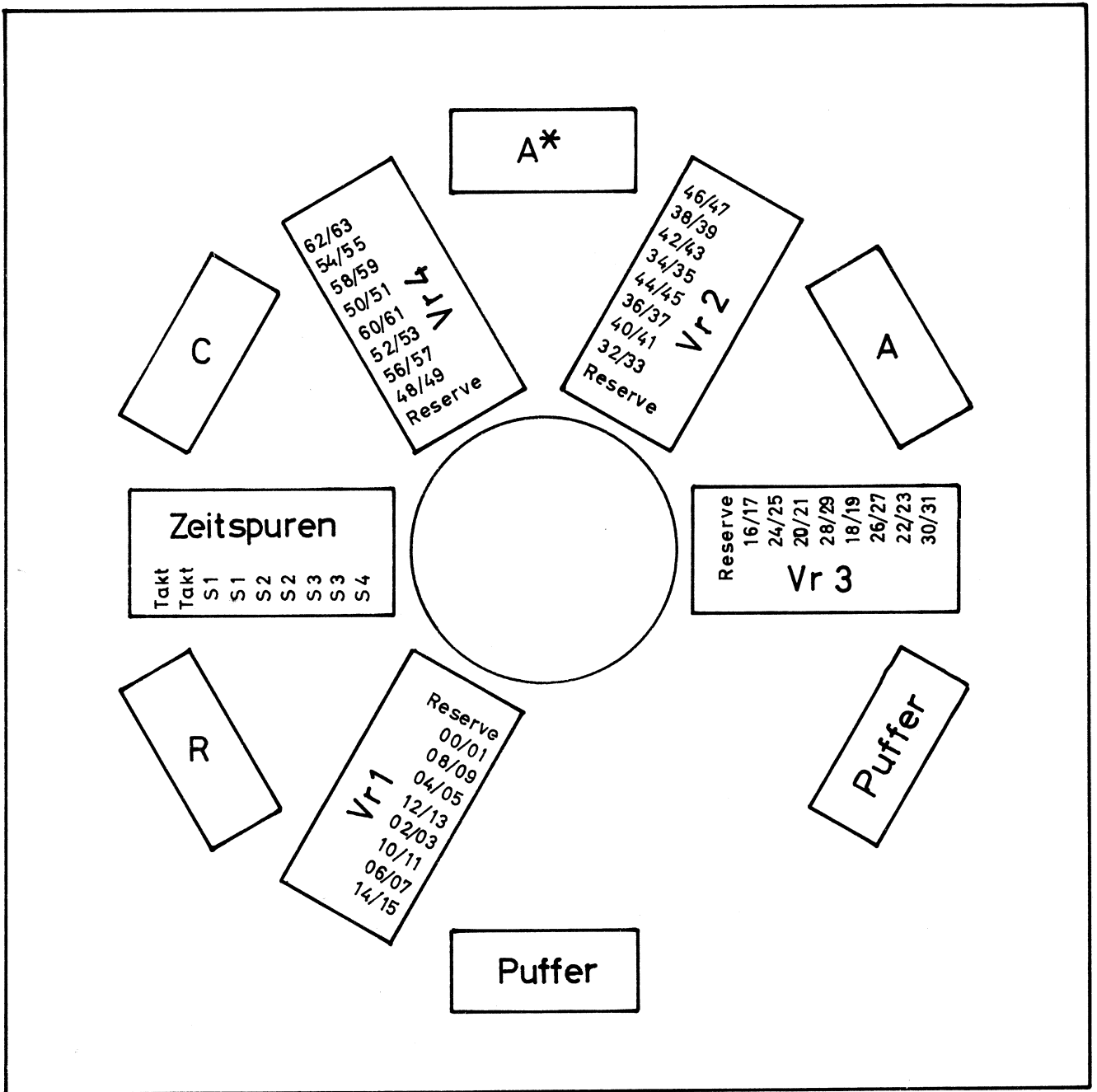


LGP-21

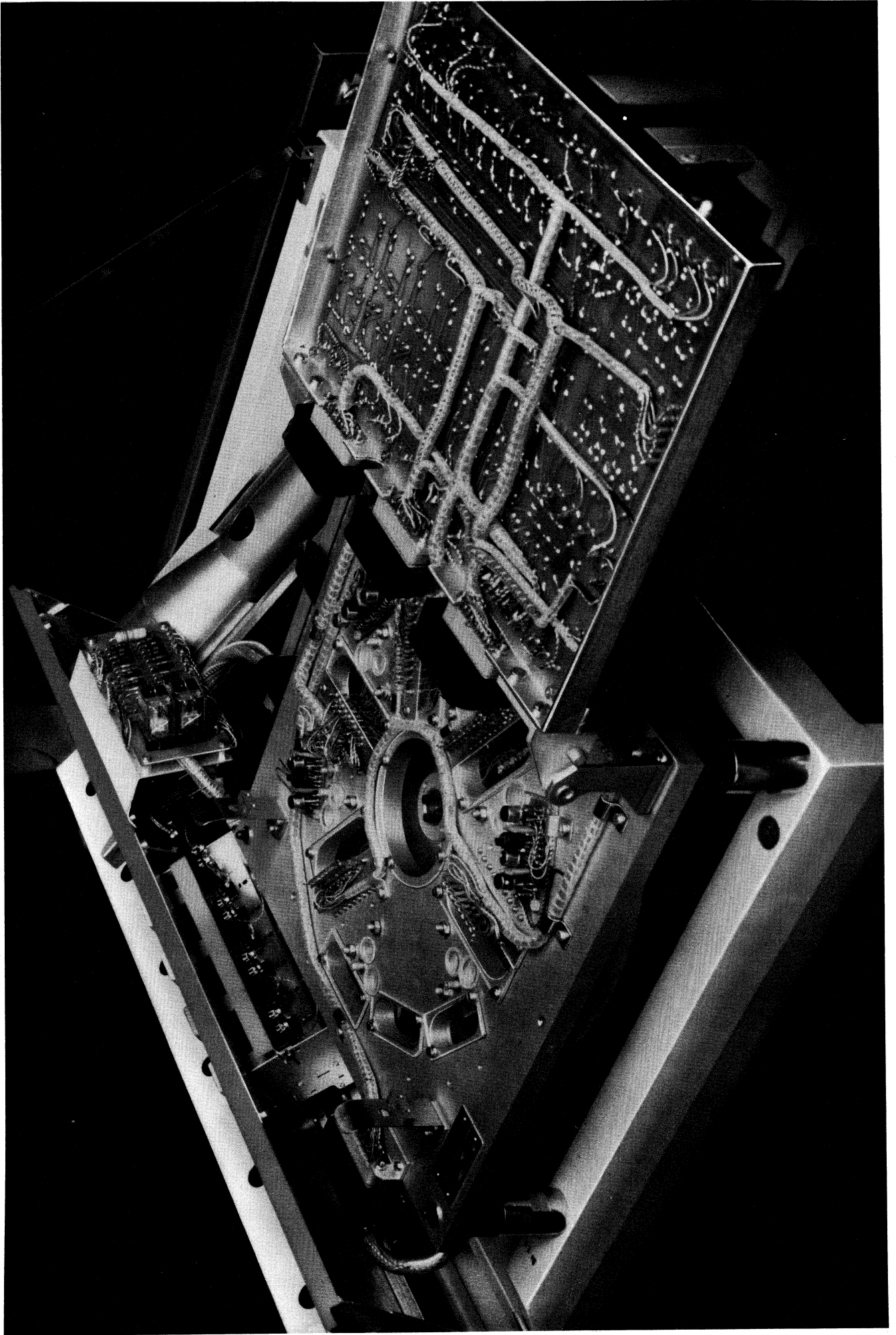




Frontplatte Rechner

Plattenspeicher MSP 22

Anordnung der Köpfe



S a c h r e g i s t e r
=====

1. Allgemeines

1.1. Technische Daten der Grundausrüstung	A1 - 1
1.2. Prinzipielle Arbeitsweise	A1 - 2
1.3. Begriffe und Symbole	A1 - 4
1.4. Erweiterungen und Zusatzgeräte	A1 - 7

2. <u>Systembeschreibung</u>	A2 - 1
2.1. <u>Stromversorgung</u>	A2 - 1
2.2. <u>Speichereinheit und zugehörige Steuerung</u>	A2 - 1
2.2.1. Schreibverfahren	A2 - 1
2.2.2. Speicherplätze	A2 - 2
2.2.3a Druckschaltungsanordnung	A2 - 3
2.2.3. <u>Hauptspeicher</u>	A2 - 5
2.2.3.1. Kopfanwahl	A2 - 5
2.2.3.2. Schreiben	A2 - 6
2.2.3.3. Lesen	A2 - 11
2.2.4. <u>Umlaufregister</u>	A2 - 13
2.2.4.1. Schreibverstärker	A2 - 14
2.2.4.2. Leseverstärker	A2 - 15
2.2.5. <u>Takt- und Zeitspuren</u>	A2 - 17
2.2.5.1. Leseverstärker für S1,S2,S3	A2 - 17
2.2.5.2. Leseverstärker für den Grundtakt cp	A2 - 17
2.2.5.3. Taktleistungsverstärker	A2 - 19
2.3. <u>Grundsaltungen</u>	
2.3.1. Flip-Flop	A2 - 21
2.3.2. Norschaltung	A2 - 23
2.3.3. Diodengekoppelter Inverter	A2 - 24
2.3.4. Emitterfolger	A2 - 24
2.3.5. One Shot	A2 - 25
2.3.6. Astabiler Multivibrator	A2 - 27
2.3.7. Translator-Magnet-Treiber	A2 - 28
2.3.8. Translator-Kupplungs-Treiber	A2 - 29
2.3.9. Leistungs-Treiber	A2 - 30
2.3.10. Treiber	A2 - 31

2.4. Zeitsteuerung (Phasen)

2.4.1. Zeitmarkierungen

2.4.2. Phasen ($\emptyset 1 \dots \emptyset 4$)	A2 - 34
2.4.2.1. Phase 1 ($\emptyset 1$)	A2 - 35
2.4.2.2. Phase 2 ($\emptyset 2$)	A2 - 36
2.4.2.3. Phase 3 ($\emptyset 3$)	A2 - 37
2.4.2.4. Phase 4 ($\emptyset 4$)	A2 - 39

2.5. Funktion der Befehle

2.5.1. Z-Befehl	A2 - 41
2.5.1.1. Z0000 - Z0100	A2 - 41
2.5.1.2. Z-Befehl mit Spuradresse > 3	A2 - 42
2.5.1.3. -Z-Befehl	A2 - 43
2.5.2. B-Befehl	A2 - 45
2.5.3. Y-Befehl	A2 - 46
2.5.4. R-Befehl	A2 - 47
2.5.5. D-Befehl	A2 - 48
2.5.6. N-Befehl	A2 - 61
2.5.7. M-Befehl	A2 - 68
2.5.8. E-Befehl	A2 - 73
2.5.9. U-Befehl	A2 - 74
2.5.10. T-Befehl	A2 - 75
2.5.11. H-Befehl	A2 - 76
2.5.12. C-Befehl	A2 - 77
2.5.13. A-Befehl	A2 - 78
2.5.14. S-Befehl	A2 - 80

2.6. Bedienungsfeld

2.6.1. Ein/Aus	A2 - 81
2.6.2. Stop	A2 - 81
2.6.3. E/A	A2 - 81
2.6.4. Start	A2 - 82
2.6.5. PST, PS4, PS8, PS16, PS32	A2 - 82
2.6.6. Modus	A2 - 82
2.6.7. Füllen/Löschen	A2 - 84
2.6.8. Ausführen	A2 - 84

2.7. <u>Ein-Ausgabe (E/A-Logik)</u>	
2.7.1. Anwahl der E/A-Einheit und Unterscheidung zwischen 4- oder 6-Bit Ein-Ausgabe	A2 - 85
2.7.2. Eingabe von Hand (Manuell)	A2 - 86
2.7.3. Eingabe über Flexowriter-Leser	A2 - 89
2.7.4. Links-Schiften	A2 - 92
2.7.5. Ausgabe über Flexowriter (Druckbefehl)	A2 - 93
2.7.6. Eingabe über Tally Leser	A2 - 95
2.8. <u>Bedeutung der Buchstaben in den logischen Gleichungen</u>	A2 - 99
2.8.1. Flip Flops	A2 - 99
2.8.2. Logische Ausdrücke	A2 - 100
2.8.3. Schreibsignale	A2 - 101
2.8.4. Lesesignale	A2 - 101
2.8.5. Bedienungsfield	A2 - 101
2.8.6. Ein-Ausgabe (E/A-Steuerung I)	A2 - 102
2.9. <u>Logische Funktionsgleichungen</u>	A2 - 105
2.9.1. Phasensteuerung (80570)	A2 - 107
2.9.2. Rechenwerk (80571)	A2 - 111
2.9.3. Kommandowerk (80572)	A2 - 116
2.9.4. E/A-Steuerung I (80573)	A2 - 119
2.10. <u>Blockschaltbilder</u>	
2.10.1. Phasensteuerung (80570)	A2 - 123
2.10.2. Rechenwerk (80571)	A2 - 125
2.10.3. Kommandowerk (80572)	A2 - 127
2.10.4. E/A-Steuerung I (80573)	A2 - 129
2.10.5. Flexowriter-Eingabe	{ A2 - 131 A2 - 133
2.10.6. Flexowriter-Ausgabe	A2 - 135
2.10.7. Tally Leser Eingabe	A2 - 137

1. Allgemeines

1.1. Technische Daten der Grundausrüstung

Das LGP-21 Standard-System setzt sich aus dem Rechner LGP-21 und einem modifizierten Flexowriter (Firma Friden) als Ein-/Ausgabe-Einheit zusammen.

Rechnertyp	Digitalrechner
Zahlendarstellung	binär
Adressensystem	Einadressensystem
Arbeitsweise	in Serie
Taktfrequenz	100 kHz
Speichertyp	magn. Plattenspeicher
Speicherkapazität	4096 Worte
Wortlänge	30 Bit + Vorzeichen + Spacerbit
Speicherdrehzahl	1500 U/min
Zugriffszeit	min. 5,8msec - Mittel 21msec
Anzahl der Befehle	23
Rechengeschwindigkeit (ohne Zugriffszeit):	
N-Multiplikation	ca. 21,7 msec
M-Multiplikation	ca. 22 msec
Division	ca. 22,3 msec
alle anderen Befehle	ca. 0,3 msec
Ein-/Ausgabeformat	dezimal, alphanumerisch, hexadezimal
Flexowriter	ca. 10 Zeichen/sec

Anschlußwerte:

220 V; 50 Hz, einphasig. Das Netz muß entsprechend den FTZ-Vorschriften entstört sein.

Zulässige Netzschwankungen	$\pm 10 \%$
Zulässige Umgebungstemperatur	$+10^{\circ}\text{C} \dots +35^{\circ}\text{C}$
Leistungsaufnahme	300 Watt

Abmessungen:	<u>Länge</u>	<u>Tiefe</u>	<u>Höhe</u>	<u>Gewicht</u>
Rechner	65cm	70cm	88cm	162kg
Tisch m. Schreibmasch.	120cm	70cm	75cm	127kg

1.2. Prinzipielle Arbeitsweise:

Der LGP-21 arbeitet mit alphabetischen und numerischen Informationen. Die interne Zahlendarstellung ist rein dual. Mit Hilfe von gespeicherten Unterprogrammen kann jedoch eine dezimale Ein- bzw. Ausgabe erfolgen.

Auf dem Speicher befinden sich 32 Hauptspeicherspuren mit je 128 Zellen (beachte A2-3!) zur Speicherung von Befehlen und Zahlen (Rechnerprogramm).

4 weitere Spuren sind Umlaufregister: Der Akkumulator A und der doppelte Akkumulator A* zur Speicherung der Operanden und Zwischenergebnisse; das Befehlsregister R, zur Speicherung des jeweils auszuführenden Befehls; und das Zählregister C, zur Speicherung der Adresse des nächsten Befehls.

Zur Zeitsteuerung des internen Rechenablaufs dienen 4 Taktspuren: Cp, S1, S2, S3.

Eine Hauptspeicherspura besteht aus 128 Zellen zu je 32 Bit. Das erste Bit gibt das Vorzeichen des Zelleninhaltes an, das letzte Bit ist eine Leerstelle. In den Bitpositionen 1 bis 30 kann der Absolutbetrag einer Zahl gespeichert werden. Die Bit-Position 0 enthält jeweils das Vorzeichen der Zahl (s. Bild 1.1).

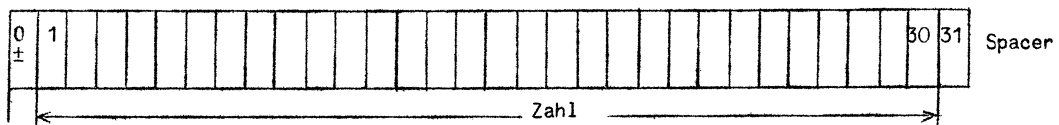


Bild 1.1 Zahlendarstellung

Eine Zelle, die zum Speichern eines Befehls benutzt wird, enthält den entsprechenden Befehl und die Adresse des Operanden. Die restlichen Bit-Positionen können noch zum Speichern von Zahlen verwendet werden (s. Bild 1.2).

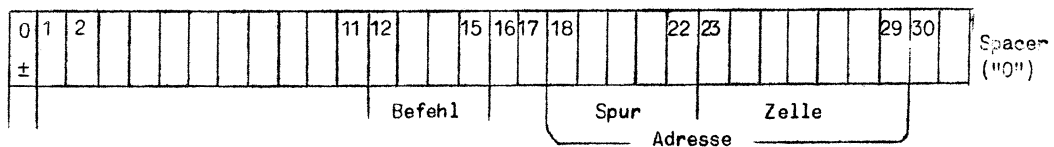


Bild 1.2. Befehlswort

Der Rechner führt einen Befehl in vier Arbeitsgängen aus:

- Phase 1 Das Zählregister bestimmt die Position im Hauptspeicher, in der sich der Befehl befindet. Die Spur wird ausgewählt, und bei Erreichen der betreffenden Zelle wird ein Koinzidenzsignal gegeben.
- Phase 2 Der Befehl wird vom Hauptspeicher in das Befehlsregister übertragen, gleichzeitig wird im Zähler eine "1" addiert.
- Phase 3 Der Operand, der durch die Adresse im Befehl bestimmt ist, wird im Hauptspeicher gesucht. Der zweite Operand befindet sich bereits im Akkumulator als das Ergebnis vorausgegangener Befehle.
- Phase 4 Der Rechenautomat führt die Operation aus, wie es im Befehl angegeben ist.

Die Zykluszeit beträgt 18 Wortzeiten, bei einer Sonderausführung des LGP-21 9 Wortzeiten. Diese Zykluszeit ist identisch mit der Zeit, die für die Ausführung eines Befehls bei optimierten Programmen notwendig ist mit Ausnahme der Befehle M, N und D, die zur Befehlsausführung eine Scheibenumdrehung plus eine Zykluszeit benötigen.

Alle Programme, die für den LGP-30 optimal sind, sind auch für die Normalausführung des LGP-21 optimal.

Programmierung siehe Programmierhandbuch.

1.3. Begriffe und Symbole

Der LGP 21 ist ein freiprogrammierbarer digitaler Rechenautomat. Alle arithmetischen und logischen Funktionen werden von einem gespeicherten Programm gesteuert. Der Speicher für Programm und Information wird als Hauptspeicher bezeichnet. Der Speicher für die Operationen und deren Ausführung als Register. Die logischen Verknüpfungen werden in der Logik durchgeführt. Die Zeitsteuerung (Takte) entspricht einem festverdrahteten Unterprogramm. Die E/A-Geräte dienen zur Ein- und Ausgabe.

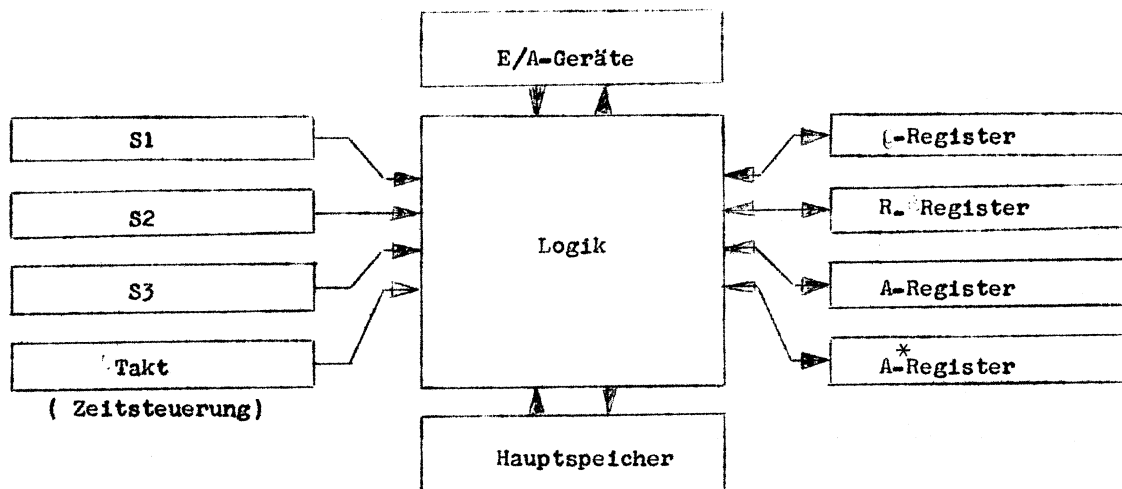


Bild 1.3.

