op grond van het adresgedeelte der opdracht,
- c) op grond van enkele gespecificeerde signalen.

Deze selectie gebeurt met behulp van de SG-plaatjes.

In het eerste geval moet signaal SOT' laag zijn. Dat is het geval gedurende de opdrachtcyclus, als de signalen GCR2' en HB1' hoog zijn (zie blz. 51). In geval b moet signaal SOR' laag zijn, dat is het geval gedurende de getalcyclus als de

SG7' signalen GCR2 en TSS' hoog zijn (zie blz. 82).
 1. OR4' SOR' De SG-signalen worden verdeeld in vijf groepjes
 2. OT4' SOT' van drie. Met de drie minst significante cijfers
 worden de signalen VS00' t/m VS07' gevormd.

Met de daarop volgende drie de signalen VSBo' t/m VSB7' en met de 6e, 7e en 8e cijfers de signalen HSo' t/m HS7'. De 9e, 10^e en 11e cijfers vormen de signalen PSo' t/m PS7'. Met de laatste groep worden de signalen KSo' t/m KS3' gevormd voor selectie van het levend geheugen en de signalen DKSo' t/m DKS3' voor het dood geheugen.

Tijdsignalen

In de standen 1 en 5 van de geheugencontrolling moet het mogelijk zijn uit het geheugen een opdracht respectievelijk een getal te halen. Dat gebeurt met behulp van de signalen DL10' en DL14'. De cijfers bij deze signalen geven de tijdsduur aan in microseconden.

	DL10'		DL14'
1	GCRo' GCTR2'	1	GCRo' GCTRo'
2	GCRo' GCTR1' GCTRo	2	GCRo' GCTR1'
		3	GCRo' GCTR2' .

Het signaal GCRo' is laag in de standen 1,2,5 en 6 van de geheugencontrolling. Dat signaal is aangesloten op alle niveau's van de plaatjes DL10' en DL14'. Deze niveau's kunnen dan laag worden. De standen 2 en 6 zijn hieronder begrepen, hoewel er dan meestal niet wordt gelezen. De geheugen cijfertijdring kan dan niet gaan werken, want deze cijfertijdring begint alleen te tellen in de oneven standen van de geheugencontrolling, hoewel hij wel kan uitlopen in een even stand (zie blz. 31), zodat deze tijdsignalen wel eens daarbinnen vallen.

Op het eerste niveau van plaatje DL14' is het signaal GCTRo' aangesloten. Dit niveau is dus laag gedurende 1 en 2 gct en ook gedurende 5 en 6 gct (zie blz. 27). Op het tweede niveau komt signaal GCTR1' en dat is laag van 2 t/m 5 gct, het derde niveau wordt laag door signaal GCTR2', dat is van 4 t/m 7 gct. Deze signalen overlappen elkaar, zodat het signaal DL14 hoog is na 0gct tot en met 7 gct, dus gedurende 14 μ sec.

Op dezelfde wijze ontstaat signaal DL10. Van plaatje DL10' is het eerste niveau laag van 4gct t/m 7gct door signaal GCTR2' en het tweede niveau gedurende 3 en 4 gct door combinatie van de signalen GCTR1' en GCTR $\bar{0}$ op dit niveau. Signaal DL10 duurt van 3 t/m 7 gct en is dus 10μ sec lang. Beide signalen eindigen gelijktijdig. Zoals we reeds gezien hebben moeten de horizontale drijflijnen eerder stroom voeren dan de verticale in verband met de storing. Signaal DL14 wordt dan ook gebruikt voor de horizontale drijfstromen en signaal DL10 voor de verticale drijfstromen.

In de standen 3 en 7 van de geheugencontroleling moet de mogelijkheid bestaan een woord in het geheugen op te bergen. Hiervoor worden de signalen van de plaatjes DS10' en DS12 gebruikt. In de standen 0-3-4-7 van de geheugencontroleling is $(GCR_0) = 0$. Ook nu zijn de even standen weer hieronder begrepen om dezelfde reden als in het voorgaande geval.