Die Basis der Logik ist die Entscheidung zwischen zwei Möglichkeiten, und zwar zwischen logisch "nein" = "0" oder logisch "ja" = "1" (digital).

Im LGP 21 werden die beiden Zustände folgendermaßen dargestellt:

logisch "1" = 0 V bis - 2 V

logisch "0" = -10V bis -20 V

Die Logik mit dieser Spannungsdarstellung bezeichnet man als "Positive Logik".

Für die einzelnen Verknüpfungspunkte lassen sich mit Hilfe der Bool'schen Algebra Gleichungen aufstellen. In diesen Gleichungen stehen alle Bedingungen die erforderlich sind, um für die Erfüllung einer logischen Funktion eine "ja"-Aussage zu erhalten.

Der invertierte Ausgang einer Verknüpfung wird als "Quer"-Ausgang bezeichnet (z.B. A invertiert = A, d.h., wenn A = 0 V hat, dann muß A = -20V haben).

Setz- bzw. Rücksetzsignale für Speicherelemente werden durch ein Apostroph (A') gekennzeichnet.

Beispiel einer Gleichung:

$$F' = \underline{F} \underline{G} \underline{H} T_3 P_1 + \dots$$

Das F-Flip-Flop wird gesetzt, wenn gleichzeitig F, G, H, T_3 und P_1 logisch "1" (0 V) sind. Das Oder (+) gibt an, daß das F-Flip-Flop noch von anderen Bedingungen gesetzt werden kann.

Der Ausgang F des Flip-Flops wird als "Ein-Seite" bezeichnet, der Ausgang F als "Aus-Seite". Das Flip-Flop wird durch ein F' zurückgesetzt (Aus-Seite = 0 V).

2. Systembeschreibung

Der LGP 21 besteht im wesentlichen aus folgenden Funktionsgruppen:

Stromversorgung

Speicherteil (mit Ansteuerung)

Logik

Bedienung (mit Ein- und Ausgabe)

2.1. Stromversorgung

Das Netzteil (Bild 3.1) des Rechners ist steckbar angeordnet. Es enthält die gesamte Stromversorgung des Rechners. Das Netzteil kann an die üblichen Netzspannungen angepaßt werden. Mit Hilfe einer Transistorverzögerungsschaltung wird der Speicherscheibenmotor nach Hochlauf (ca. 8 sec) auf 115 V umgeschaltet und gleichzeitig die verzögerte Gleichspannung -20d eingeschaltet. Außerdem enthält das Netzteil die Betriebskondensatoren für die Lüfter und den Speicherscheibenmotor. Das Netzteil ist nicht stabilisiert, da alle Schaltungen für die nach VDE zulässigen Netzspannungsschwankungen dimensioniert sind.

Die Daten des Netzteils sind im Datenblatt 80605Db1-X(4) (2 Blatt) und in der Prüfanweisung 80605 Pa1-X(4) zusammengefaßt.

2.2. Speichereinheit und zugehörige Steuerung

2.2.1. Schreibverfahren:

Als Speichermedium dient eine mit Kobalt beschichtete rotierende Scheibe. Die Lese-Schreibköpfe schweben in einem Abstand von ca. 5μ über der Scheibe auf einem Luftpolster, das sich durch die Drehbewegung der Scheibe bildet.

Jeder Kopf ist einer bestimmten Spur zugeordnet. Der radiale Abstand der einzelnen Spuren (ca. 0,8 mm) ist durch die Schreibkernbreite zuzüglich eines gewissen Sicherheitsabstandes gegeben. Beim Schreiben wird durch den Kopf ein so großer Strom geschickt, daß die magnetische Kobaltschicht bis in die Sättigung getrieben wird.

Durch Umkehren der Stromrichtung erfolgt eine entgegengesetzte Magnetisierung. Beim Lesen wird durch die Magnetisierungsänderung eine Spannung in den Lesekopf induziert. Jeder positive Leseimpuls zeigt einen Zustandswechsel in der einen Richtung, jeder negative einen Wechsel in der anderen Richtung an. Dieses Schreibverfahren wird als "non return to zero" (NRZ) bezeichnet, da nur die beiden Sättigungszustände vorkommen, die den beiden Binärwerten "0" und "1" zugeordnet sind.

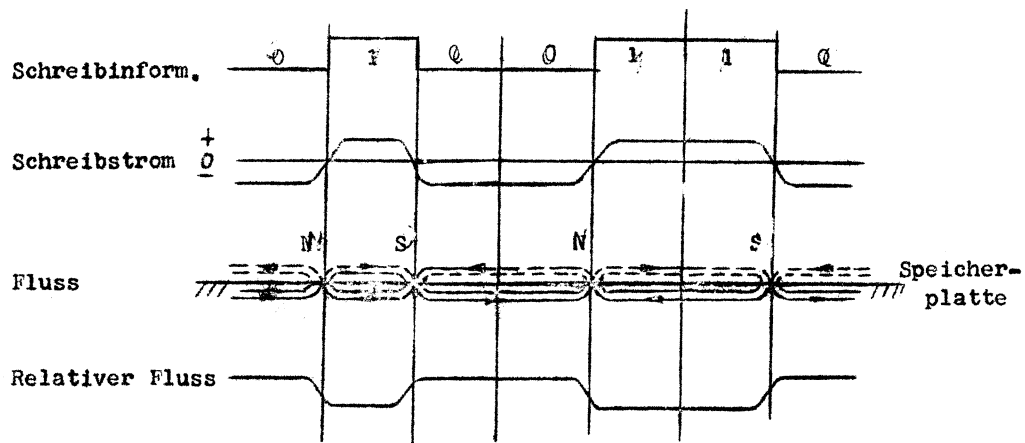


Bild 2.1.

2.2.2. Speicherplätze:

Die Speicherkapazität des Hauptspeichers beträgt 127 000 Informationsbits. Die Information wird in Serie (d.h. Bit nach Bit) übertragen und geschrieben. Eine Speicherspür nimmt 4096 Bit auf und ist in 128 Sektoren mit je 32 Bit unterteilt. Ein Sektor entspricht einem Wort.

Die Umdrehung der Speicherscheibe dauert ca. 41 msec, eine Bitzeit beträgt ca. 10 μ sec.

Der Hauptspeicher besteht aus 32 Spuren. Die den einzelnen Spuren zugeordneten Köpfe sind in 4 Blöcke zu je 8 (+ 1 Reservekopf) zusammengefaßt.

Die Zeitspuren C_p , S_1 , S_2 , S_3 und je eine beschriebene Reservespur befinden sich auf einem fünften Block.

Die 4 Umlaufregisterköpfe sind einzeln angeordnet.

Mechanischer Lageplan der Köpfe siehe Bild 3.2.

Beim Hauptspeicher wird zur Erleichterung der Programmierung und in Analogie zum LGP 30 Spur und Sektor durch jeweils 6 Binärstellen dargestellt. Hiermit ergibt sich eine Speichereinteilung in 64 Spuren mit je 64 Sektoren.

Beispiel:

Das gesuchte Wort sei Spur 48 Sektor 52; dann lautet die Adresse hexadezimal: 30KO (1. Code-Karte).

Die Numerierung der Hauptspeicherköpfe geht von 00 bis 31. Jede unter einem Kopf liegende Spur ist in zwei Programmspuren zu je 64 Sektoren eingeteilt.

Jede geradzahlige Spurnummer - bzw. jede ungeradzahlige Spurnummer minus eins - dividiert durch zwei ergibt die zugehörige Kopfnummer (s. Bild 3.2. und 3.3.).

2.2.3. Druckschaltungsanordnung

Die einzelnen Druckschaltungen für die Speichereinheit mit zugehöriger Steuerung sind auf der Speichereinheit und in dem schwenkbaren Kartengehäuse 1 (Verdrahtungsplan 80645 VP1-X(2)) untergebracht.

Die Aufteilung der Elektronik auf die einzelnen Karten ist in den Blockschaltbildern 3.3. und 3.4. dargestellt.

- Position 1 befindet sich direkt auf der Speichereinheit.
- Position 2 befindet sich auf der klappbaren Druckschaltung über dem Speicher.
- Position 3-6 befindet sich im Kartengehäuse 1.

2.3. Hauptspeicher (siehe auch Prüfvorschrift 80001-Pa1-X(4))

2.3.1. Kopfanwahl

Die Anwahl der einzelnen Köpfe erfolgt wie die des Knotenpunktes einer Matrix in Zeilen und Spalten (siehe Bild 3.3 bis 3.7).

Die einzelnen Spalten (Kanäle) Ch_1 bis Ch_8 werden durch P_3 , P_4 und P_5 angewählt, die einzelnen Zeilen (Blöcke) Vr_1 bis Vr_4 durch P_1 und P_2 .

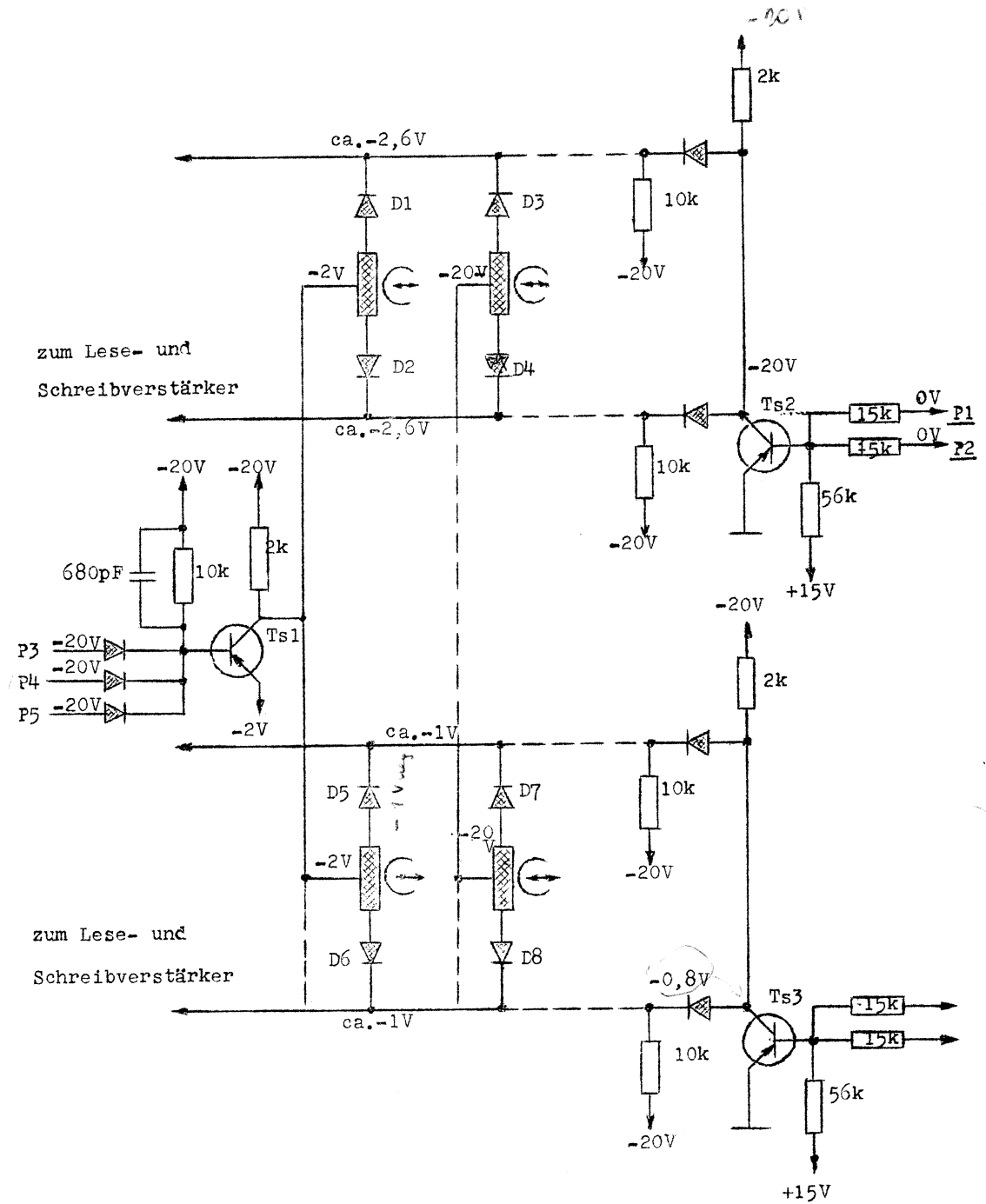


Bild 2.2.

Der Kopf 00 sei angewählt (Bild 2.2). Der entsprechende Kanal wird über den diodengekoppelten Inverter mit P_3, P_4, P_5 ($= -20V$) angewählt. Bei dem angewählten Kanal liegt an der Mittelanzapfung des Kopfes $-2V$, da Ts_1 durchgeschaltet ist.

Bei den nicht angewählten Kanälen liegt an der Mittelanzapfung des Kopfes $-20V$. Der entsprechende Block wird über ein Nor-Gatter mit P_1, P_2 ($= 0V$) angewählt. Hiermit sind die Matrixdioden D_1 und D_2 leitend. Sämtliche anderen Matrixdioden sind gesperrt. Die Sperrschwelle für das Lesesignal der nicht angewählten Köpfe beträgt einmal ca. $2V$ (z.B. D_5, D_6) oder ca. $18V$ (z.B. D_3, D_4).

Um beim Schreiben zu verhindern, daß mit jedem Kopf des angewählten Kanals geschrieben werden kann, ist die Schreiberlaubnis für die Schreibverstärker noch mit der Blockanwahladresse gekoppelt.

2.3.2. Schreiben:

Der Schreibstrom für den angewählten Kopf wird aus einer Konstantstromquelle gespeist. Dadurch erreicht man einen schnellen Stromanstieg in der Schreibspule. Der maximale Schreibstrom ist nach ca. $2 \mu\text{sec}$ erreicht.

Arbeitsweise der Konstantstromquelle:

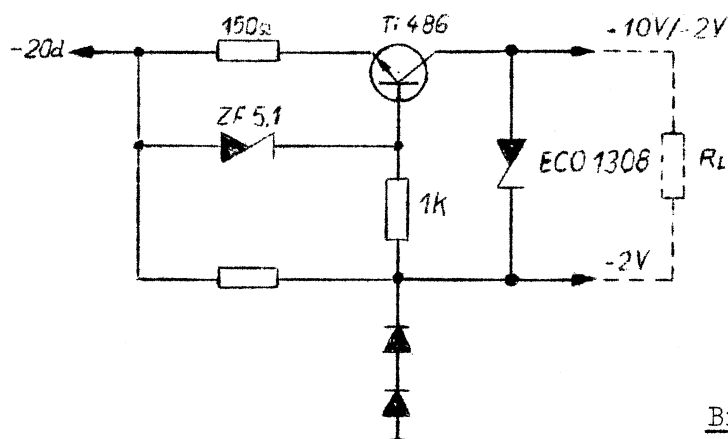


Bild 2.3.

Der Transistor wirkt als veränderbarer Widerstand, der den Strom auf 30mA begrenzt. Je positiver das Potential U_E gegenüber U_B wird, je größer wird der Widerstand des Transistors. Das Basispotential U_B beträgt ca. $14,9\text{V}$.

Das Emitterpotential wird durch den Spannungsabfall an R_1 bestimmt

$$U_E = -20d + I_1 \cdot R_1$$

R_1 ist zusammen mit der Zenerdiode D_2 so bemessen, daß der Strom nicht über 30 mA ansteigen kann. Der Spannungsabfall zwischen Emitter und Kollektor ist abhängig vom Lastwiderstand ($R_L = \text{klein}$, $U_{CE} = \text{groß}$; $R_L = \text{groß}$, $U_{CE} = \text{klein}$). Wird nicht geschrieben, so fließen die 30 mA über die Zenerdiode D_1 und es steht am Kollektor ca. 10V. Wird geschrieben, fließen die 30 mA über den angewählten Kopf. Der Lastwiderstand R_L entspricht dann dem Innenwiderstand des Kopfes. Die Spannung am Kollektor bricht bis auf ca. -2V zusammen, da $R_L < \frac{10V}{30mA}$ ist (siehe Bild 2.4).

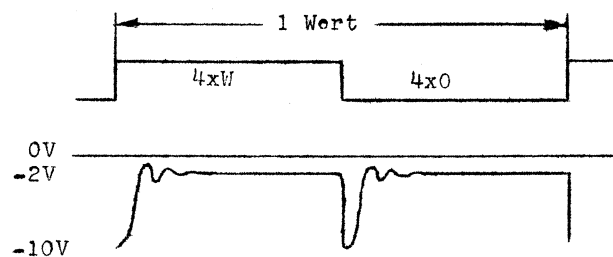


Bild 2.4.

Entsprechend dem Aufbau der Matrix werden vier getrennte Schreibverstärker benötigt. In Bild 2.5 ist der angewählte Schreibverstärker für Block Vr_1 (Kopf 00) dargestellt.

Da die Schreiberlaubnis W und die Schreibinformation Vw' an allen Schreibverstärkern gleichzeitig anliegen, müssen alle nicht benötigten Schreibverstärker gesperrt werden. Dieses geschieht entsprechend der zugehörigen Blockanwahl mit $P_1, \underline{P}_1, P_2, \underline{P}_2$ (siehe auch 2.3.1). Die Schreibinformation Vw' und \underline{Vw}' sind komplementär. Es kann nur einer der beiden Transistoren leiten. Auf dem Kopfkern befinden sich 2 Spulenhälften mit gleichem Wicklungssinn. Entsprechend der Ansteuerung von

Ts₁ oder Ts₂ fließt der Strom durch die eine oder andere Wicklungshälfte und erzeugt eine Magnetisierungsänderung, die einer logischen "1" oder "0" entspricht.

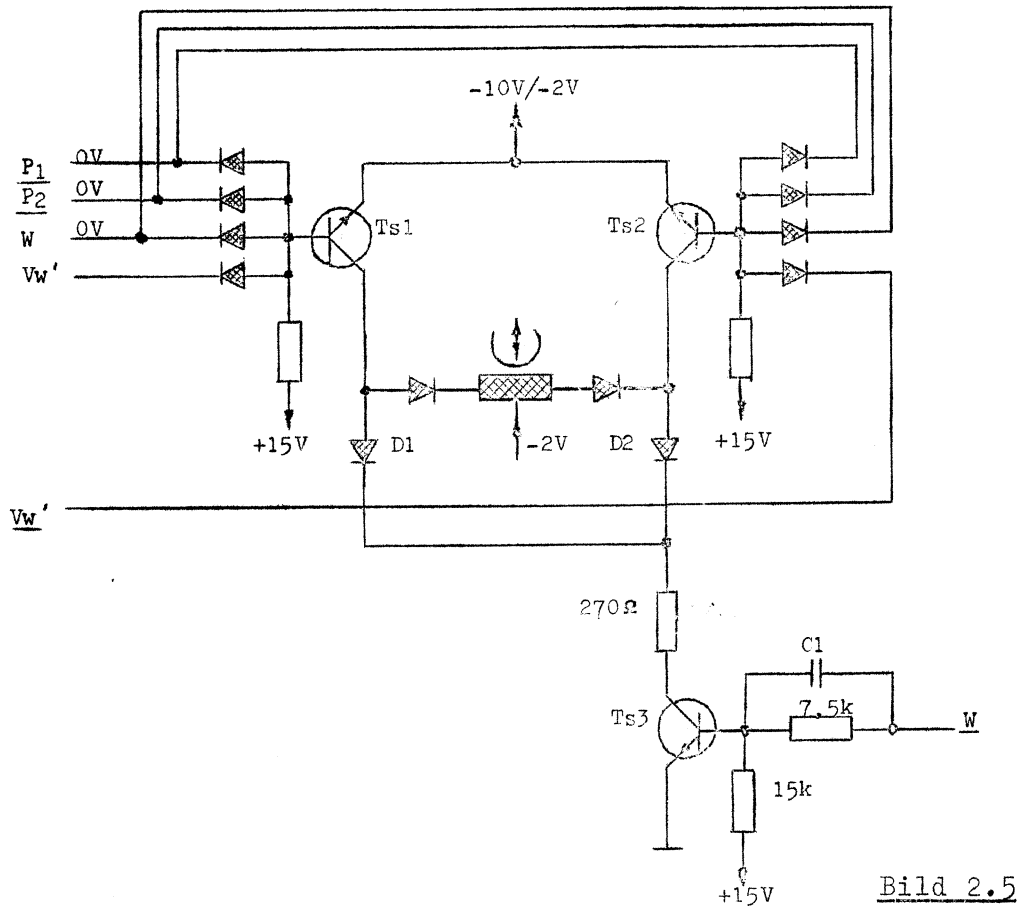


Bild 2.5

Die Schreiberlaubnis W setzt sich zusammen aus:

$$W = \underbrace{F G H}_{\emptyset 4} \underbrace{Q_1 Q_2 Q_3}_{\text{H-, C-Befehl}} + \underbrace{F G S_2}_{\emptyset 4} \underbrace{Q_1 Q_2 Q_3}_{\text{Y-, R-Befehl Adreßzeit}}$$

Die Schreibinformation Vw' setzt sich zusammen aus:

$$Vw' = L \left[\underbrace{A Q_1}_{\text{A-Information beim Y-Befehl}} + \underbrace{A Q_4}_{\text{A-Information beim C-, H-Befehl}} + \underbrace{Q_1 Q_4}_{\text{R-Befehl}} \underbrace{(K C + \bar{K} C)}_{\text{Addiere "1" zum C-Register}} \right]$$

Spacerbit-Schreibinformation: "0"

Das Schreiben einer "0" zur Spacer-Bit-Zeit ist erforderlich, um beim Lesen am Anfang eines Wortes mit einem definierten Informationsbit zu beginnen (begründet durch das NRZ-Schreibverfahren).

In Bild 2.6. ist schematisch der Stromkreis beim Schreiben dargestellt.

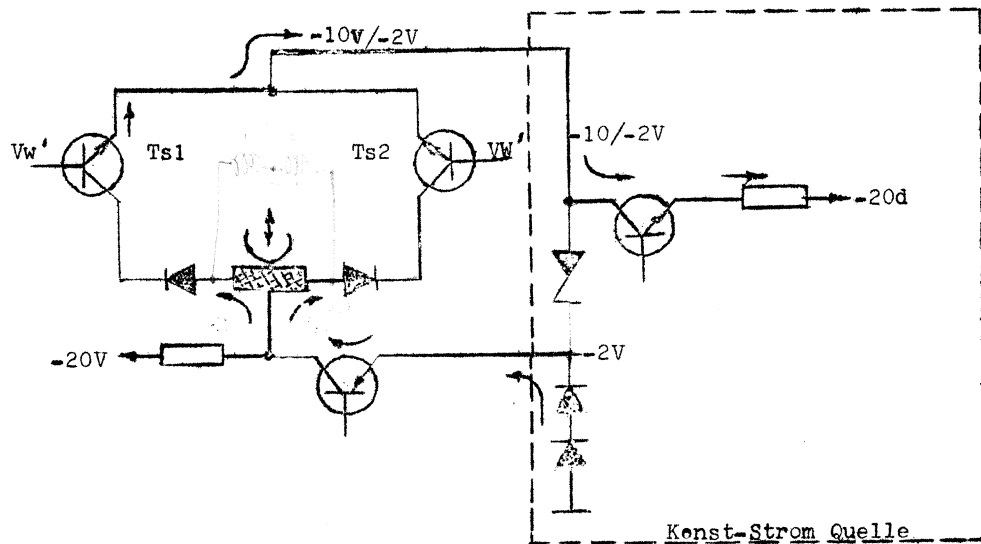


Bild 2.6.

Mit Hilfe des Transistors Ts_3 (Bild 2.5) wird bei Schreibende die gespeicherte Energie im Kopf gegen 0V abgeleitet. Bei Schreibende sind Ts_1 und Ts_2 gesperrt. Dadurch tritt eine Abschaltspannungsspitze auf, die zum Zerstören der Schreibtransistoren führen würde, wenn die Abschaltspitze nicht durch Ts_3 begrenzt wäre.

Die Schaltung ist hauptsächlich beim Y- und R-Befehl von Bedeutung, da hier das Schreiben innerhalb einer Wortzeit erfolgt und das nachfolgende Bit nicht gelöscht werden darf. Das vom Kopf erzeugte Feld muß innerhalb einer Bitzeit abgebaut sein.

2.3.3. Lesen (siehe auch Bild 3.3, 3.5, 3.6, 3.8)

Am Ausgang des Leseverstärkers und am Ausgang des Lese-Flip-Flops stehen zu jeder Zeit Lesesignale an (auch beim Schreiben). Die Torung der Information erfolgt erst später in der Logik.

Der Leseverstärker ist ein linearer Differentialverstärker mit einer Spannungsverstärkung von ca. 290. Das gelesene Signal beträgt ca. 150 bis 350 mV. Bei größeren Signalen wird der Leseverstärker übersteuert. Um eine zu starke Übersteuerung beim Schreiben zu verhindern (lange Erholzeit des Verstärkers) wird durch die Dioden D_1 und D_2 (Bild 2.7) die Spannung am Eingang des Vorverstärkers auf ca. 0,6V begrenzt. Die Widerstände R_3 und R_4 dienen zur Entkopplung.

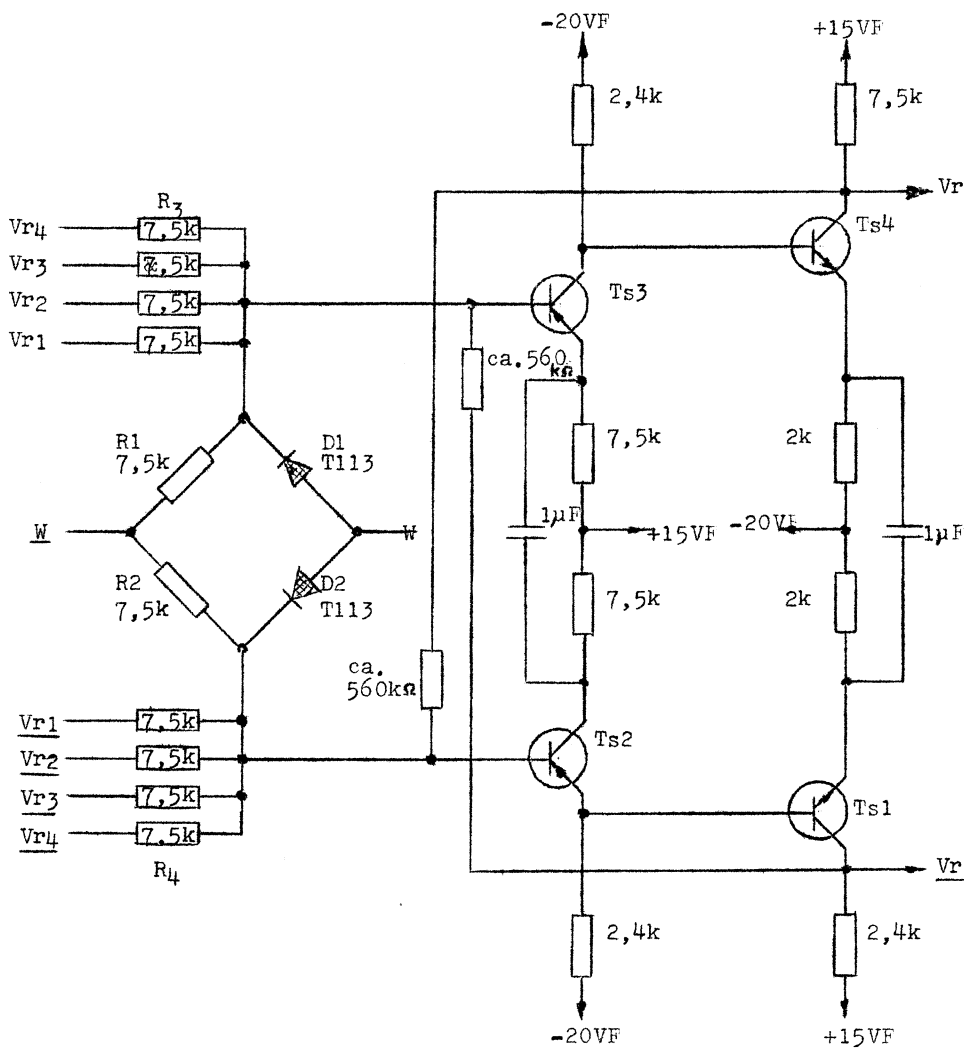


Bild 2.7.

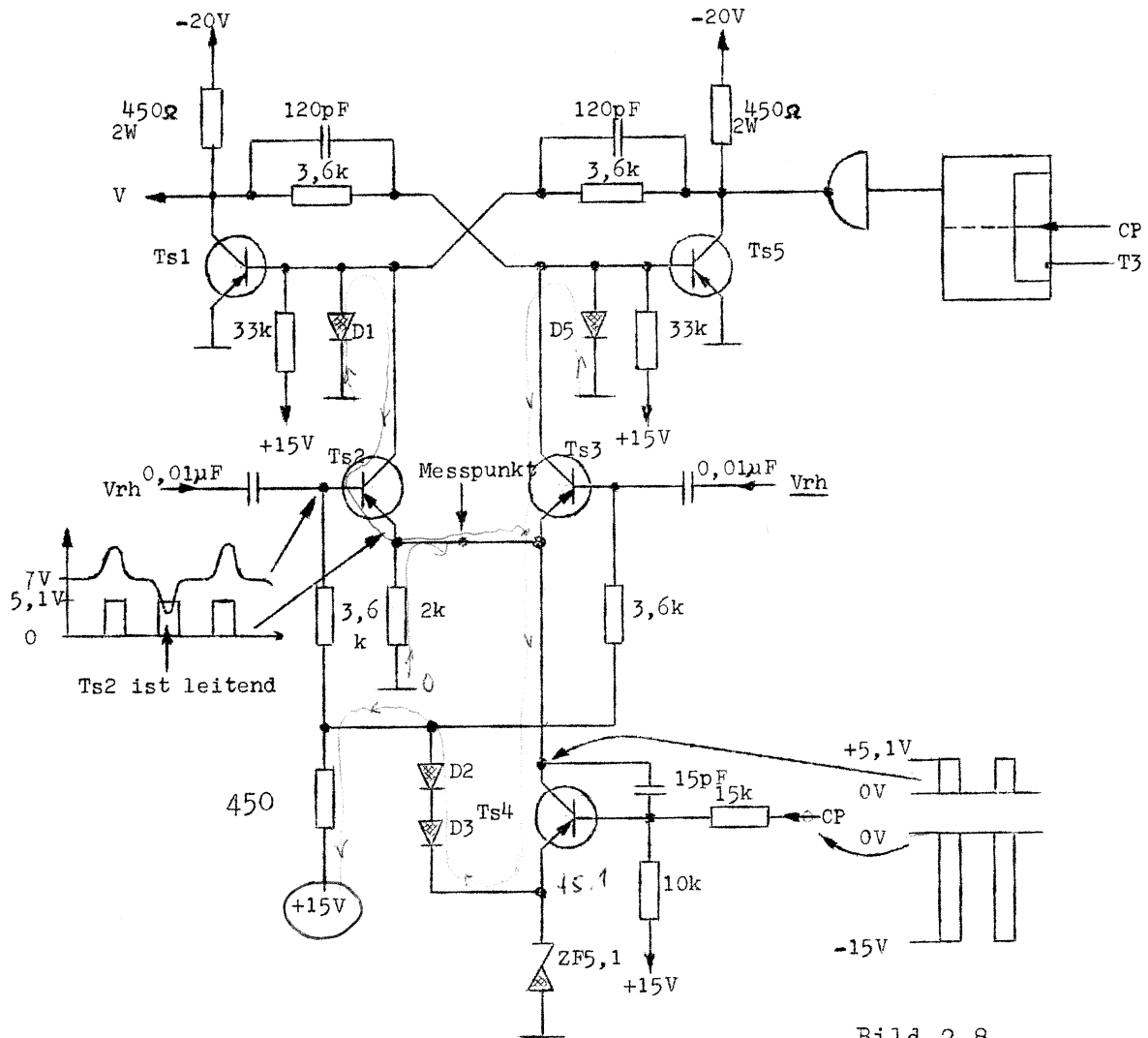


Bild 2.8.

Die gegenseitig um 180° phasenverschobenen Ausgangssignale des Vorverstärkers sind kapazitiv auf die Basen der Transistoren Ts_2 und Ts_3 gekoppelt (siehe Bild 2.8). Die Basis von Ts_2 und Ts_3 liegt entsprechend der Spannung $U = U_Z + 2 \cdot U_D$ ($= 5,1 + 2 \cdot 0,6 = 6,3V$) gleichstrommäßig auf einem Potential von +6V bis +7V. Diesem Gleichstrompotential überlagert sich das Ausgangssignal des Vorverstärkers.

Am Emitter von Ts_2 und Ts_3 liegt über Ts_4 ein invertierter Takt $\underline{cp} = CP_1 \dots$ von 0V bzw. +5,1V. Bei $cp=0V$ ist Ts_4 durchgeschaltet. Sein Kollektor liegt auf +5,1V. Bei $cp=-15V$ ist Ts_4 gesperrt. Sein Kollektor liegt über $R = 2k$ auf 0V.

Wenn Ts_4 leitend ist und am Eingang V_{rh} bzw. $\underline{V_{rh}}$ ein negativer Impuls anliegt, wird Ts_2 bzw. Ts_3 durchgeschaltet und das positive Potential des Kollektors von Ts_4 auf die Basis von Ts_1 bzw. Ts_5 gegeben. Hierdurch wird das V-Flip-Flop entsprechend der gelesenen Information gesetzt. Wenn Transistor Ts_4 gesperrt ist, kann keine positive Spannung an Ts_1 bzw. Ts_5 gelegt werden. Dadurch wird erreicht, daß nur Signale die zur Taktzeit kommen das V-Flip-Flop setzen können.

Der Transistor Ts_5 des V-Flip-Flops wird zur Spacer-Bit-Zeit durch einen Setzimpuls an seinem Kollektor auf 0V gezogen.

Dadurch ist gewährleistet, daß zur Spacer-Bit-Zeit unabhängig von der gelesenen Information das V-Flip-Flop auf "0" gesetzt wird.

2.2.4. Umlaufregister

Der LGP 21 besitzt 4 Umlaufregister. Das A-, C- und R-Register sind 1 Wort-Umlaufregister. Das A*-Register ist ein 2 Wort plus 1 Bit - Umlaufregister.

Der A*- und A-Registereingang liegen parallel. Das A*-Register wird nur während des M-, N- und D-Befehles benutzt.

Bei den Umlaufregister-Köpfen befinden sich auf einem Schuh je ein Lese- und Schreibkopf. Beide Köpfe liegen auf der gleichen Spur, d.h. die Information, die der Schreibkopf schreibt, wird vom Lesekopf gelesen.

Der Abstand zwischen Lese- und Schreibkopfspalt beträgt je nach Art des Registers 32 oder 65 Bitzeiten.

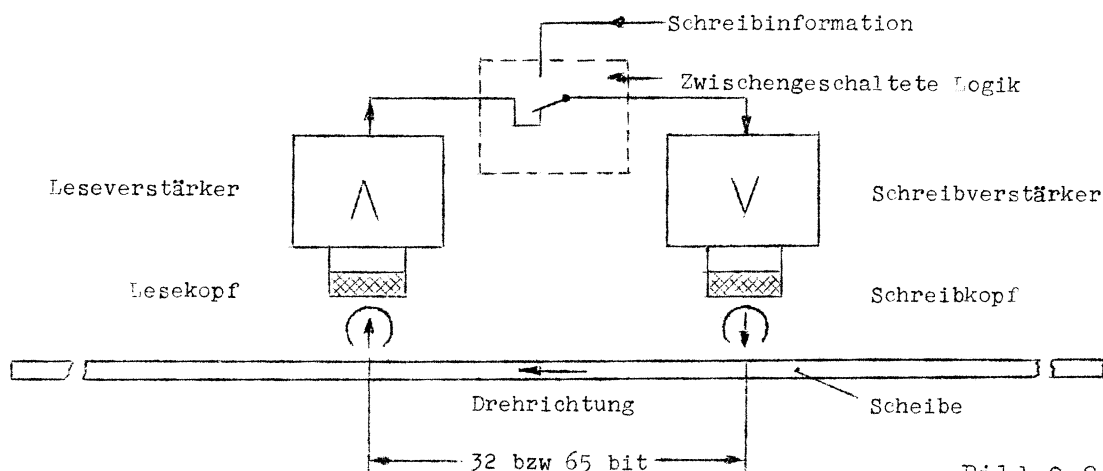


Bild 2.9.

Das geschriebene Bit wird 32 bzw. 65 Bitzeiten später wieder gelesen und gleichzeitig neu aufgeschrieben (siehe Bild 2.9). Die Information steht zu jeder Wortzeit zur Verfügung. Durch die zwischengeschaltete Logik kann der Umlauf zum Einschreiben eines neuen Wortes unterbrochen werden.

2.2.4.1. Schreibverstärker (siehe auch Bild 3.10)

Der Schreibverstärker ist als Flip-Flop (Bild 2.10) ausgebildet (Erklärung des Flip-Flop s. 2.3.2.1). Da jeweils einer der beiden Transistoren Ts_1 oder Ts_2 leitend ist, fließt ein Strom von ca. 30 mA durch den Schreibkopf. Z.B.: Ts_1 ist leitend. Der Strom fließt von 0V über Ts_1 , R_1 , Schreibkopf, R_2 und R_3 nach -20V. Kippt das Flip-Flop, kehrt sich die Stromrichtung um. Damit ergibt sich eine für die "0" und "1" Unterscheidung gewünschte entgegengesetzte Magnetisierungsrichtung auf der Scheibe.

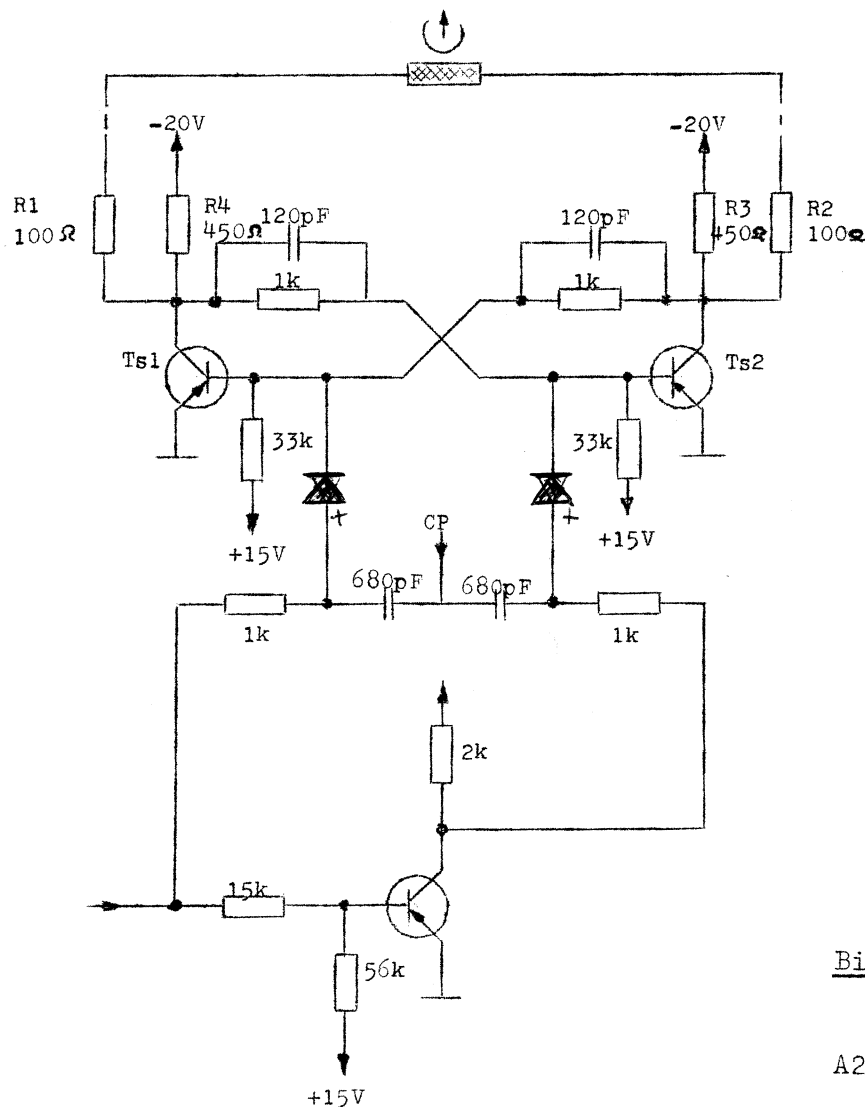


Bild 2.10

2.2.4.2. Lese-Verstärker (siehe auch Bild 3.12)

Das Lese-Signal des Kopfes wird in einem linearen Differentialverstärker (ähnlicher Aufbau wie beim Hauptspeicher und bei den Taktspuren) vorverstärkt und kapazitiv auf die nächste Stufe gekoppelt (Bild 2.11). Die Dioden D_1 und D_2 erzeugen eine Vorspannung von ca. +0,6V und verhindern eine positive Spannung von +15V an der Basis. Sie dienen zur Störunterdrückung. Die negative Hälfte des Analogsignals steuert jeweils Ts_1 oder Ts_2 durch. Der leitende Transistor setzt dann zusammen mit der positiv gehenden Flanke von CP das nachgeschaltete Flip-Flop.