	DS10'		DS12
1	GCR_0 GCTR1'		1 GCR_0 '
2	GCR_0 GCTR2' GCTR $\bar{0}$ '		2 GCTR $\bar{0}$ GCTR1

Niveau 1 van plaatje DS10' is laag, als $(GCR_0) = 0$ en als signaal GCTR1' laag is. Dat is van 2 t/m 5 gct. Het tweede niveau is laag gedurende de tijd, dat de signalen GCTR2' en GCTR $\bar{0}$ ' tegelijkertijd laag zijn, dat is dus gedurende 5 gct en 6 gct. Signaal DS10' is dus gedurende 10μ sec laag en wel van 2 gct t/m 6 gct. Signaal DS12' is laag als signaal GCR_0 ' hoog is en één van de signalen GCTR $\bar{0}$ of GCTR1 hoog is. Dit geeft een totaal tijd van 1 t/m 6 gct of wel 12μ sec. Signaal DS10' doet de stroom door de horizontale en verticale drijflijnen lopen, terwijl signaal DS12' de stroom door de schrijflijn veroorzaakt. De stroom door de schrijflijn moet voorkomen, dat een kern in de stand 1 wordt gezet en komt daarom 2μ sec eerder. Immers zou de stroom door de schrijflijn worden veroorzaakt door hetzelfde signaal, als de stroom door de drijflijn, dan bestaat de kans in verband met eventuele vertragingen, dat het veld van de drijflijnen eerst de kern omzet en daarna pas wordt gehalveerd door het tegenveld van de schrijflijn. De kern is dan ten onrechte omgegaan en deze fout kan niet door het veld van de schrijflijn ongedaan worden gemaakt.

De signalen, die uit de SG-signalen ontstaan en de DL- en DS-signalen worden in de basismachine gevormd en dan naar de geheugenkast gevoerd. In de geheugenkast worden deze signalen verder verwerkt, zoals in het voorgaande reeds is aangegeven.

Selectiesignalen in de geheugenkasten

De SG-signalen worden dus in de basismachine al omgecodeerd in een aantal signalen, waarmee het juiste woord in het geheugen wordt geselecteerd. De aldus verkregen signalen worden over de diverse geheugenkasten verdeeld en daar omgekeerd en verder verwerkt. Bovendien wordt hierbij afgeweken van de aantallen ingangen, die door een eenheid mogen worden gevoed. Vandaar dat in de geheugenkasten meer E-eenheden worden toegepast bij een lager aantal te voeden ingangen.

Het magnetisch geheugen is ondergebracht in vier kasten. In kast 0 bevinden zich acht platen levend geheugen en het gehele dood geheugen. In elk van de overige drie kasten bevinden zich alleen de acht platen levend geheugen.

Bij nadere beschouwing van tek. X1-E18 blijkt, dat van elk selectiesignaal vier onderling gelijkwaardige signalen worden gemaakt, nl. voor elke kast de eigen signalen. Die signalen worden dan ook alleen onderscheiden door het betreffende kastnummer voor de signaalsymbolen te plaatsen. De signalen voor het dood geheugen worden onderscheiden door een "D" voor de symbolen. Alle selectiesignalen voor het dood geheugen, behalve de plaat-signalen, worden ook verkregen alleen door omkering. De laatste ontstaan door combinatie van de vier minst significante met de vier meest significante oorspronkelijke plaat-signalen (PS).

Uit de tekening blijkt verder, dat de KS- en CI-signalen worden onderscheiden door achtervoeging van de letters A, B en C.

Dit geeft geen principiële onderscheiding. De overigens gelijkbenoemde signalen hebben dezelfde functie. De oCI-signalen bijvoorbeeld worden alle naar dezelfde serie schrijfversterkers gezonden (zie tek. X1-E19), maar van de $\frac{1}{2}$ E-eenheden mag één uitgangssignaal slechts 10 ingangen van de versterkers voeden.

Bij het levend geheugen zijn 4096 plaatsen (adressen) in één kast ondergebracht. Naar analogie hiermee wordt gezegd, dat het dood geheugen ook uit vier "kasten" bestaat. Het woord "kast" duidt hier alleen op een aantal (4096) plaatsen in het dood geheugen. De 4096 woorden in het dood geheugen zijn op één raam gemonteerd. In kast 0 bevinden zich dus vier van zulke ramen. Elke kast dood geheugen heeft een eigen complete serie versterkers.

De opbouw van het dood geheugen

De toepassing van de magnetische kernen in het dood geheugen berust alleen op het principe, dat er een puls ontstaat op de leeslijn, wanneer de kern wordt omgezet door een drijfstroom. De draad, waardoor de drijfstroom loopt, wordt door een kern gevlochten, als deze een één en langs de kern, als deze een nul moet produceren (zie fig. 27).