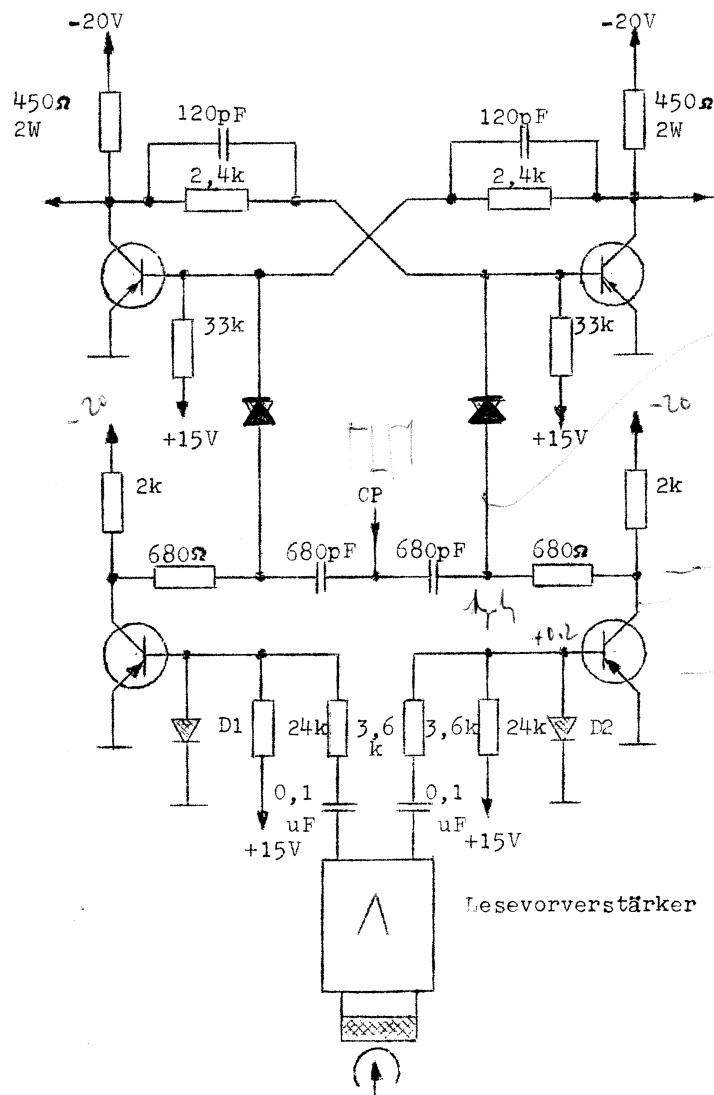


Bild 2.11

2.2.5. Takt- und Zeitspuren

Zur Steuerung des Rechenablaufs im LGP 21 dienen die vier Taktspuren C_p , S_1 , S_2 und S_3 . Diese werden bei S&F mit einem externen "Spuren-Schreibgerät" aufgeschrieben und dürfen danach nicht mehr verändert werden. Für jede Taktspur ist eine beschriebene Reservespur vorhanden (siehe Anordnung der Köpfe Bild 3.2). Die Taktspuren sind nur mit Leseverstärkern ausgerüstet. Die Köpfe sind normale Hauptspeicherköpfe.

Achtung!

Durch Ziehen der den Spuren zugeordneten Karten unter Spannung bzw. durch Löten an den Taktspurkreisen mit einem an Netzspannung liegenden LötKolben, können Störbits auf die Spuren geschrieben werden. Sie sind dann unbrauchbar!

2.2.5.1. Leseverstärker für S_1 , S_2 und S_3 (s. auch Bild 3.13)

Die Leseverstärker für die Spuren S_1 , S_2 , S_3 sind mit dem Leseverstärker für die Umlaufregister identisch (siehe Abschnitt 2.2.4.2).

2.2.5.2. Leseverstärker für den Grundtakt C_p (siehe auch Bild 3.13)

Das Lesesignal wird zunächst wieder in einem linearen Differentialverstärker verstärkt und dann über einen Resonanzkreis ($C19$, $R43$, $L1$) auf die Basis von $Ts13$ gegeben. Durch negative Signale wird $Ts13$ leitend. Sein Ausgangssignal wird mit $C21$ (680 pF) und $R47$ (470 Ω) differenziert. Die positiven Spitzen setzen das nachfolgende One Shot (monostabiler Multivibrator; Taktzeit min 3,5 μ s, max 5 μ s).

Der über $R49$, $R50$ normalerweise leitend gehaltene $Ts14$ sperrt bei einem positiven Impuls über $D14$. Über den Emitterfolger ($Ts15$) koppelt der negative Sprung am Kollektor von $Ts14$ auf die Basis von $Ts16$. $Ts16$ wird leitend. Über den Kondensator $C23$ (680 pF) wird der positive Sprung am Kollektor von $Ts16$ kapazitiv auf die Basis von $Ts14$ zurückgekoppelt. Das Sperren von $Ts14$ wird beschleunigt. Man erhält am Kollektor von $Ts14$ steile negative Flanken.

Ts14 bleibt solange gesperrt, bis das Potential an seiner Basis nach der Zeitkonstanten, gebildet aus $(R49 + R50)$. C23 wieder negativ geworden ist. Die Zeitkonstante und damit die Sperrzeit von Ts14 kann mit dem Potentiometer R49 variiert werden. Die Zeit sollte ca. 4 bis 4,5 μ sec betragen. Das Ausgangssignal Cp des One-Shots dient zur Ansteuerung der nachfolgenden Taktverstärker auf den einzelnen Karten.

Achtung!

Der Ausgang Cp des One-Shots ist gegen 0V und positive Spannungen nicht kurzschlußfest.

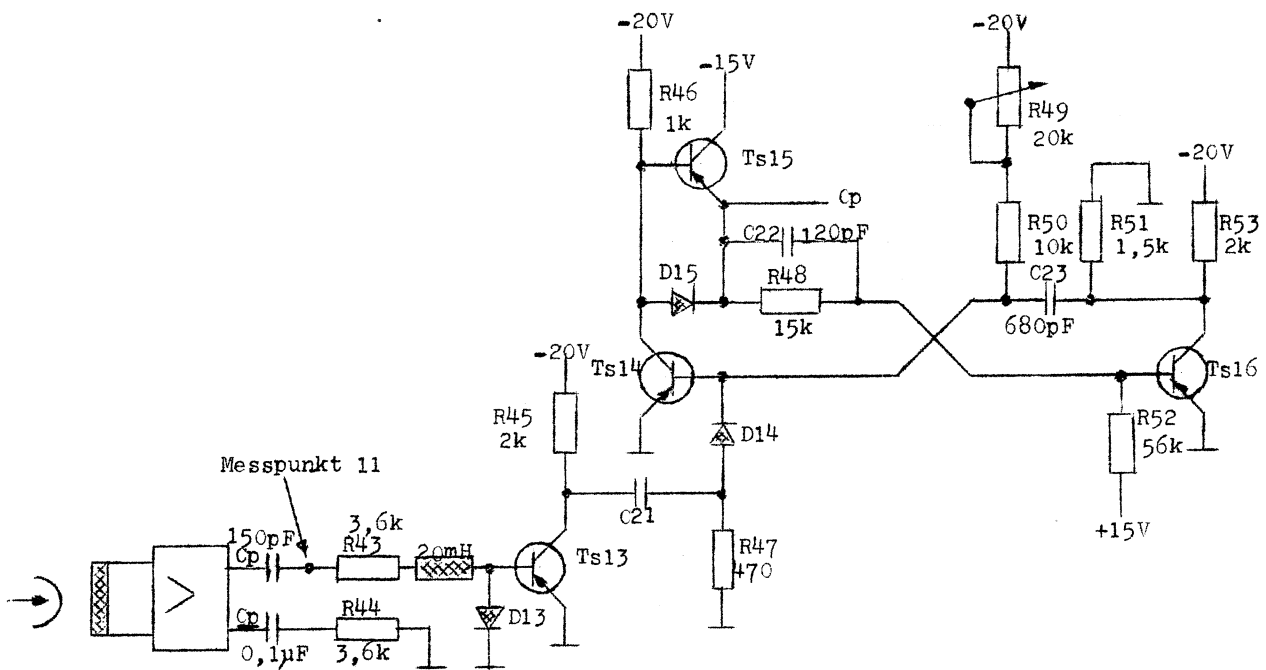


Bild 2.12

2.2.5.3. Takt-Leistungsverstärker

Der nachfolgende Taktverstärker besteht aus einem Emitterfolger und einem niederohmigen Inverter (Bild 2.13).

Er wird zur Ansteuerung der einzelnen Flip-Flops benutzt (max 5 Flip-Flops pro Verstärker).

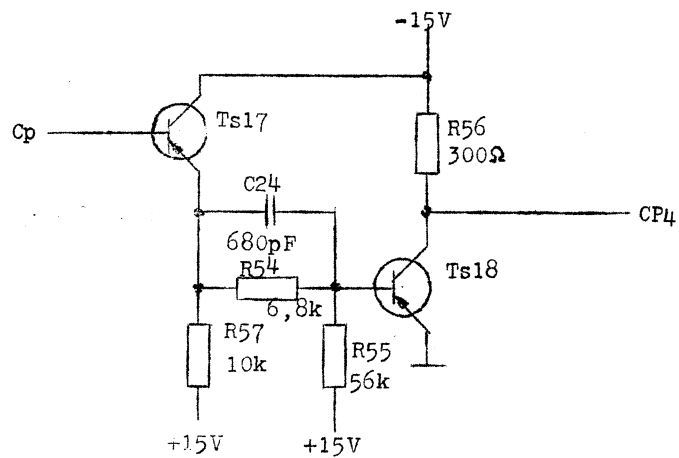


Bild 2.13

2.3. Grundsaltungen

2.3.1. Flip Flop

Das Flip Flop dient zur Informationsspeicherung. Es ist ein bistabiler Multivibrator, der durch Setzimpulse von einem in den anderen seiner zwei stabilen Zustände gekippt werden kann.

Das Flip Flop wird mit einer logischen "1" (=0V) gesetzt. Zur Setz- bzw. Rücksetzzeit müssen beide Eingänge A' , \underline{A} entgegengesetzte Wertigkeiten haben. Falls keine Setzbedingungen gegeben sind, liegen beide Eingänge auf logisch "0" (= -10V...-20V).

Die zwei Ausgänge A , \underline{A} des Flip Flops haben grundsätzlich entgegengesetzte Wertigkeit.

$A = 0V = \text{logisch "1"}^H$ $\underline{A} = -20V = \text{logisch "0"}^L$
 $A = -20V = \text{logisch "0"}^L$ $\underline{A} = 0V = \text{logisch "1"}^H$

Das Flip Flop kann nur zur Taktzeit gesetzt werden und auch nur dann, wenn das Setzsignal am Eingang A' , \underline{A} ca. 3,3 bzw. 2 μsec vor dem Takt ansteht. Diese Eingangszeitkonstanten sollen das Flip Flop störunanfälliger machen.

Zwei Flip Flop-Typen finden im Rechner Verwendung. Sie unterscheiden sich nur in ihren Eingangszeitkonstanten von 3,3 μsec bzw. 2 μsec und der damit verbundenen Schaltungsänderung (unterschiedliche Eingangswiderstände, siehe Bild 2.14).

Arbeitsweise: (Bild 2.14)

Annahme: Das Flip Flop ist im eingeschwungenen Zustand. Der Transistor Ts2 ist leitend (Ts1 also gesperrt). Damit ist das Basispotential von Ts2 mit ca. 0,5V und das von Ts1 mit ca. +1V gegeben.

\underline{A} wird nun z.B. 0V (logisch "1"). An der Anode von D2 liegt folglich ein Potential von 0V. Infolge des Spannungsabfalls von ca. 0,6V an Dioden in Durchlaßrichtung, bleibt D2 noch gesperrt.

Die Anode von D1 liegt auf einem Potential von $-12V\dots -20V$, da am Eingang A' keine Setzbedingung vorliegt. D1 ist sicher gesperrt.

Über die Kondensatoren C1 und C2 wird der Spannungssprung des Taktes CP auf das Potential der Anoden von D1 und D2 differenziert. Bei einem positiven Spannungssprung wird D2 leitend und sperrt Ts2. D1 bleibt gesperrt, weil der Spannungssprung nicht größer als $12V$ werden kann, da die Amplitude von CP auf $12V$ begrenzt ist.

Über den Kopplungswiderstand R2 wird Ts1 leitend. Bei leitendem Ts1 wird Ts2 über den Kopplungswiderstand R1 gesperrt gehalten.

Die Kondensatoren C1 und C2 dienen nur zur Beschleunigung.

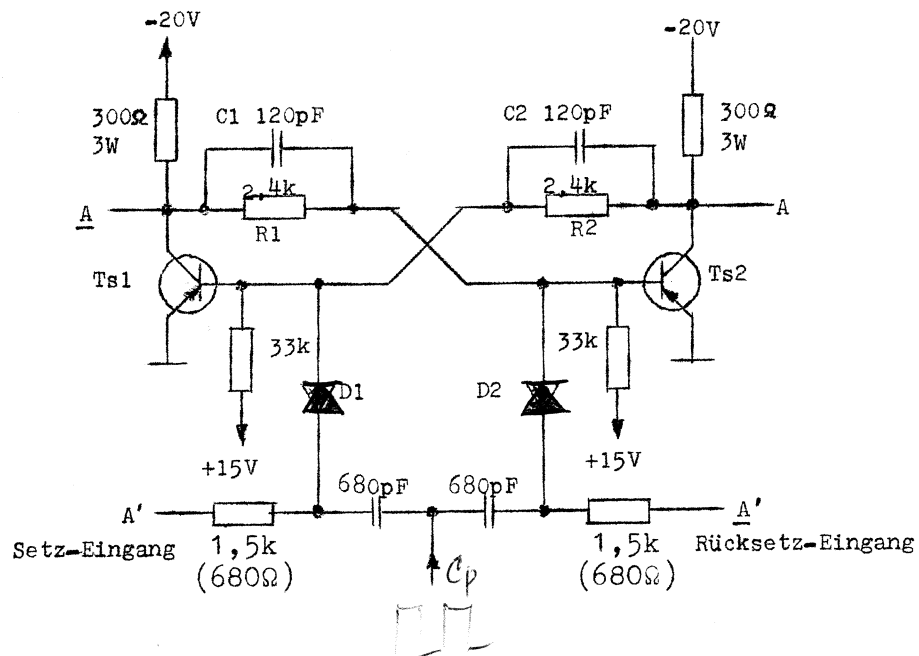


Bild 2.14

2.3.2. Nor-Schaltung

Die Nor-Schaltung entspricht einem ODER-Gatter mit Inverter für negative Signalrichtung, einem UND-Gatter mit Inverter für positive Signalrichtung.

Sein Ausgang ist logisch "1" (=0V), wenn mindestens ein Eingang logisch "0" (= -20V) ist.

Sein Ausgang ist logisch "0" (-20V), wenn alle Eingänge logisch "1" (=0V) sind.

Die Eingangspegel müssen in einem Bereich von -6V bis -20V bei log. "0" und -0,1...-0,6V bei log. "1" liegen. Die Ausgangsamplitude ist von der jeweiligen Belastung abhängig. Sie liegt zwischen -6V bis -20V.

Im LGP 21 werden zwei verschiedene Nor-Gatter-Typen eingesetzt. Sie unterscheiden sich in ihrer Ausgangsimpedanz und der damit verbundenen maximalen Eingangszahl (Bild 2.15).

Das niederohmige Nor-Gatter dient zur Einspeisung einer größeren Zahl anderer Gattereingänge (maximal 15 Stück). Die Zahl der Eingänge für dieses Gatter ist auf max 5 begrenzt.

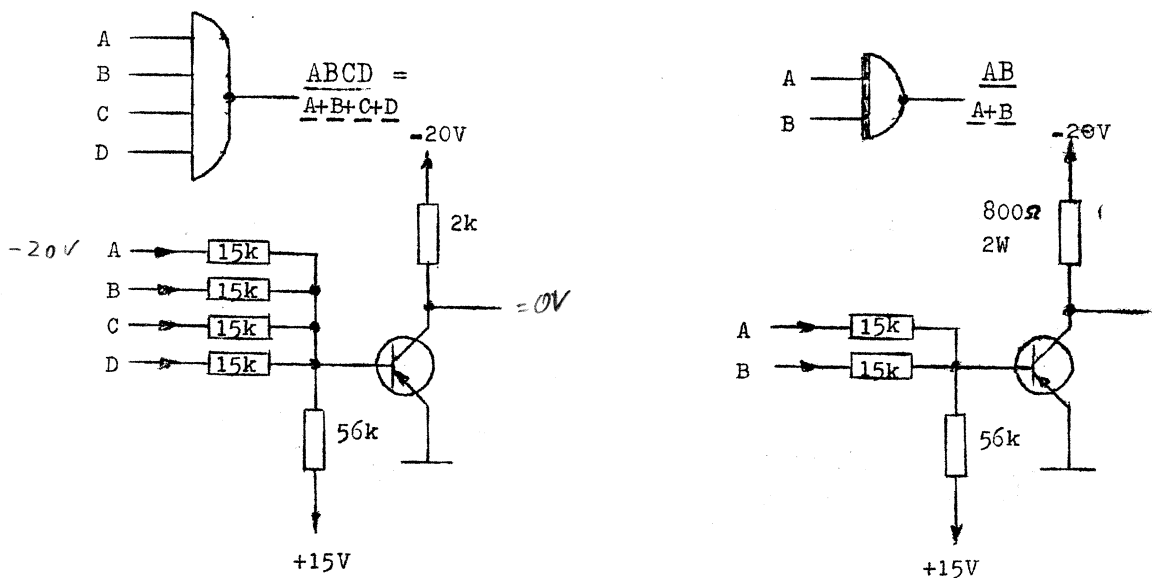


Bild 2.15

2.3.3. Diodengekoppelter Inverter

Der diodengekoppelte Inverter ist ein Und-Gatter mit Inverter für negative Signalrichtung, ein ODER-Gatter mit Inverter für positive Signalrichtung.

Sein Ausgang ist logisch "0" (= -20V), wenn wenigstens ein Eingang logisch "1" (=0V) ist.

Sein Ausgang ist logisch "1" (=0V), wenn alle Eingänge logisch "0" (= -20V) sind.

Der diodengekoppelte Inverter hat eine ca. 10 mal schnellere Schaltzeit als das Nor-Gatter.

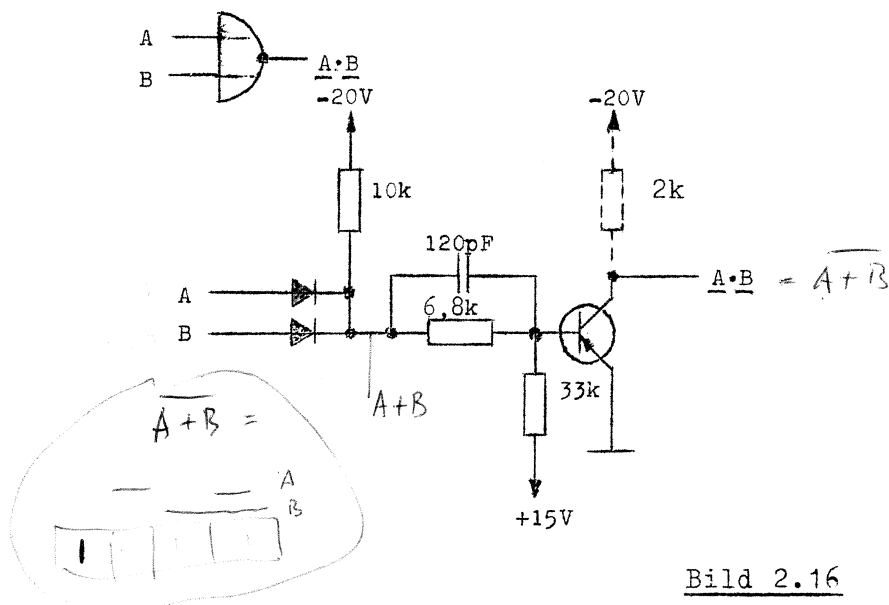


Bild 2.16

2.3.4. Emitterfolger

Der Emitterfolger ist ein Leistungsverstärker mit einer Spannungsverstärkung < 1 .

Das Eingangssignal wird nicht invertiert!

Der Emitterfolger ist niederohmig gegen -15 Volt und wird deshalb zur Einspeisung niederohmiger Eingangsstufen bzw. einer größeren Anzahl von Folgestufen verwendet.

Achtung!

Der Emitterfolger ist gegen 0V und positive Spannungen nicht kurzschlußfest.

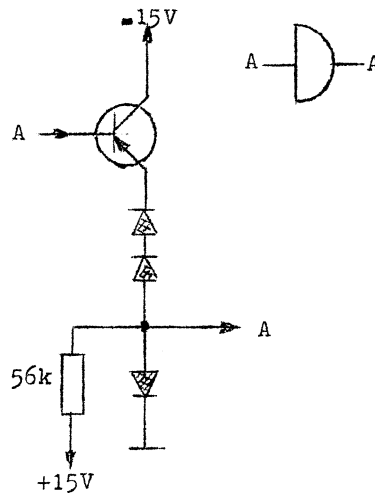


Bild 2.17

2.3.5. One Shot (Monostabiler Multivibrator)

Normalerweise sind die Steuersignale für die elektro-mechanischen Ein-/Ausgabegeräte im Verhältnis zu den internen Rechnersignalen erheblich länger. Die One-Shots werden deshalb hauptsächlich für eine Impulsverlängerung eingesetzt.

Im Ruhezustand ist der Ausgang A logisch "1" (=0V) und A logisch "0" (-20V). Mit der positiven Flanke eines Eingangssignals an A* wird das One Shot für eine definierte Zeit gesetzt, d.h. sein Ausgang A wird logisch "1" (0V), sein Ausgang A wird "0" (-20V). Danach kippt es entsprechend seiner Zeitkonstanten in den Ruhezustand zurück.

Mit einem Potentiometer (R1) kann die Zeitkonstante variiert werden.

Im Ruhezustand wird Ts1 über R1, R6 leitend gehalten. Sein Basispotential liegt bei ca. -0,5V. Ts2 ist entsprechend seinem Basispotential von +3V gesperrt (Spannungsteiler Ts1, R4, R9). Zur Störunterdrückung ist D1 mit ca. 7 Volt durch den Spannungsteiler R5, R3 gesperrt.

Die Setzimpulse an A* werden über C2 R3 differenziert. Durch positive Spitzen > 8V wird D1 leitend. Die Basis von Ts1 wird dadurch positiv und Ts1 sperrt.

Dadurch wird Ts2 leitend und der Kondensator C1 wird umgeladen. Die Folge ist, daß an der Basis von Ts1 entsprechend dem Spannungssprung an C1 ein positives Potential steht.

Dieses kann nur über R1 und R6 abgebaut werden, und Ts1 bleibt damit entsprechend der Zeit $(R1+R6) \cdot C1$ gesperrt. Wird Ts1 wieder leitend, wird Ts2 gesperrt und das One Shot befindet sich wieder in Ruhelage.

Die Diode D3 liegt beim Sperren von Ts2 in Sperrichtung. Das Kollektorpotential von Ts2 ist also unabhängig von der Umladezeitkonstanten von C1. Die negative Flanke bleibt steil.

Da sich der Kondensator C1 infolge der gesperrten Diode D3 nicht wie bei einem normalen One Shot über R11 umladen kann, mußte R8 und D2 eingefügt werden. Damit die Erholzeit des One Shots (Umladung von C1 auf ein definiertes Potential) nicht übermäßig lang wird, begrenzt der Spannungsteiler R7 R8 die negative Spannung auf ca. -8,5V.

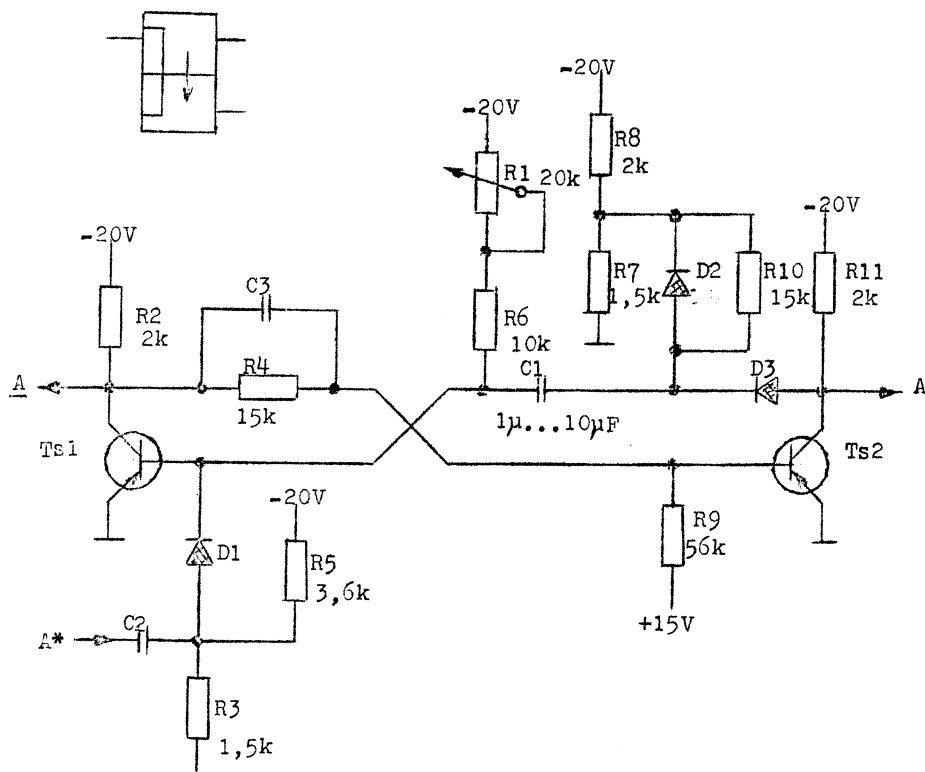


Bild 2.18

2.3.6. Astabiler Multivibrator

Der astabile Multivibrator (Bild 2.20) kippt selbständig in einem bestimmten Rhythmus zwischen den beiden möglichen Schaltstellungen.

Die Kippzeiten T_1 und T_2 sind bestimmt für:

T_1 durch die Zeitkonstante $C_1 \cdot (R_1 + R_3)$

T_2 durch die Zeitkonstante $C_2 \cdot (R_2 + R_{10})$

Die Spannung $-20d$ verhindert ein Anlaufen des Multivibrators beim Einschalten des Rechners. Während dieser Zeit hat der Ausgang A ständig $OV = "1"$.

Wird Ts_2 leitend, so wird über C_1 ein positiver Spannungssprung gekoppelt und Ts_1 gesperrt. Diese positive Spannung an seiner Basis wird über R_1 und R_3 solange abgebaut, bis Ts_1 seinen Basisstrom über R_3 und R_1 wieder ziehen kann. Dadurch wird Ts_1 leitend und sperrt über D_1 und C_2 den Transistor Ts_2 . Ts_2 bleibt solange gesperrt, bis das positive Potential an seiner Basis über R_{10} und R_2 abgebaut ist. Dann wird Ts_2 wieder leitend und der Vorgang läuft erneut wie oben beschrieben ab.

Die Dioden D_1 und D_2 liegen beim Sperren des zugehörigen Transistors in Sperrrichtung. Die Kollektorpotentiale sind also unabhängig von den Umladezeitkonstanten für C_1 und C_2 . Die negativen Flanken bleiben steil.

Damit die Erholzeit des Multivibrators (Umladung von C_1 und C_2 auf definierte Potentiale) nicht zu lang wird, begrenzt der Spannungsteiler R_6 R_8 und R_5 R_7 die negative Spannung auf ca. $8V$.

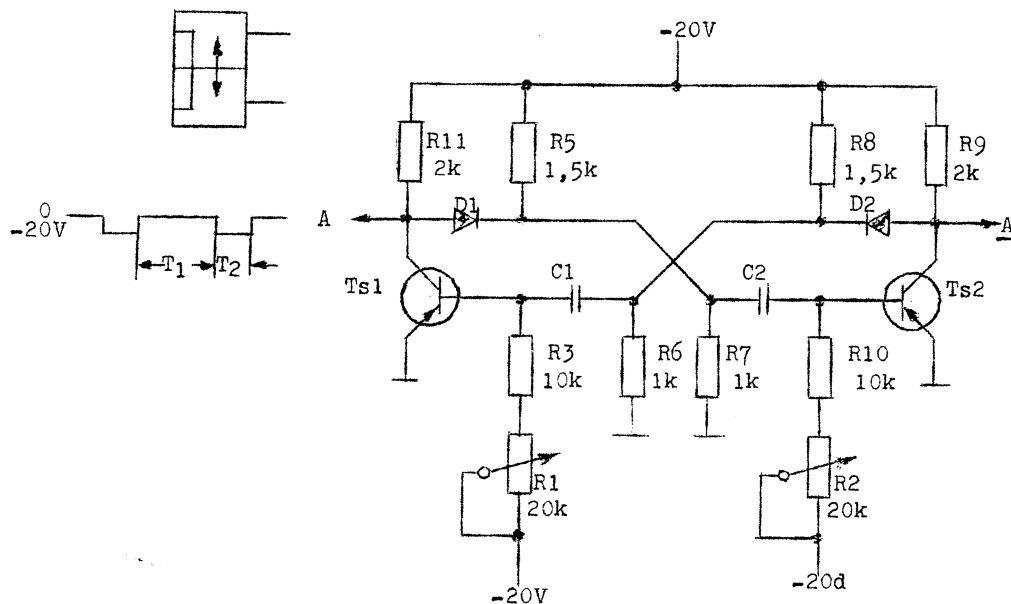


Bild 2.20

2.3.7. Translator-Magnet-Treiber

Der Translator-Magnet-Treiber arbeitet ähnlich wie ein Flip Flop. Er muß gesetzt und zurückgesetzt werden.

Sind die Eingänge A und B logisch "1", ist auch der Ausgang Tp logisch "1".

Der Treiber kann nur durch das Eingangssignal C logisch "0" zurückgesetzt werden.

Im Ruhezustand ist Ts1 leitend, weil wenigstens ein Eingang A oder B negativ ist.

Wenn A oder B (bzw. A und B) negativ sind, ist die Diode D2 leitend, die Basis von Ts1 negativ und Ts1 leitet. Die Basis von Ts2 liegt über den Spannungsteiler Ts1, R7, R6 auf einem positiven Potential von ca. +1,6V. Damit sind Ts2 und Ts3 gesperrt. Über den Spannungsteiler R10, R5, R8 ist D1 mit ca. -4V gesperrt.

Wenn A und B logisch "1" werden, wird Ts1 gesperrt.

Ts2 kommt über den Spannungsteiler R9, R7, R6 zum Leiten. R10 ist wesentlich niederohmiger als R11. Am Emitter von Ts2 stellt sich negatives Potential ein. Damit wird Ts3 leitend. Über den Spannungsteiler R5, R8 und D1 wird D2 gesperrt. Ts1 bleibt gesperrt bis über den Eingang C negatives Potential an die Basis von Ts1 gelegt wird. Der Translator-Magnet-Treiber ist damit wieder zurückgesetzt.

D3 soll verhindern, daß sich am Emitter von Ts2 ein positiveres Potential als +1V einstellt (1V reicht zum Sperren von Ts3), weil sonst Ts2 nicht über den Spannungsteiler R7, R6 gesperrt werden kann.

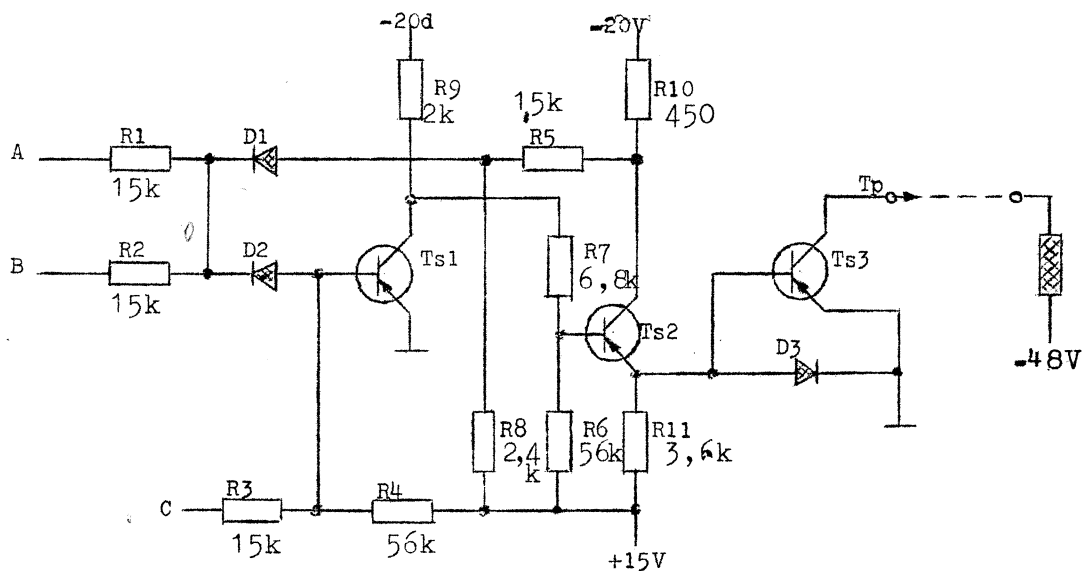


Bild 2.21

2.3.8. Translator-Kupplungs-Treiber

Der Translator-Kupplungs-Treiber dient als Treiber für die Translator-Kupplung.

Die Stufe wird zur Taktzeit CP und Eingang A logisch "1" gesetzt und zur Taktzeit CP und Eingang B logisch "0" zurückgesetzt.

Im Ruhezustand ist Ts1 über den Spannungsteiler R11, R3, R4 leitend und damit Ts2 sowie Ts3 über den Spannungsteiler Ts1, R6, R9 gesperrt. Wird A 0V, ist D2 noch gesperrt. Die folgende positive Flanke von CP öffnet D2 und sperrt Ts1.

Der negative Sprung an seinem Kollektor koppelt über C3, R6 auf die Basis von Ts2. Ts2 und Ts3 werden leitend. Die Stufe verharrt in diesem stabilen Zustand (siehe auch Abschnitt 2.3.1.).

Ts2 und Ts3 werden gesperrt. Der negative Sprung an seinem Kollektor koppelt über C2, R3 auf die Basis von Ts1, der somit leitend wird. Auch dieser Zustand ist stabil.

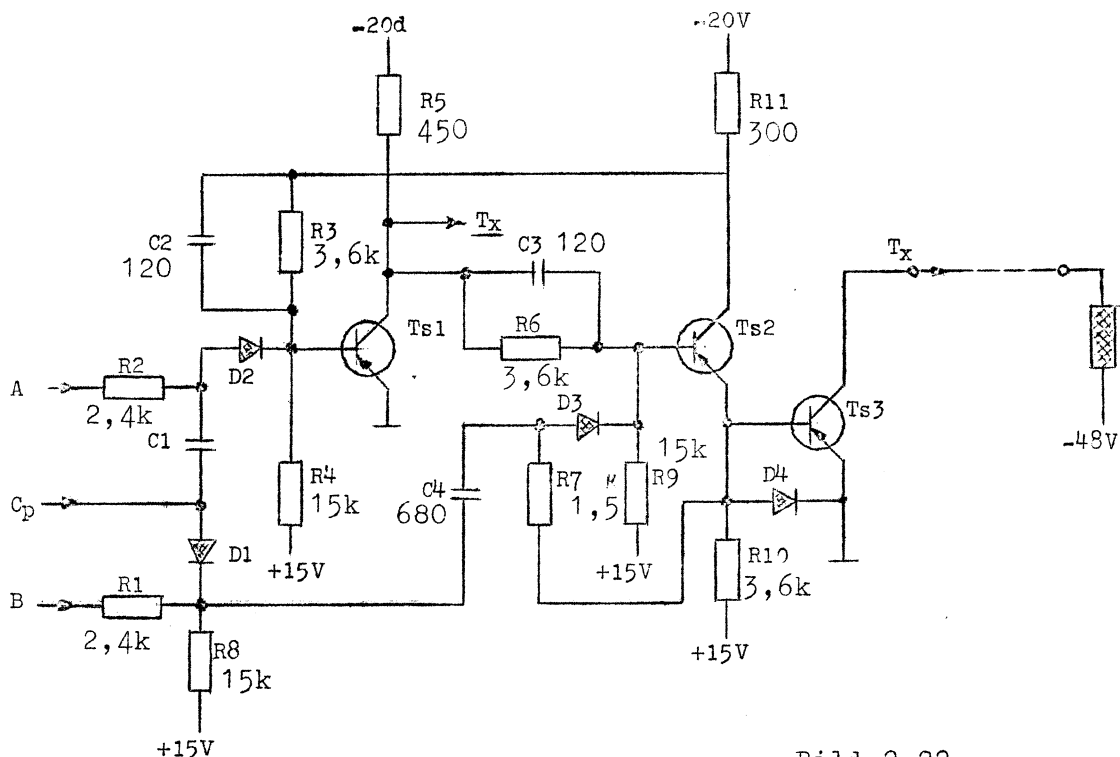


Bild 2.22

2.3.9. Leistungs-Treiber

Der Kupplungstreiber ist ein Leistungsverstärker. Das Eingangssignal wird nicht invertiert. Die Stufe besteht aus Inverter, Emitterfolger, Inverter. Die Dioden D1 und D2 sollen bei leitendem Ts1 das Sperren von Ts2 gewährleisten. Ist Ts1 gesperrt, so ist nur R3 für die Basisstrombegrenzung von Ts2 maßgebend.

D3 soll verhindern, daß sich am Emitter von Ts2 ein positiveres Potential als +1V einstellt (1V reicht zum Sperren von Ts3), weil sonst Ts2 nicht über den Spannungsteiler R5, R6 gesperrt werden könnte. (Bild 2.23)

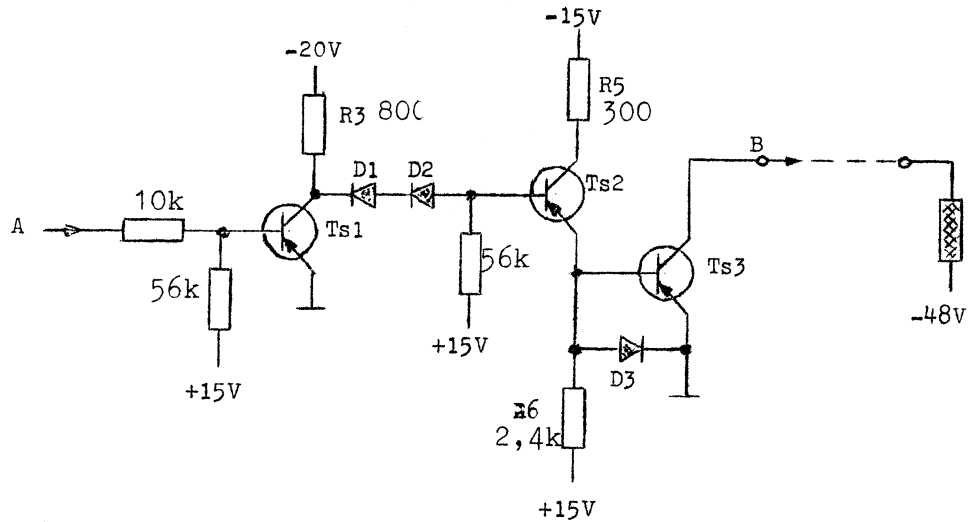


Bild 2.23

2.3.10. Treiber

Der Translator-Treiber ist ein Leistungsverstärker. Der Ausgang C ist nur leitend, wenn gleichzeitig A und B logisch "1" sind.

Die Stufe besteht aus einer Nor-Schaltung und einem Inverter.

Die Diode soll bei leitendem Ts1 ein sicheres Sperren von Ts2 gewährleisten. Ist Ts1 gesperrt, so ist nur R1 für die Basisstrombegrenzung von Ts2 maßgebend.

(Bild 2.24)

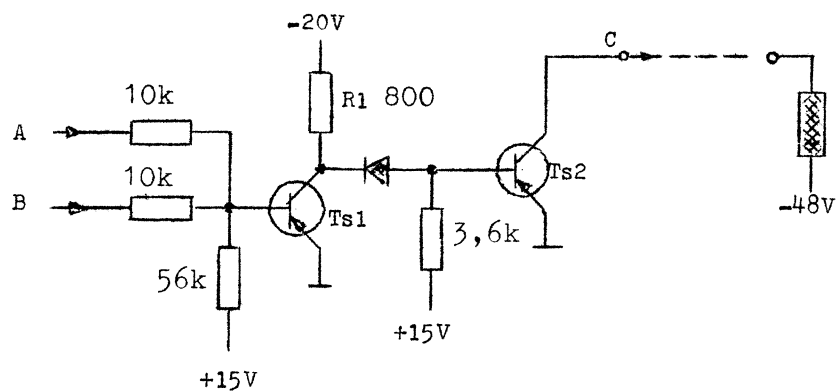


Bild 2.24

2.4. Zeitsteuerung

2.4.1. Zeitmarkierungen

Die Organisation der Worteinteilung pro Scheibenumdrehung erfolgt durch 4 Taktspuren, aus denen durch Und- und Oder-Schaltungen weitere Zeiteinteilungen gewonnen werden (Bild 2.25).

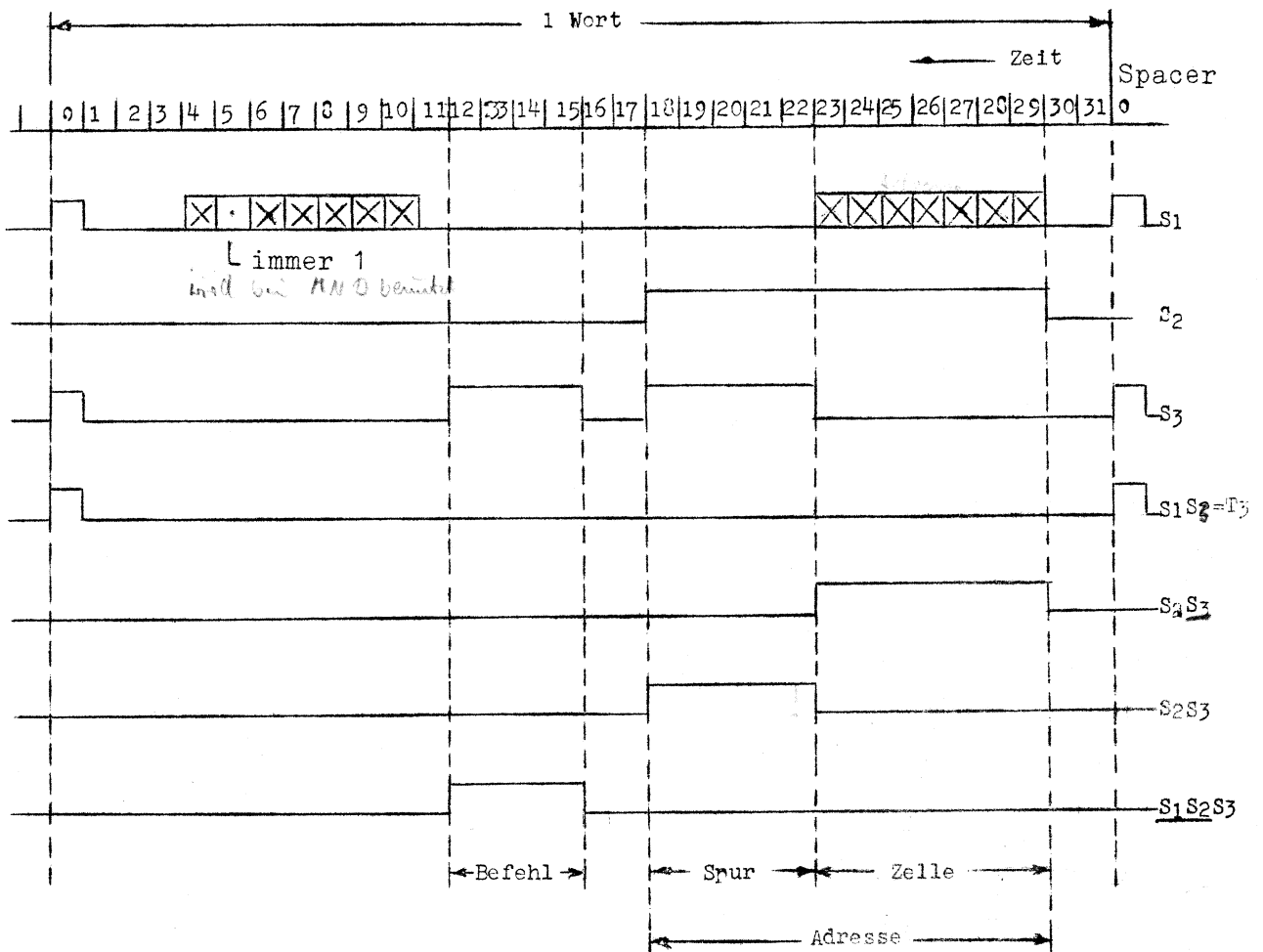


Bild 2.25

Jede Spur ist in 128 Sektoren eingeteilt, wobei die fortlaufende Numerierung der Sektoren in der S1-Spur enthalten ist.