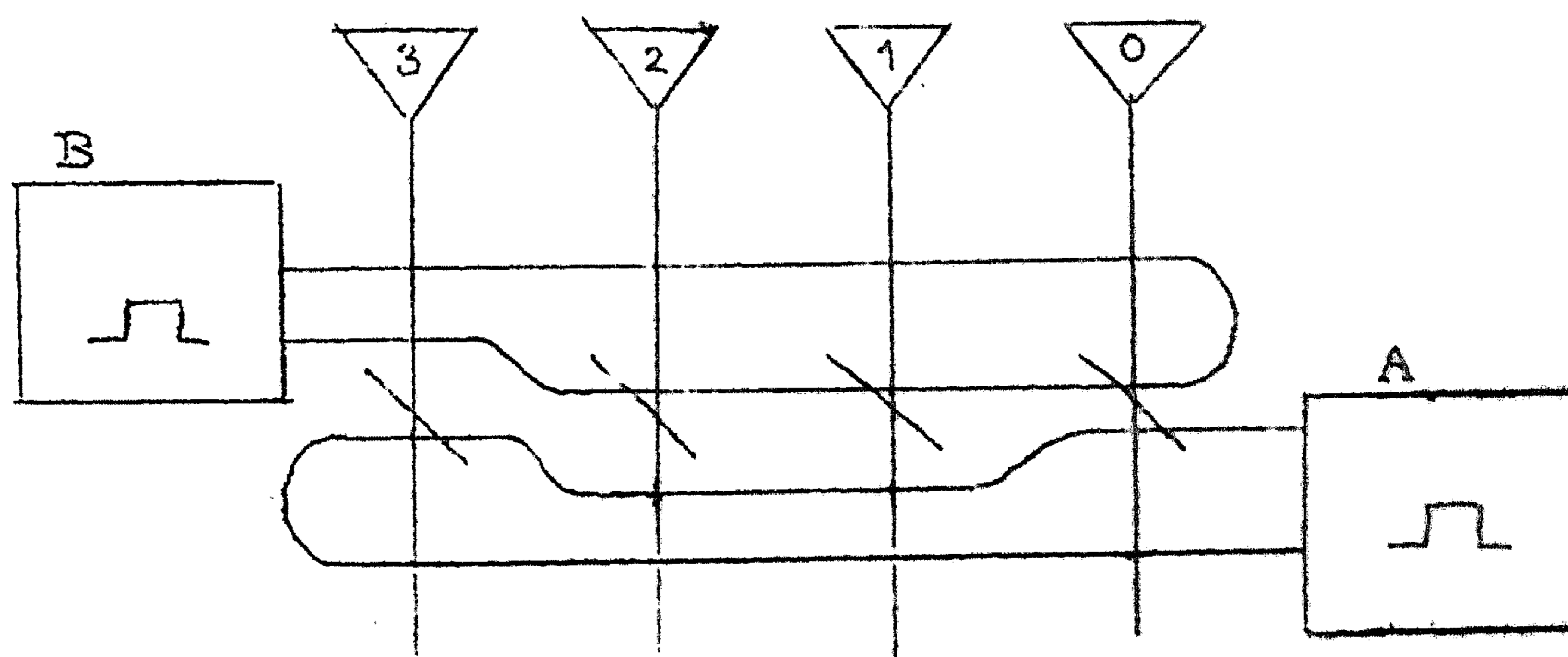


fig. 27

Wanneer men een 9 uit de kernen wil laten komen, vlecht men de draad door de kern voor het 0e cijfer en het 3e cijfer. Voor een 7 wordt de draad door de kernen 0, 1 en 2 gevlochten. In de leesversterkers komen de gewenste cijfers, wanneer door één van de draden een stroom i_{\max} wordt gestuurd. Wordt door beide draden tegelijkertijd stroom gestuurd, dan ontstaat er een ander gegeven in de uitleesversterkers, want bij de genoemde voorbeelden zouden alle kernen een 1 produceren. Dit laatste wordt bij de X-1 niet toegepast.

Wanneer de selectie op bovenstaande wijze werd uitgevoerd, zou er voor ieder woord een versterker moeten worden toegepast. De gevlochten drijflijn wordt echter gebruikt voor alle woorden en is van links naar rechts, dan weer van rechts naar links en zo voort door de respectievelijke rijen kernen gevlochten (fig. 28).