Die optimalen Adressen siehe Optimalisierungsscheibe.

2.4.2. Phasen ($\emptyset 1 \dots \emptyset 4$)

Der Zyklus für die Ausführung eines jeden Befehls hat 4 Phasen ($\emptyset 1 \dots \emptyset 4$). In Bild 2.26 sind die Aufgaben der einzelnen Phasen dargestellt.

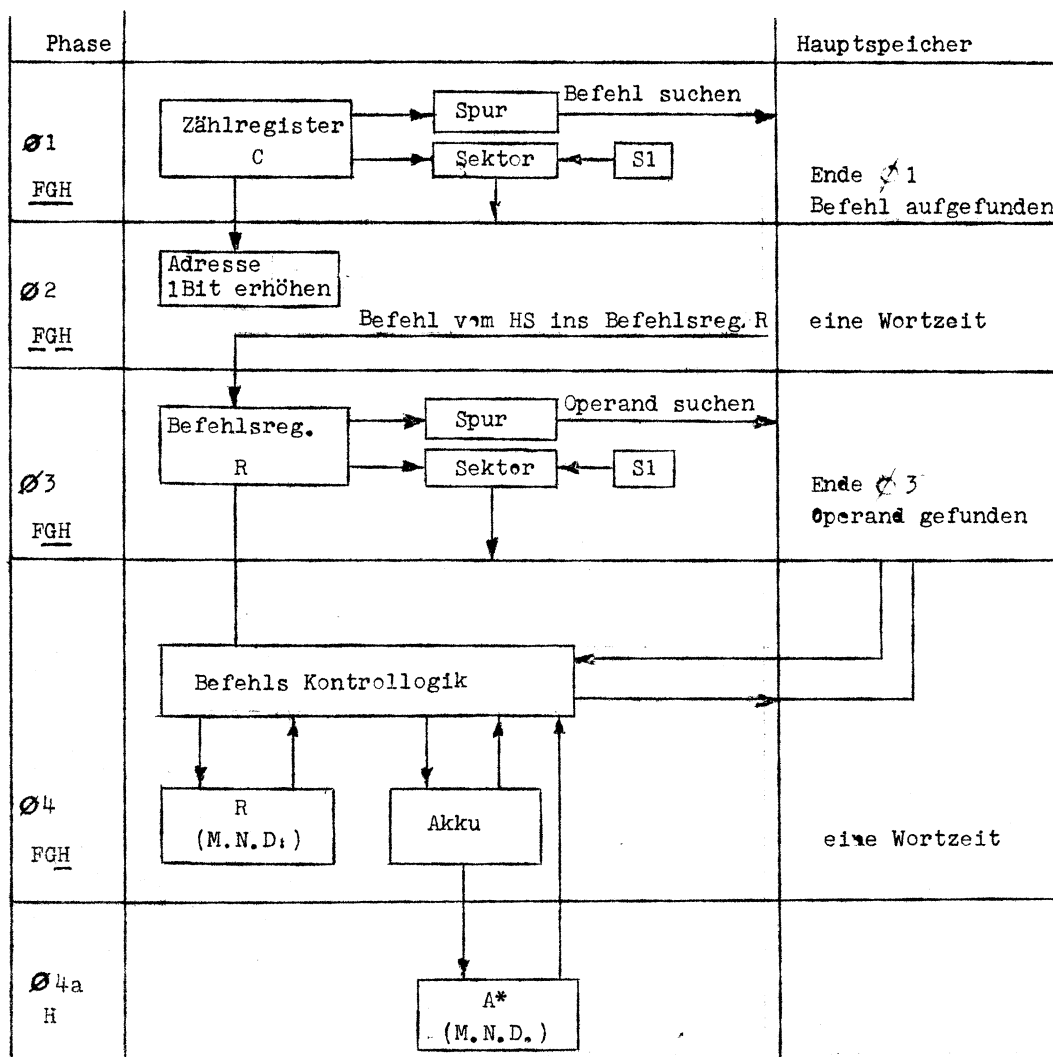


Bild 2.26

Die vier Phasen werden mit den Flip-Flops F, G und H festgelegt. Das H-Flip-Flop ist nur während der verlängerten $\emptyset 4$ gesetzt.

$$\emptyset 1 = \underline{F} \underline{G} \underline{H}$$

$$\emptyset 2 = \underline{F} \underline{G} \underline{H}$$

$$\emptyset 3 = \underline{F} \underline{G} \underline{H}$$

$$\emptyset 4 = \underline{F} \underline{G} \underline{H}$$

$$\emptyset 4a = \dots \underline{H}$$

Im Normalfall erfolgt der Phasenablauf immer in dem oben angegebenen Zyklus (Besonderheiten s. einzelne Befehle).

$\emptyset 1$ und $\emptyset 3$ dauert bis zum erfolgreichen Vergleich der Sektoradresse (max 1 Scheibenumdrehung).
 $\emptyset 2$ und $\emptyset 4$ dauert eine Wortzeit. Bei der Multiplikation und Division wird die $\emptyset 4$ verlängert in $\emptyset 4a$ (Zeitdauer siehe M, N, D - Befehl).

2.4.2.1. Phase 1 ($\emptyset 1$)

In $\emptyset 1$ wird der Befehl gesucht, dessen Adresse im C-Register umläuft.

In jeder Wortzeit werden die Bits, die im C-Register während der Sektoradresezeit $S2$, $S3$ erscheinen, mit der Sektornummer, die von $S1$ gelesen wird, Bit für Bit verglichen.

Bei Übereinstimmung aller 7 Bits ist der nächste Sektor die gewünschte Adresse. Damit wird $\emptyset 1$ beendet.

Zum Vergleich wird das K-Flip-Flop benutzt.

K wird zu Beginn einer jeden Wortzeit gesetzt.

Falls auch nur ein Bit von C und $S1$ während der Adresezeit $S2$, $S3$ nicht übereinstimmen, wird K zurückgesetzt.

Bei Übereinstimmung aller Bits bleibt K über die ganze Wortzeit gesetzt. Damit wird $\emptyset 1$ beendet und G gesetzt.

$$K' = T3 \underline{Faf} (\underline{F} + \dots)$$

└ keine Ein- oder Ausgabe

$$\underline{K'} = \underline{G} \underline{H} \underline{S2} \underline{S3} \underline{Faf} (\underline{S1} \underline{r1} + \underline{S1} \underline{r1})$$

$\emptyset 1(\emptyset 3)$ └ Sektoradresezeit

$$r1 = \underline{F} C + \dots$$

$S1 \underline{r1} + \underline{S1} r1 = \dots \underline{C} S1 + \dots C \underline{S1}$, d.h. dieser Ausdruck ist nur bei Nichtübereinstimmung von C und $S1$ logisch "1".

$$G' = \underline{G} \underline{H} T3 K Q2 \quad (\text{damit beginnt } \emptyset 2)$$

└ kein stop

Der Umlauf in $\emptyset 4$ ist ausgeklammert. Er ist von dem jeweiligen Befehl abhängig.

Die Adresse des gerade gefundenen Befehles steht noch im C-Register. Sie wird jetzt, vorbereitend für die nächste $\emptyset 1$, um 1 erhöht.

$$C'w = \underbrace{\underline{F} \underline{G} \underline{H}}_{\emptyset 2} \underline{S2} \left(\underline{K} \underline{C} + \underline{K} \underline{C} \right) + \dots$$

|
Adreßzeit

$$K' = T3 \underline{Faf} (\underline{F} + \dots)$$

$$\underline{K}' = \underline{F} \underline{G} \underline{H} \underline{S2} \underline{Faf} \underline{C} + \dots$$

K wird zu Beginn der $\emptyset 2$ gesetzt.

K wird zurückgesetzt, sobald eine "0" in der Adresse im C-Register erscheint. Hiermit ist eine Addition um 1 erreicht.

Die neue (Folge-)Adresse läuft in C während $\emptyset 3$, $\emptyset 4$, $\emptyset 1$ um:

$$C'w = \underbrace{\underline{brc} \underline{S2} \underline{G} \underline{C}}_{\text{Umlauf der Adressen in } \emptyset 1 \text{ und } \emptyset 3} + \underline{brc} \underline{S2} \underline{F} \underline{C} \underbrace{(\underline{Q1} + \underline{Q2} + \underline{Q3} + \underline{Q4})}_{\emptyset 4 \text{ alle Befehle außer U}}$$

$\emptyset 2$ ist nach einer Wortzeit beendet. F wird gesetzt und G zurückgesetzt. Damit beginnt $\emptyset 3$.

$$\underline{G}' = G \underline{F} T3$$

$$F' = G \underline{F} \underline{H} T3 \underline{Q1}$$

2.4.2.3. Phase 3 ($\emptyset 3$)

In Phase 3 wird die Adresse des im R-Register stehenden Befehles gesucht.

In jeder Wortzeit werden die Bits, die im R-Register während der Sektoradreßzeit $S2$ $\underline{S3}$ erscheinen, mit der Sektornummer, die von $S1$ gelesen wird, Bit für Bit verglichen.

Bei Übereinstimmung aller 7 Bits ist der nächste Sektor die gewünschte Adresse. Damit wird $\emptyset 3$ beendet.

Zum Vergleich wird das K-Flip-Flop benutzt. K wird zu Beginn jeder Wortzeit gesetzt. Falls auch nur ein Bit von R und S1 während der Adresszeit S2 S3 nicht übereinstimmen, wird K zurückgesetzt.

Bei Übereinstimmung aller Bits bleibt K über die ganze Wortzeit gesetzt.

$$K' = T3 \underline{Faf} (\underline{F} + \dots)$$

└ keine Ein- oder Ausgabe

$$\underline{K'} = \underline{G} \underline{H} \underline{S2} \underline{S3} \underline{Faf} (\underline{S1} \underline{r1} + \underline{S1} \underline{r1})$$

($\emptyset 1$) $\emptyset 3$ Sektoradreßzeit

$$r1 = \underline{F} \underline{H} \underline{R} + \text{-----}$$

$\emptyset 3$

$\underline{S1r1} + \underline{S1r1} = \dots \underline{C} \underline{S1} + \dots \underline{C} \underline{S1}$, d.h. dieser Ausdruck ist nur bei Nichtübereinstimmung von R und S1 logisch "1"

$$G' = \underline{G} \underline{H} T3 K Q2 \quad (\text{Damit beginnt } \emptyset 4)$$

└ kein Stop

Um auch die ausgewählte Spur noch vor Beginn der $\emptyset 4$ zu speichern, werden die 5 Bits aus dem R-Register während der Spurzeit S2 S3 in die P-Flip-Flops gegeben. Dazu werden die P-Flip-Flops als Shift-Register geschaltet.

Das auch im Shift-Register enthaltene P6 wird nur bei der Ein-Ausgabe benötigt.

Die Bits von R werden nach P1 gegeben und eine Bitzeit später nach P2 geschiftet usw. In P5 steht das meistbedeutende Bit.

$$Aw' = A \underline{H} \underline{To} \left[\underline{F} + \underline{G} + Q1 \underbrace{Q3 \quad Q4}_{\substack{\text{E/A-Taste nicht} \\ \text{gedrückt}}} + Q2 (Q3+Q4) + \dots \right] + \dots$$

Umlauf: $\emptyset 1 \dots \emptyset 3$ P-,H-Befehl U-,T-,Y-,R-,Z-Befehl

Für alle anderen Befehle ist die A-Logik erweitert. Sie wird bei den einzelnen Befehlen besprochen.

$\emptyset 4$ ist für alle Befehle außer M, N und D nach einer Wortzeit beendet, und der Rechner geht wieder in $\emptyset 1$.

F und G werden zurückgesetzt:

$$\underline{F}' = G \underline{F} \underline{H} T3$$

$$\underline{G}' = G \underline{H} T3$$

2.5. Funktion der arithmetischen Befehle

Die nachfolgende Befehlslogik beschreibt die zur Ausführung der einzelnen Befehle benötigten Funktionen (s. auch Programmieranleitung LGP 21).

2.5.1. Z-Befehl

$$Z \hat{=} \underline{Q1} \underline{Q2} \underline{Q3} \underline{Q4} \hat{=} 0000$$

Beim Z-Befehl mit der Spuradresse 0000 oder 0100 erfolgt ein Stop (0100 nur bei der Normalausführung, s. auch Modell 81 und Vorrangsteuerung).

Beim Z-Befehl mit der Spuradresse 0200 oder 0300 erfolgt ein normaler Phasenablauf. Der Befehl hat sonst keine Funktion.

Beim Z-Befehl mit der Spuradresse 0300 wird der Zustand der PS-Tasten abgefragt. Bei gedrückter PS-Taste wird der nächste Befehl ausgeführt. Ist eine PS-Taste nicht gedrückt und das zugehörige P-Flip-Flop (Spuradresse) gesetzt, so wird der nächste Befehl übersprungen.

Beim Z-Befehl wird außer der normalen Z-Funktion der Überlauf getestet.

2.5.1.1. Z 0000 und Z 0100

In $\emptyset 2$ wird der Z-Befehl ins R-Register gebracht. Falls P1 gesetzt ist, erfolgt die Rücksetzung.

$$\underline{P1'} = G \ T3 \ P1 \ P1b$$

└─ Faf

$\emptyset 3$ ist nach einer Wortzeit beendet, da das Suchen eines Operanden entfällt. G wird gesetzt.

$$G' = \underbrace{F \ G \ T3}_{\emptyset 3} \ \underbrace{\underline{Q2} \ \underline{Q3} \ \underline{Q4}}_{Z-(P)\text{Befehl}} \ \underline{be} \ \underline{Faf}$$

Zur Spuradresszeit (S2 S3) der $\emptyset 3$ wird die Spuradresse des Z-Befehls aus dem R-Register in die P-Flip-Flops geschiftet.

Die Schiftzeit beträgt nur 5 Bitzeiten. Damit wird P6 immer 0 (P1 wurde in $\emptyset 1$ zurückgesetzt).

Der Z 0100 wird in den P-Flip-Flops zum Z 0000.

Nach der folgenden $\emptyset 4$ (1 Wortzeit) bleibt der Rechner in $\emptyset 1$ stehen. Das G-Flip-Flop wird nicht gesetzt. Das Setzen von G wird durch Q2 entschieden.

$$G' = \underline{G} \underline{H} T3 K Q2 \underline{be} \underline{bs} Ga$$

└─Faf

Zur T3-Zeit der $\emptyset 4$ wird Q2 für alle Befehle außer Z 0000 und Z 0100 gesetzt:

$$Q2' = \underbrace{F G T3}_{\emptyset 4} \underline{Q2} \underline{01} \underbrace{(Q1+Q2+Q3+Q4+P1+P2+P3+P4+P5+P6)}_{\text{außer Z0000, Z0100}}$$

kein Step, kein Manuell

Durch Drücken der "Start"-Taste wird der Stop aufgehoben. Q2 wird wieder gesetzt.

$$Q2' = \underbrace{F G H}_{\emptyset 1} bs$$

└─"Start"

2.5.1.2. Z-Befehl mit Spuradressen > 3

Q1 entscheidet in $\emptyset 2$ über die Ausführung des Z-Befehls. Ist Q1 in $\emptyset 2$ gesetzt, wird der nächste Befehl übersprungen. F kann nicht gesetzt werden. Die folgende $\emptyset 3$ wird nicht ausgeführt. Da am Ende von $\emptyset 2$ G jedoch wie immer zurückgesetzt wird, läuft der Rechner wieder in $\emptyset 1$. Durch die Adressenerhöhung in $\emptyset 2$ ist jetzt ein Befehl übersprungen worden.

$$F' = \underline{F} \underline{G} \underline{H} \underline{Q1} T3 + \dots$$

Q1 wird am Ende von $\emptyset 2$ und $\emptyset 4$ zurückgesetzt. Damit beginnt wieder ein normaler Phasenablauf.

$$\underline{Q1}' = \underbrace{G H}_{\emptyset 2, \emptyset 4} Q1 T3 + \dots$$

Die Spuradresse des Befehls steht in $\phi 2$ in den P-Flip-Flops. Die Setzbedingung für Q1 lautet also:

$$Q1' = \underbrace{G \ H}_{\phi 2(\phi 4)} \underbrace{Q1 Q2 Q3 Q4}_{Z\text{-Befehl}} T3 \underbrace{(P1 Tb1 + P2 Tb2 + P3 Tb3 + P4 Tb4 + \dots)}_{PS\text{-Tasten}} + \dots$$

2.5.1.3.-Z-Befehl

Ist bei einem -Z-Befehl durch den vorhergehenden Befehl ein Überlauf vorhanden (das C-Register ist negativ), wird das Überlauf-Bit zurückgesetzt und der nächstfolgende Befehl ausgeführt.

Ist kein Überlauf vorhanden, wird der nächste Befehl übersprungen, d.h. Q1 wird in $\phi 2$ gesetzt (weiterer Ablauf s. 2.5.1.2.).

$$Q1' = \underbrace{G \ H}_{\phi 2(\phi 4)} T3 \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{Z\text{-Befehl}} (R \ C + \dots) + \dots$$

└kein Überlauf
entschlüsselt -

Bei der Ausführung des A- oder S-Befehles werden Teile der Addierlogik zur Erkennung des Überlaufes zur Vorzeichenzeit (T3) benutzt. Das C-Register wird entsprechend gesetzt.

$$Cw' = \underbrace{F \ G}_{\phi 4} \underbrace{Q1 Q2 Q3}_{A \text{ oder } S} T3 \underbrace{S}_{A\text{-Befehl}} (I1 I2 L + I1 I2 L) + S(I1 I2 L + I1 I2 L) + \dots$$

Vorzeichenzeit

Bei der Ausführung der Division kann, wenn der Dividend größer als der Divisor ist, ein Überlauf während der zweiten Wortzeit der $\phi 4a$ stattfinden.

Da die Division ein logischer Ablauf von A- und S-Befehlen ist, werden die Vorzeichen des Dividenden und des Teilquotienten zur Erkennung des Überlaufes benutzt.

Das Vorzeichen des Teilquotienten wird während der zweiten Wortzeit der ϕ_{4a} in A gespeichert und das des Dividenden in P6. Damit wird das C-Register beim Überlauf durch folgenden Ausdruck gesetzt:

$$Cw' = \underbrace{F \ G \ H \ P1 \ T3}_{\substack{\text{2te Wortzeit} \\ \phi_{4a}}} \underbrace{Q3}_{\substack{\text{Divi-} \\ \text{sicn}}} \underbrace{(P6 \ A + P6 \ A)}_{\text{Überlauf}}$$

Vorzeichen

Folgt dem Überlauf ein -Z-Befehl, wird das Überlauf-Bit zurückgesetzt.

Die Zurücksetzung erfolgt durch Unterbrechung des Umlaufes im C-Register in ϕ_4 .

$$Cw' = \underbrace{G \ S2 \ \underline{brc} \ C}_{\substack{\text{Umlauf bei} \\ \text{-Befehl in} \\ \phi_4 \text{ unter-} \\ \text{brochen}}} \underbrace{(R+F+T3+Q1+Q2+Q3+Q4+\dots)}_{\phi_2} + \underbrace{G \ T3 \ C \ \underline{brc}}_{\substack{\text{Umlauf bei je-} \\ \text{dem Befehl au-} \\ \text{Ber Z} \quad \quad \quad \text{Umlauf des} \\ \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{Überlauf-Bit} \\ \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \quad \text{in } \phi_1 \text{ u. } \phi_3}}$$

-Verhindert Übernahme des Vorzeichens aus dem R-Register beim U- oder T-Befehl

2.5.2. B-Befehl

$$B \hat{=} \underline{Q1} \underline{Q2} \underline{Q3} Q4 \hat{=} 0001$$

Mit dem B-Befehl wird der Inhalt einer Hauptspeicherzelle (V) in den Akkumulator (A) gebracht ($V \rightarrow A$).

Der Ablauf der Phasen 1...3 siehe Abschnitt 2.4.2.

Zu Beginn der ϕ_4 wird der Umlauf von A unterbrochen und das vom Hauptspeicher gelesene Wort V in den Akkumulator geschrieben.

$$Aw' = \underbrace{F \ G \ H}_{\phi_4} \underbrace{\underline{Q1} \ \underline{Q2} \ \underline{Q3} \ Q4}_{\text{B-Befehl}} \ V + \dots$$

Der neue Inhalt läuft in A um.

$$Aw' = (\underbrace{F}_{\phi_1, \phi_2} + \underbrace{G}_{\phi_3}) \underbrace{H \ T_0}_{\text{Umlauf}} A + \dots$$

-E/A-Schalter nicht gedrückt

Am Ende von ϕ_4 werden F und G zurückgesetzt.

$$\underline{F}' = F \ G \ H \ T_3 + \dots$$

$$\underline{G}' = G \ H \ T_3 + \dots$$

2.5.3. Y-Befehl

$$Y \hat{=} \underline{Q1} \underline{Q2} Q3 \underline{Q4} \hat{=} 0010$$

Durch den Y-Befehl wird bei einer Hauptspeicherzelle der Adreßteil mit der entsprechenden Bitkombination des Akkumulators (Adreßteil) überschrieben. Der weitere Inhalt der Hauptspeicherzelle wird nicht verändert.

Die Schreiberlaubnis W erfolgt zur Adreßzeit in ϕ_4 .

$$W = \underbrace{F G \underline{Q1} \underline{Q2} Q3}_{\phi_4 \text{ Y(R)-Befehl}} \underbrace{S2 + \dots}_{\text{Adreßzeit}}$$

Die Schreibinformation Vw' wird aus dem Akkumulator geholt.

$$Vw' = L (\underline{Q4} A + \dots)$$

|
└─ kein R-Befehl

L wird zu Beginn von ϕ_4 gesetzt.

$$L' = F G \underline{Q2} + \dots$$

|
└─ auch beim Y-Befehl

2.5.4. R-Befehl

$$R \hat{=} \underline{Q1} \underline{Q2} Q3 Q4 \hat{=} 0011$$

Mit dem R-Befehl wird zur Adresse im C-Register eine "1" addiert. Die neue Adresse wird in den Hauptspeicher gebracht.

Die Schreiberlaubnis W erfolgt wie beim Y-Befehl (2.5.3.).

$$W = F G \underline{Q1} \underline{Q2} Q3 S2 + \dots$$

Die Schreibinformation wird aus dem C-Register geholt.

$$Vw' = L \underline{Q1} Q4 (\underline{K C} + \underline{K C} + \dots)$$

R-Befehl Addition einer "1" zur Adresse

K wird am Ende einer jeden Wortzeit gesetzt. Es wird zurückgesetzt sobald eine "0" in der Adresse in C erscheint.

Hiermit ist eine Addition von 1 zur letzten Stelle erreicht.

$$\underline{K'} = \underline{G H} S2 \underline{C} \underline{Faf} (Q3 + \dots) + \dots$$

$\underbrace{\quad}_{\emptyset 2, \emptyset 4}$

 \downarrow
R-Befehl

2.5.5. D-Befehl

$D \hat{=} \underline{Q1} \ Q2 \ \underline{Q3} \ Q4 \ \hat{=} \ 0101$

Beim D-Befehl wird der Inhalt des Akkumulators (Dividend) durch den Inhalt einer Hauptspeicherzelle (Divisor) dividiert. Der gerundete Quotient steht im Akkumulator.

Der LGP 21 ist eine Festkommamaschine. Das Maschinenkomma folgt der Vorzeichenstelle. Damit ergibt sich, daß in der Maschinensprache Zahlen ≥ 1 nicht darstellbar sind. Der Zahlenbereich in der Maschinensprache reicht von -1 bis +0,999 ($\approx +1$).

Jeder Divisionsbefehl wird im Rechner als Festkommaoperation ausgeführt.

Für jede Division ist Voraussetzung, daß der absolute Inhalt des Divisors (Hauptspeicherzelle) größer als der absolute Inhalt des Dividenden (Akkumulator) ist, damit der Quotient < 1 wird. Durch entsprechende Wahl des gedachten Kommas von Divisor und Dividend kann dieses erreicht werden (Differenz von Maschinenkomma zu gedachtem Komma).

Bei der Division von Zahlen mit beliebigem Vorzeichen ergeben sich die im folgenden aufgeführten Möglichkeiten. Negative Zahlen werden als Komplement zur Konstanten $C = 2$ dargestellt (z.B. $2-a$). Die hierdurch notwendigen Korrekturen werden im Divisionsablauf berücksichtigt (s. Rint, IV. Band ab S. 665).

$a : b$	\longrightarrow	$a : b$
$(-a) : b$	\longrightarrow	$(2-a) : b$
$a : (-b)$	\longrightarrow	$a : (2-b)$
$(-a) : (-b)$	\longrightarrow	$(2-a) : (2-b)$

Bei der Division wechseln Subtraktion des Divisors und Schiften des laufenden Restes um 1 Bit miteinander ab. Der Divisor wird zunächst vom Dividenden, bei den folgenden Subtraktionen dann vom Rest abgezogen. Das Vorzeichen eines jeden Restwertes wird weggespeichert und

ist für den Wert einer entsprechenden Ergebnisstelle maßgebend. Die Subtraktion wird zur Addition, wenn Divisor und der vorhergehende Restwert ungleiches Vorzeichen haben. Die Ergebnisbildung geschieht am Ende der Division aus den gespeicherten Vorzeichen der Restwerte. Gegebenenfalls wird auch die Aufrundung vorgenommen.

Die Division dauert 64 Wortzeiten (32 x Subtraktion, 32 x Schiften) plus 3 weitere Wortzeiten. ϕ_4 des D-Befehls geht durch Setzen des H-Flip-Flops in ϕ_{4a} über. F und G dienen der Unterscheidung einzelner Wortzeitgruppen am Anfang und Ende der Division. Das abwechselnde Subtrahieren und Schiften wird mit dem P1-Flip-Flop gesteuert.

Aus dem Vorzeichen des Restes wird in der 67. Wortzeit das Vorzeichen des Quotienten ermittelt. Der Rest wird in der 3. Wortzeit aus den Bit-Positionen 1-32 in die Positionen 33-64 geschifft. Dazu wird A^* (2 Wort plus 1 Bit) für die Dauer von 32 Takten als Umlaufregister benutzt. Durch das Vorhandensein der 65. Bitstelle erscheint der bei Pos. 33-64 stehende Restwert in der folgenden Subtraktionswortzeit gegenüber dem Divisor (R) um 1 Bit verspätet. Das 65. Bit ist eine rechtsseitige Erweiterung des Restwertes und muß darum immer Null sein. Diese Forderung wird erfüllt, wenn zwei Wortzeiten vorher an der 64 Bit weiter links befindlichen Stelle (Vorzeichen) immer eine Null geschrieben steht. Dazu wird in sämtlichen ungeradzahligen Schiftwortzeiten aus dem Schreibsignal Aw' das Vorzeichen ausgeblendet.

Bevor nach einer Schiftwortzeit die nächste Subtraktionswortzeit beginnt, nimmt das P6-Flip-Flop das Vorzeichen des vorhergehenden Restes auf.

Dieses ist zur Vorzeichenzeit T3 der Schiftwortzeit im Akku A in der Position 32 angekommen. Es kann gelesen werden.

Von der 4. Wortzeit ab findet in allen geradzahligen

Subtraktionswortzeiten eine arithmetische Verknüpfung von Divisor und geschiftetem Restwert statt. Über Subtraktion oder Addition entscheidet das ins P6-Flip-Flop übernommene Vorzeichen des zuletzt ermittelten Restwertes. Der geschiftete Restwert wird aus A* gelesen. Während der Einschreibung eines neu berechneten Restwertes in den Akku, erreicht das Vorzeichen des vorherberechneten Restwertes zur T3-Zeit die Bitposition 65. Hier steht es jetzt als Ergebnisstelle rechts neben den schon vorhandenen Stellen.

Nach dem Einschreiben folgt mit Ausnahme der 66. Wortzeit wieder eine Schiftwortzeit. Nach der 66. Wortzeit erfolgt die Ergebnisbildung.

Die im folgenden beschriebenen Funktionen können zweckmäßigerweise parallel zum Text auch in der Divisionstabelle (s. S.A2-59) und im Blockschaltbild verfolgt werden.

Die Stellung des P1-Flip-Flops ist in $\emptyset 4$ beliebig. Es wird jedoch am Ende $\emptyset 4$ zurückgesetzt. In $\emptyset 4a$ dient P1 zur Unterscheidung der geradzahligen und ungeradzahligen Wortzeiten, ausgenommen der 67. Wortzeit.

In allen geradzahligen Wortzeiten ist $P1 = 0$. Es wird subtrahiert. In allen ungeradzahligen ist $P1 = 1$. Es wird geschiftet (ausgenommen 1. und 67. Wortzeit).

$$\underline{P1'} = G \underset{\emptyset 4}{T3} P1 \underset{\text{Faf}}{P1b} + H \underset{\emptyset 4a}{P1} T3$$

$$P1' = H \underset{\emptyset 4a}{\underline{P1}} T3$$

Damit der Divisor (V) zu jeder zweiten Wortzeit für die Subtraktion verfügbar ist, wird er in der normalen Phase 4 ins R-Register geschrieben.

$$Rw' = \underset{\emptyset 4}{G} \underset{D\text{-Befehl}}{H} \underbrace{Q1 \ Q2 \ Q4}_{\text{D-Befehl}} \cdot V$$

Das Vorzeichen des Divisors (V) wird für die gesamte Dauer der $\emptyset 4a$, wie bei der Multiplikation, im P5-Flip-Flop gespeichert.

$$P5' = \underbrace{G \ H}_{WZ1} \quad T3 \quad P5a \cdot \underbrace{V}_{\text{Multiplikand/Divisor}}$$

Faf

$$\underline{P5'} = \underbrace{G \ H}_{WZ1} \quad T3 \quad P5a \quad \underline{V}$$

Multiplikand/Divisor

In P6 wird am Ende von $\emptyset 4$ das Vorzeichen des Dividenten (A) gespeichert und in $\emptyset 4a$ vor Beginn einer jeden Subtraktionswortzeit das Vorzeichen des laufenden Restes.

Der Inhalt von P6 bleibt jeweils zwei Wortzeiten lang erhalten.

$$P6' = \underbrace{FGH}_{\emptyset 4} \cdot P6a \quad T3 \cdot \underbrace{A}_{\text{Vorzeich.}} + H \ P1 \ \underbrace{Q3}_{\emptyset 4a} \quad T3 \quad \underbrace{A}_{\text{Restwert}}$$

Dividend Vorzeichen

$$\underline{P6'} = \underbrace{FGH}_{\emptyset 4} \quad P6a \quad T3 \cdot \underline{A} + H \ P1 \ \underline{Q3} \quad T3 \quad \underline{A}$$

$\emptyset 4a$

Das Vorzeichen des Dividenten (A) wird in der $\emptyset 4$ (1. Wortzeit) ausgeblendet.

$$Aw' = \underbrace{H \ A}_{\emptyset 4} \ \underline{To} \ \left[\underbrace{T3}_{\text{Umlauf}} \ \underbrace{Q1 \ Q2}_{\text{D-Befehl}} \ (Q4 + \dots) + \dots \right]$$

Vorzeichen-Ausblendung

Während der weiteren Rechnung ($\emptyset 4a$) geht der Rest dem Betrag nach gegen Null, da bis zur 66. Wortzeit subtrahiert bzw. addiert wird. Die Entscheidung über Subtraktion oder Addition erfolgt durch S. Es wird immer subtrahiert, wenn P5 und P6 gleich sind.

$$S = \underbrace{H \ Q3}_{\emptyset 4a} \ \underbrace{(P5 \ P6 + P5 \ P6)}_{\text{Gleichheit}}$$

Division

In der zweiten Wortzeit wird vom Dividenden (A) entsprechend der Stellung von S der Divisor (V) subtrahiert oder addiert.

$$I1 = \underbrace{\underline{F} \underline{G} \underline{P1} \underline{Q3}}_{\substack{\text{D-Befehl} \\ \text{2. Wortzeit } \emptyset 4a}} (\underbrace{A + \underline{P6} \underline{T3}}_{\substack{\text{Dividend} \\ \text{Vorzeichen des Dividenden (A)}}})$$

$$I2 = \underbrace{H \underline{Q3} \underline{P1}}_{\substack{\emptyset 4a \\ \text{Division}}} R + \dots$$

└ Divisor
└ geradzahlige Wortzeit

Ein entstehender Übertrag wird im L-Flip-Flop gespeichert.

$$L' = (\underline{T3} + \dots) (\underbrace{S \underline{I1} \underline{I2} \underline{L}}_{\text{Addition}} + \underbrace{S \underline{I1} \underline{I2} \underline{L}}_{\text{Subtraktion}}) + \dots$$

$$\underline{L}' = \underline{T3} \underline{Q3} + \underline{Q2} \underline{Q1} (\underbrace{S \underline{I1} \underline{I2} \underline{L}}_{\text{Subtraktion}} + \underbrace{S \underline{I1} \underline{I2} \underline{L}}_{\text{Addition}}) + \dots$$

Anfang jeder D-Bef. Subtraktion Addition
Wortzeit beim
D(A,S)-Befehl

Das Ergebnis wird nach A und A* geschrieben.

$$Aw' = A*w' = \underbrace{H \underline{P1}}_{\substack{\text{2. Wortzeit} \\ \emptyset 4a}} (\underbrace{L \underline{I1} \underline{I2} + L \underline{I1} \underline{I2} + L \underline{I1} \underline{I2} + L \underline{I1} \underline{I2}}_{\text{Addierlogik}})$$

In allen ungeradzahligen Wortzeiten, von der 3. bis zur 65. Wortzeit, wird der Rest der in der vorderen Hälfte von A* (auch in A) steht, in die zweite Hälfte von A* geschiftet (um 32 Bit).

$$I1 = \underbrace{\underline{G} \underline{H} \underline{P1} \underline{Q3}}_{\substack{\text{Division} \\ \text{3., 5... 63. Wortzeit}}} A^* + \underbrace{\underline{F} \underline{H} \underline{Q3}}_{\text{65., 66. Wortzeit}} A^*$$

$$I2 = \underline{H} \underline{P1} \underline{Q3} R = "0"$$

nicht ungeradzahlig

$$Aw' = H \underbrace{T3}_{\substack{\text{Ausblendung des} \\ \text{Vorzeichens}}} \left(\cancel{L I1 I2} + \underbrace{L I1 I2}_{\text{Umlauf von } A^*} + \cancel{L I1 I2} + \cancel{L I1 I2} \right)$$

Während aller weiteren geradzahligen Wortperioden wird von dem in der zweiten Hälfte von A* stehenden Rest der Divisor ($V \hat{=} R$) subtrahiert bzw. addiert. Das Ergebnis wird nach A und A* geschrieben.

Dadurch, daß A* 65 Bit lang ist, erscheint der Rest in der Subtraktionswortzeit gegenüber dem Divisor (R) um 1 Bit verspätet.

$$I1 = \underbrace{F G H A^*}_{4.6\dots64.\text{Wortzeit}} + \dots$$

$$I2 = H \underbrace{P1 Q3}_{\text{geradzahlige Wortzeit}} R + \dots$$

S, L, A-Logik wie in der zweiten Wortzeit.

Mit dem Ende der 66. Wortzeit steht der tatsächliche oder invertierte ganze Quotient im Akkumulator (Aw'-Logik wie bei der 3., 5....63. Wortzeit).

Das Vorzeichen des Divisors ($V \hat{=} R$) gibt an, ob der Quotient noch invertiert werden muß. Ist der Divisor positiv ($P5 = 1$), wird invertiert.

Der Quotient wird gerundet, wenn das Vorzeichen von Rest (P6) und Divisor (P5) übereinstimmen. Die Rundung erfolgt durch Subtraktion einer 1 von jeder Bitstelle; dieses entspricht der Addition einer 1 zur wenigstbedeutenden Stelle.

Invertierung und Rundung erfolgen in der 67. Wortzeit.

$$I1 = \underbrace{\underbrace{F G H A^*}_{67.\text{Wortzeit}} \underbrace{P5}_{\text{Divisor +}}}_{\text{Divisor +}} + \underbrace{\underbrace{G H Q3}_{65..67.\text{Wortzeit}} \underbrace{P5 A^*}_{\text{Divisor -}}}_{\text{Divisor -}} + \dots$$

$$I2 = \underbrace{F \ G \ H}_{67. \text{Wortzeit}} \underbrace{(P5 \ P6 + \underline{P5} \ \underline{P6})}_{\text{Subtraktion einer 1}} + \dots$$

$$S = \underbrace{F \ G \ H}_{67. \text{Wortzeit}} + \dots$$

$$Aw' = \underbrace{F \ G \ H}_{67. \text{Wortzeit}} \underbrace{(L \ \underline{I1} \ \underline{I2} + L \ \underline{I1} \ \underline{I2} + L \ \underline{I1} \ \underline{I2} + L \ \underline{I1} \ \underline{I2})}_{\text{Addierlogik}}$$

Fall (+a) : (+b)

Sind Divisor und Dividend positiv ($P5 = 0$) und teilbar ohne bleibenden Rest ($P6 = 1$), so wird

$$I1 = \begin{matrix} (P5 & A^* & + & \underline{P5} & \underline{A^*}) & \hat{=} & \underline{A^*} \\ \underset{0}{\downarrow} & & & \underset{1}{\downarrow} & & & \end{matrix}$$

$$I2 = \begin{matrix} (P5 & P6 & + & \underline{P5} & \underline{P6}) & \hat{=} & 0 \\ \underset{0}{\downarrow} & \underset{1}{\downarrow} & & \underset{1}{\downarrow} & \underset{0}{\downarrow} & & \end{matrix}$$

$$S = \underline{F \ G \ H} + \hat{=} 1$$

und damit $Aw' \hat{=} \underline{A^*}$

Ist eine Aufrundung aufgrund eines bleibenden Restes erforderlich ($P6 = 0$), so wird

$$I2 = \begin{matrix} (P5 & P6 & + & \underline{P5} & \underline{P6}) & \hat{=} & 1 \\ \underset{0}{\downarrow} & \underset{0}{\downarrow} & & \underset{1}{\downarrow} & \underset{1}{\downarrow} & & \end{matrix}$$

Subtraktion von 1,111...1 ist gleichbedeutend mit Addition des entsprechenden Komplements (0,000...1).

Fall (-a) : b

Das negative Vorzeichen des Dividenden geht am Ende der normalen Phase 4 in das P6-Flip Flop. In der 2. Wortzeit wird dadurch das Signal $S = 0$, so daß die Division mit einer Addition anstelle einer Subtraktion beginnt. Wegen der vorausgesetzten Größenverhältnisse muß der Rest positiv werden.

Aus seiner Null in der Vorzeichenstelle wird in der 67. Wortzeit das Vorzeichen des Quotienten "1" ($Aw' \hat{=} A^*$). Außer in der 2. Wortzeit hat das negative Vorzeichen des Dividenden keinen direkten Einfluß auf den Divisionsablauf. Nur indirekt pflanzt sich seine Wirkung über folgende Kette fort:

Signal S (Wortzeit 2) - neues Restwertvorzeichen - P6-Flip-Flop - Signal S (Wortzeit 4) - neues Restwertvorzeichen usw.

Die Ergebnisbildung in der 67. Wortzeit geschieht wie bei positivem Dividenden, der Quotient ist negativ und steht als Komplement im Akkumulator.

Fall a : (-b)

Das negative Vorzeichen des Divisors wird am Ende der normalen Phase 4 vom P5-Flip Flop übernommen und darin bis zum Divisionsende gehalten. Dadurch wird in der 2. Wortzeit das Signal S = 0. Die Division beginnt also mit einer Addition ($a+(-b)$), wobei das Vorzeichen des Restwertes wegen der vorausgesetzten Größenverhältnisse negativ werden muß. In den weiteren Wortzeiten wird jetzt subtrahiert, wo unter den gleichen Bedingungen ($P6=1$) bei positivem Divisor addiert werden wäre.

Sind Dividend und Divisor ohne bleibenden Rest teilbar, so wird in der 67. Wortzeit

$$I1 = \left(\underset{1}{P5} A^* + \underset{0}{P5} A^* \right) \hat{=} A^* \quad \text{und mit } P6=1 \text{ auch}$$

$$I2 = \left(\underset{1}{P5} \underset{1}{P6} + \underset{0}{P5} \underset{0}{P6} \right) \hat{=} 1$$

und damit $Aw' = A^* - 1,111\dots1$

Das in diesem Beispiel erforderliche negative Ergebnis erhält man aus dem Vorzeichen des Restwertes der 2. Wortzeit, der Bedingung $I1 = A^*$ in der 67. Wortzeit und der Addition einer "1" zur letzten Stelle. Durch die Aufrundung des Ergebnisses erhöht sich die letzte Stelle des vorhandenen Komplements um 1, jedoch wird dadurch der

Betrag der negativen Zahl in der letzten Stelle um 1 vermindert.