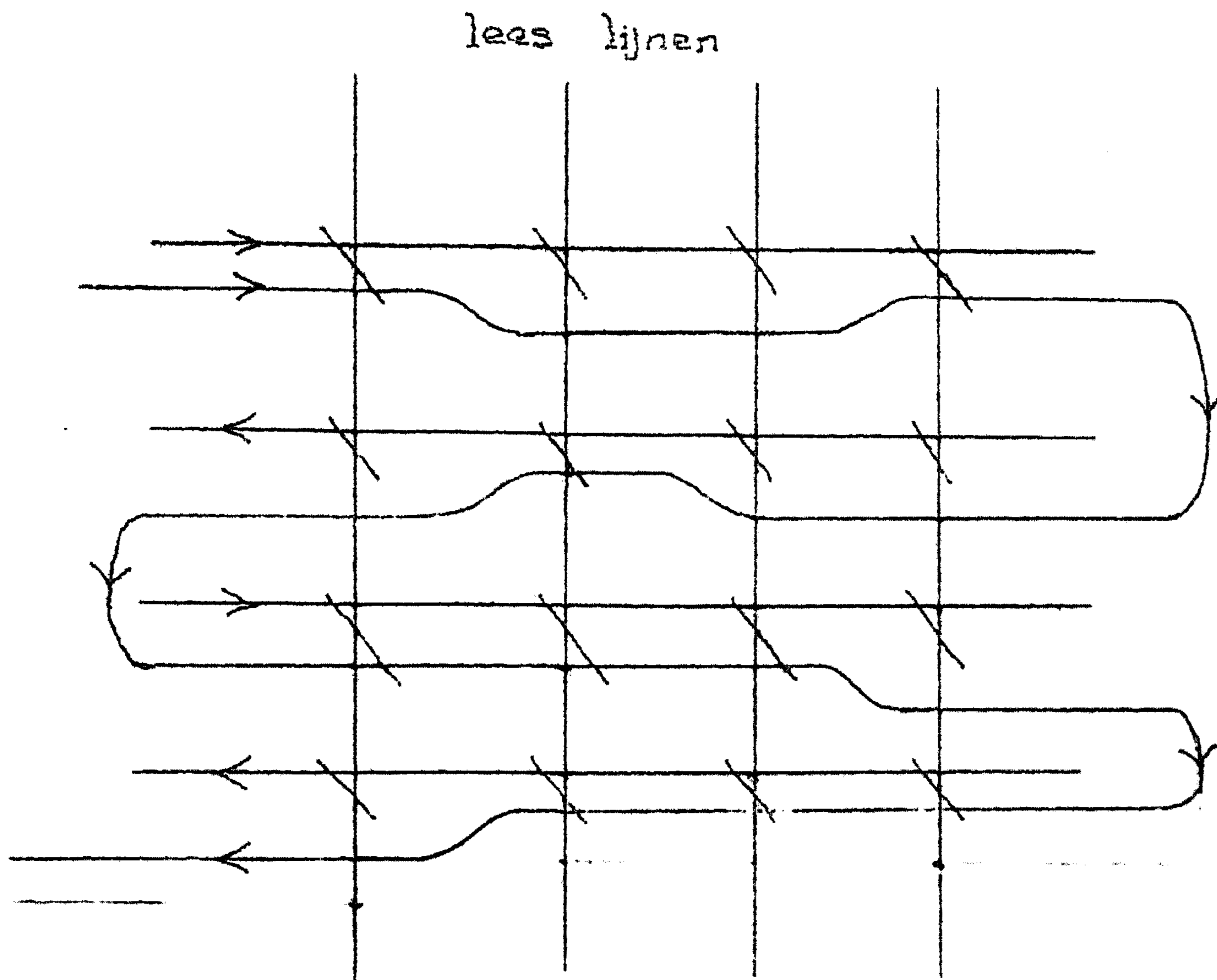


fig. 28

Door deze drijflijn loopt een stroom $\frac{1}{2}i_{\max}$. Geen van de kernen zal dus omgaan. De versterkers behorend bij deze gevlochten drijflijn worden naar het voorbeeld van het levend geheugen de verticale versterkers genoemd. Door alle kernen van een woord loopt nog een drijflijn, die een stroom voert van $\frac{1}{2}i_{\max}$, als die betreffende rij kernen is geselecteerd. Deze drijflijn wordt de horizontale drijflijn

genoemd. De stroom door de horizontale drijflijn moet natuurlijk dezelfde richting hebben als de stroom van de gevlochten lijn heeft door die betreffende rij kernen.

In het geselecteerde woord zijn een aantal kernen in stand 0 gebracht. Deze kernen moeten weer worden omgezet, voordat het woord opnieuw kan worden gelezen. Daartoe zijn door alle kernen twee draden gevlochten, die elk $\frac{1}{2}i_{\max}$ voeren en wel in zodanige richting, dat alle kernen van een kast in stand 1 worden gezet, ook van de niet geselecteerde woorden, hoewel deze laatste natuurlijk allemaal al een één bevatten. Immers het omzetten gebeurt na elke keer, dat er gelezen wordt in het dood geheugen.

Een rij van 28 kernen levert een woord, maar diezelfde kernen kunnen ook een ander woord leveren, wanneer daardoor een andere draad op een andere wijze is gevlochten. Het maximum aantal woorden, dat een rij kernen kan leveren, wordt alleen bepaald door het aantal draden, dat door een kern kan worden geregen.