Ist bei der angenommenen Division eine Aufrundung aufgrund eines bleibenden Restes erforderlich, so wird die Subtraktion einer 1 von der letzten Stelle des Betrages der negativen Zahl unterlassen. Hierfür wird mit

$$P_6 = 0 \quad \text{auch} \quad I_2 = \begin{pmatrix} P_5 & P_6 & + & \underline{P_5} & \underline{P_6} \\ \dot{1} & \dot{0} & & \dot{0} & \dot{1} \end{pmatrix} \hat{=} 0$$

Fall (-a) : (-b)

Die negativen Vorzeichen werden wieder in der 1. Wortzeit in die Flip-Flops P6 und P5 gespeichert. In der 2. Wortzeit wird das Signal S = 1. Die Division beginnt wieder normal mit einer Subtraktion (-a)-(-b). Wegen der vorausgesetzten Größenverhältnisse muß der Restwert positiv werden. Aus seiner Null in der Vorzeichenstelle wird in der 67. Wortzeit das Vorzeichen des Quotienten zu Null ermittelt ($Aw' \hat{=} A^*$).

Das negative Vorzeichen des Dividenden übt einen direkten Einfluß auf den Divisionsablauf nur in der 2. Wortzeit aus. In den weiteren Wortzeiten wird wegen des negativen Divisors ($P_5=1$) jetzt subtrahiert, wo unter den gleichen Bedingungen ($P_6=1$) bei positivem Divisor addiert worden wäre. Die Ergebnisbildung läuft wie im "Fall a : (-b)" beschrieben ab.

Wortzeitgruppen in Phase $\emptyset 4$ und $\emptyset 4a$

F und G werden in der $\emptyset 4a$ zur Unterscheidung der einzelnen Wortzeitgruppen benutzt.

Am Ende von $\emptyset 4$ werden F und G für zwei Wortzeiten zurückgesetzt.

$$\begin{array}{l} \underline{F'} = F \ G \ \underline{H} \ T3 \\ \underline{G'} = G \ \underline{H} \ T3 \end{array} \quad \longrightarrow \quad \underline{F} \ \underline{G} \ H = \begin{array}{l} 2. \text{ und } 3. \text{ Wort-} \\ \text{zeit} \end{array}$$

Am Ende der 3. Wortzeit wird F wieder gesetzt.

$$\underline{F'} = \underline{F} \ \underline{G} \ H \ P1 \ T3 \quad \longrightarrow \quad F \ \underline{G} \ H = \begin{array}{l} 4. \text{ bis } 64. \text{ Wort-} \\ \text{zeit} \end{array}$$

Am Ende der 64. Wortzeit wird G gesetzt.

$$\underline{G'} = \underline{G} \ H \ Q4 \ Ga \ \underline{be} \ T3 \ K \longrightarrow F \ G \ H = \begin{array}{l} 65., 66. \text{ Wortzeit} \\ | \\ \text{Sektorkoinzidenz nach } 64 \text{ Wortzeiten} \end{array}$$

Das K-Flip-Flop dient zum Sektorvergleich. K wird zu Beginn einer jeden Wortzeit gesetzt. Bei Nichtübereinstimmung der Sektoradreßspur S1 und der Bits 4-10 aus dem C-Register wird K zurückgesetzt. Bei Übereinstimmung bleibt K über die ganze Wortzeit gesetzt.

$$K' = T3 \ \underline{Faf} \ (F + G + R + Q3 + Q4 + \dots)$$

$$\underline{K'} = H \ \underline{S2} \ (\underline{C} \ \underline{S1} + \underline{C} \ \underline{S1} \ \underline{S3})$$

$\emptyset 4a$ | Antikoinzidenz | Bit 1...11, 16, 17, 30, 31

Aus der S1-Spur wird zu jeder Wortzeit der $\emptyset 3$ während S2 S3 eine (neue) Sektoradresse in das C-Register geschrieben:

$$Cw' = \underline{\underline{G \ H \ S2 \ S3}} \ S1 + \dots$$

$\emptyset 2$

Ende $\emptyset 3$ steht die Operandenadresse des D (N, M)-Befehles im C-Register und läuft in $\emptyset 4$ um.

$$Cw' = G \underset{\substack{| \\ \emptyset 3, \emptyset 4}}{S2} C \underset{\downarrow}{brc} (Q1+Q2+Q3+Q4+\dots) + H C \underset{\substack{\downarrow \\ \text{Nicht Füllen/} \\ \text{Löschen}}}{brc} + \dots$$

Umlauf Umlauf $\emptyset 4a$

Nach der Sektorkoinzidenz kann G am Ende der 64. Wortzeit gesetzt werden.

Am Ende der 66. Wortzeit wird F zurückgesetzt:

$$\underline{F'} = F G \underset{\substack{| \\ \text{gerade Wortzeit}}}{P1} T3 \quad \underline{F} G H \hat{=} 67. \text{ Wortzeit}$$

Wortzeit LGP 21	Wortzeit Beispiel	F	G	H	P1	P5	P6	II	I2	S	Aw	Akkuinhalt am Wortzeitende		Bemerkungen	
								(A)	(V)	(1)		A	A*		
1	1 1 0 X	X				X		(A+T3P6)	(V)	(1)	A.T3	0	0	1 1 1 1 0 0 0	Dividend durch B-Bef. in Akku
2	0 0 1 0	0				0		(A+T3P6)	R	$(P5P6+P5P6) = 1$	$\Sigma = (A+T3P6) - R$	1	1	1 0 1 1 0 0 0	
3	0 0 1 1	0				0		A*	0	$= 1$	$\Sigma .T3 = A*.T3$	0	1	1 1 1 1 0 0 0	
4	1 0 1 0	0				1		A*	R	$= 0$	$\Sigma = A*pl.R$	0	0	1 0 1 0 0 0 1	
5	1 0 1 1	0				1		A*	0	$= 0$	$\Sigma .T3 = A*.T3$	0	1	1 1 0 0 0 1 0	
6	1 0 1 0	0				0		A*	R	$= 1$	$\Sigma = A* - R$	0	0	0 0 0 0 0 0 0	
7	1 0 1 1	0				0		A*	0	$= 1$	$\Sigma .T3 = A*.T3$	0	1	1 0 0 0 1 0 0	0: Null geworden durch I3
8	1 0 1 0	0				0		A*	R	$= 1$	$\Sigma = A* - R$	1	0	1 1 0 0 0 0 0	
9	1 0 1 1	0				0		A*	0	$= 1$	$\Sigma .T3 = A*.T3$	0	1	0 0 0 1 0 0 0	
10	1 0 1 0	0				1		A*	R	$= 0$	$\Sigma = A*pl.R$	1	0	1 1 0 0 0 0 0	restl. Stellen des Dividenden
11	1 0 1 1	0				1		A*	0	$= 0$	$\Sigma .T3 = A*.T3$	0	0	0 0 0 1 0 0 1	
12	1 0 1 0	0				1		A*	R	$= 0$	$\Sigma = A*pl.R$	1	0	1 1 0 0 0 0 0	
13	1 0 1 1	0				1		A*	0	$= 0$	$\Sigma .T3 = A*.T3$	0	0	0 0 1 0 0 1 1	Ergebnisstellen
14, 16...62	1 0 1 0	0				1		A*	R	$= 0$	$\Sigma = A*pl.R$	1	0	1 1 0 0 0 0 0	
15, 17...63	1 0 1 1	0				1		A*	0	$= 0$	$\Sigma .T3 = A*.T3$	0	0	0 0 1 0 0 1 1	
64	1 0 1 0	0				1		A*	R	$= 0$	$\Sigma = A*pl.R$	1	0	1 1 0 0 0 0 0	
65	1 1 1 1	0				1		A*	0	$= 0$	$\Sigma .T3 = A*.T3$	0	0	1 0 0 1 1 1 1	
66	1 1 1 0	0				1		A*	R	$(P5P6+P5P6) = 0$	$\Sigma = A*pl.R$	1	0	1 1 0 0 0 0 0	
67	0 1 1 1	0				1		A*	0	$= 1$	$\Sigma = A* - 0$	0	1	1 0 0 0 0 0 0	

$\frac{b}{a} = \frac{0,111100}{15} = 0,111100000$
 Dezimal: $0,111100000$
 Divisor: $5 @ 3 = @ 2 : 3$

Divisionstabelle
LGP 21

2.5.6. N-Befehl

$$N \hat{=} \underline{Q1} \ Q2 \ Q3 \ \underline{Q4} \ \hat{=} \ 0110$$

Beim N-Befehl wird der Inhalt des Akkumulators (Multiplikator) mit dem Inhalt einer Hauptspeicherzelle (Multiplikand) multipliziert.

Die wenigstbedeutende Hälfte des Produktes steht im Akkumulator.

Jeder Multiplikationsbefehl wird im LGP 21 intern als Festkommaoperation ausgeführt. Das Maschinenkomma steht zwischen Vorzeichen und meistbedeutendem Bit.

Die Multiplikation besteht aus wiederholten Additionen des Multiplikanden zum laufenden Teilprodukt und Schiften des Teilproduktes um 1 Bit nach jeder Addition.

Weil jeder Faktor 32 Bit hat, wird 32 mal addiert und 32 mal geschiftet. Damit umfaßt das Produkt einschließlich Vorzeichen und Spacerbit 64 Bit. Das gesamte Ergebnis steht in A*. Für eine Weiterverarbeitung kann bei der N-Multiplikation aber nur die wenigstbedeutende Hälfte des A*-Inhaltes verwendet werden. In der Vorzeichenstelle steht der meistbedeutende Wert (kein Vorzeichen!).

Bei der Multiplikation von Zahlen mit beliebigem Vorzeichen, ergeben sich die im folgenden aufgeführten Möglichkeiten. Negative Zahlen werden als Komplement zur Konstanten 2 dargestellt (z.B. 2-a) (siehe auch Division):

	Ist-Wert	Soll-Wert
$a \times b \hat{=} a \times b$	$\hat{=} a . b$	$+ab \hat{=} ab$
$-a \times b \hat{=} (2-a) \times b$	$\hat{=} 2b - ab$	$-ab \hat{=} (2-ab)$
$-a \times (-b) \hat{=} (2-a) \times (2-b)$	$\hat{=} 4-2b-2a+ab$	$+ab \hat{=} ab$
$a \times (-b) \hat{=} a \times (2-b)$	$\hat{=} 2a-ab$	$-ab \hat{=} (2-ab)$

Aus dem Ist-Wert läßt sich der Sollwert durch Addition einer Korrektur berechnen:

$$\text{Ist} + \text{Korr.} = \text{Soll} \longrightarrow \text{Korr.} = \text{Soll} - \text{Ist}$$

	Soll	Ist	Korrektur, kompl.	Korrektur, normal
a x b	ab	ab	0	0
-a x b	(2-ab)	(2b-ab)	(2-2b)	-2b
-a x (-b)	ab	(4-2b-2a+ab)	-(2-2b+2-2a)	+2a+2b
a + (-b)	(2-ab)	(2a-ab)	(2-2a)	-2a

Für den Sonderfall $(-a) \times (-b) = (-1) \cdot (-a)$ erhält man das falsche Ergebnis: (-1) , da Zahlen ≥ 1 nicht darstellbar sind. *Es wird aber nichts gemildert!*

Die im folgenden beschriebenen Funktionen können zweckmäßigerweise auch parallel zum Text in der Multiplikationstabelle (Seite und) und im Blockschaltbild des Rechenwerkes verfolgt werden.

Während der ersten Wortzeit $\emptyset 4$ wird der Multiplikand V in das R-Register geschrieben, um ihn während der Ausführung des Befehls bei jeder Addition sofort zur Verfügung zu haben.

$$Rw' = \underbrace{G \ H \ V}_{\emptyset 4 \text{ Multiplikand}} \underbrace{Q1 \ Q2 \ Q3}_{N, M\text{-Multiplikation}} + \underbrace{R \ \text{brc} \ H}_{\text{Umlauf in } \emptyset 4a}$$

Gleichzeitig wird das Vorzeichen des Multiplikanden (V) im P5-Flip-Flop gespeichert.

$$P5' = \underbrace{G \ H \ T3}_{\emptyset 4} \underbrace{V}_{\text{Multiplikand}} \underbrace{P5a}_{E/A\text{-Verriegelung}} + \dots$$

$$\underline{P5'} = \underbrace{G \ H \ T3}_{\emptyset 4} \underbrace{V}_{\text{Multiplikand}} + \dots$$

Der Multiplikator wurde durch den vorhergehenden Befehl in den Akkumulator gebracht.

$$Aw' = \underline{H} \underline{To} A (\underline{F} + \underline{G} + \dots)$$

$\begin{array}{c} \downarrow \\ \phi_1, \phi_2 \quad \phi_3 \\ \text{---Umlauf A} \end{array}$

In der ϕ_4 wird das Vorzeichen des Multiplikators (A) in P6 gespeichert und aus dem Akkumulatorumlauf ausgeblendet.

$$Aw' = \underline{H} \underline{To} A \underline{T3} \underbrace{Q1 \quad Q2 \quad Q3}_{\substack{\text{N-Multiplikation} \\ \text{Ausblendung des Vorzeichens}}} + \dots$$

$$P6' = \underbrace{F \quad G \quad H}_{\phi_4} \underline{T3} \underline{A} \underline{P6a} + \dots$$

$\begin{array}{c} \downarrow \\ \text{---E/A-Verriegelung} \\ \text{---A(Multiplikator)} \end{array}$

$$\underline{P6'} = \underbrace{F \quad G \quad H}_{\phi_4} \underline{T3} \underline{A} \underline{P6a}$$

$\begin{array}{c} \downarrow \\ \text{---A(Multiplikator)} \end{array}$

Ab der zweiten Wortzeit dient P1 zur Unterscheidung der geradzahligen und ungeradzahligen Wortzeiten. In den geradzahligen Wortzeiten (Additionszeit) ist P1 zurückgesetzt, in den ungeradzahligen Wortzeiten (Schiftzeit) ist P1 gesetzt.

$$\underline{P1'} = \underline{G} \underline{T3} \underline{P1} \underline{P1b} + \underline{H} \underline{T3} \underline{P1}$$

$\begin{array}{c} \downarrow \qquad \qquad \downarrow \qquad \qquad \downarrow \\ \phi_4 \qquad \qquad \underline{Faf} \qquad \qquad \phi_{4a} \end{array}$

$$P1' = \underline{H} \underline{T3} \underline{P1}$$

$\begin{array}{c} \downarrow \\ \phi_{4a} \end{array}$

Um das Vorzeichen des Multiplikators (in P6) zu berücksichtigen, erfolgt in der zweiten Wortzeit eine Subtraktion des Multiplikanden (R) von Null in Abhängigkeit von P6 (Berechnung des Korrekturfaktors (-b)).

$$I1 = 0$$

$$I2 = \underline{G} \underline{H} \underline{P1} \underline{P6} \underline{R}$$

└─ Multiplikand
└─ Multiplikator

$$S = \underline{F} \underline{H} \underline{Q3}$$

2 WZ. └─ M, N-Befehl
Ø4a

$$L' = (\underline{T3} + \underline{H} \underline{P1} \underline{Q3}) (\underline{S} \underline{I1} \underline{I2} \underline{L} + \dots)$$

WZ 2 - MN-Befehl |
Übertragslogik

$$\underline{L}' = \underline{T3} \underline{H} + \underline{Q2} (\underline{Q3} + \underline{Q1}) (\underline{S} \underline{I1} \underline{I2} \underline{L} + \dots)$$

└─ auch MN-Befehl Übertragslogik
└─ Rücksetzer am Ende aller Ø

$$Aw' = \underline{H} \underline{P1} (\underline{L} \underline{I1} \underline{I2} + \cancel{\underline{L} \underline{I1} \underline{I2}} + \underline{L} \underline{I1} \underline{I2} + \cancel{\underline{L} \underline{I1} \underline{I2}})$$

Komplementbildung

In der 3. Wortzeit (siehe auch Multiplikationstabelle) wird die Berechnung des Korrekturfaktors (-2b) abgeschlossen. Die Multiplikation von (-b) mit 2 erhält durch Schiften des Ergebnisses um 1 Bit. Damit wird auch der Multiplikand um 1 Bit geschiftet, und es verschwindet die zuletzt ausmultiplizierte Stelle. Die neu zu multiplizierende Stelle wird ausgeblendet und in P6 (alle ungeradzahlgigen Wortzeiten) gespeichert.

$$P6' = \underline{H} \underline{T3} \underline{P1} \underline{Q3} \underline{A^*} + \dots$$

ungeradzahlige WZ

$$\underline{P6}' = \underline{H} \underline{T3} \underline{P1} \underline{Q3} \underline{A^*}$$

Um das Vorzeichen des Multiplikanden zu berücksichtigen, wird bei negativem Multiplikanden (P5 = "1") eine "1" von jedem Bit des Multiplikators subtrahiert. Bei positivem Multiplikanden (P5 = "0") wird von jedem Bit des Multiplikators eine "0" subtrahiert (Korrekturfaktor).

$$I1 = \underbrace{H P1 Q3 A^*}_{\substack{\text{M, N-Befehl} \\ \text{ungerade Wortzeiten } \emptyset 4a}}$$

$$I2 = \underbrace{F H P1 Q3 P5}_{\substack{\text{3.WZ} \\ \text{Vorzeichen Multiplikand (R)} \\ \text{M,N-Befehl}}}$$

$$L' = \underbrace{T3 S I1 I2 L}_{\text{Übertragslogik}} + \dots$$

$$\underline{L}' = T3 \quad \begin{array}{c} P1 \\ | \\ \text{ungerade WZ} \end{array} + \underbrace{Q2 Q3}_{\text{M,N-Befehl}} \underbrace{S I1 I2 L}_{\text{Übertragslogik}} + \dots$$

$$Aw' = Aw^* = H \underbrace{T3}_{\substack{| \\ \text{kein Vorzeichen}}} \underbrace{(L I1 I2 + L I1 I2 + L I1 I2 + L I1 I2)}_{\text{Addierlogik}}$$

In der 4. bis 64. Wortzeit (61 Wortzeiten) der Multiplikation wird in den geraden Wortzeiten in Abhängigkeit von P6 der Multiplikand oder Null zum geschifteten Teilprodukt addiert.

Eine weitere Addition erfolgt in den ungeraden Wortzeiten, wenn der Multiplikand negativ ist (P5 = "1") und das Multiplikatorbit eine 1 ist (P6 = "1").

In diesem Falle wird zu jeder Stelle des Teilproduktes eine 1 addiert, sonst wird der Inhalt von A* um 1 Bit geschiftet.

$$S = 0 \quad \text{ab 4. Wortzeit}$$

$$I1 = \underbrace{F G H}_{\text{4. bis 64. WZ}} A^*$$

$$I2 = \underbrace{G H P1}_{\substack{| \\ \text{ungerade WZ}}} P6 R + \underbrace{H P1 Q3}_{\substack{| \\ \text{gerade WZ}}} \underbrace{P5 P6}_{\text{M,N-Befehl}}$$

$$L' = (T3 + H Q3 P1) \underline{S} I1 I2 \underline{L} + \dots$$

$$\underline{L}' = T_3 P_1 + Q_2 Q_3 \quad \underline{S} \quad \underline{I_1} \quad \underline{I_2} \quad L + \dots$$

$$A_w' = A_w^{*'} = (H \underline{T_3} + \underline{HP_1} + \dots)(\underline{LI_1I_2} + \underline{LI_1I_2} + \underline{LI_1I_2} + \underline{LI_1I_2})$$

Am Ende der 64. Wortzeit stehen die wenigstbedeutenden Bits in A und das volle Produkt in A*.

Die N-Multiplikation ist beendet.

Wortzeitgruppen in Ø4 und Ø4a

F und G werden in der Ø4a zur Unterscheidung der einzelnen Wortzeitgruppen benutzt.

Am Ende von Ø4 werden F und G für zwei Wortzeiten zurückgesetzt.

$$\underline{F}' = \underline{F} \quad \underline{G} \quad \underline{H} \quad T_3 \qquad \underline{F} \quad \underline{G} \quad \underline{H} = 2. \text{ und } 3. \text{ Wortzeit}$$

$$\underline{G}' = \underline{G} \quad \underline{H} \quad T_3$$

Am Ende der 3. Wortzeit wird F wieder gesetzt.

$$\underline{F}' = \underline{F} \quad \underline{G} \quad \underline{H} \quad P_1 \quad T_3 \qquad \underline{F} \quad \underline{G} \quad \underline{H} = 4. \text{ bis } 64. \text{ Wortzeit}$$

Am Ende der 64. Wortzeit werden F und H zurückgesetzt

$$\underline{F}' = \underline{F} \quad \underline{H} \quad T_3 \quad \underline{Q_4} \quad \underline{K}$$

↓
Sektorkoinzidenz
nach 64 Wortzeiten

$$\underline{H}' = \underline{H} \quad T_3 \quad \underline{Q_4} \quad \underline{K}$$

↓
N-Befehl

Das K-Flip Flop dient zum Sektorvergleich. K wird zu Beginn einer jeden Wortzeit gesetzt. Bei Nichtübereinstimmung der Sektoradresse S1 und der Bits 4 bis 10 aus dem C-Register wird K zurückgesetzt. Bei Übereinstimmung bleibt K über die ganze Wortzeit gesetzt.

$$K' = T_3 \quad \underline{Faf} \quad (\underline{F} + G + R + Q_3 + Q_4 + \dots)$$

$$K' = H \underline{S2} (\underbrace{C \underline{S1} + C \underline{S1} \underline{S3}}_{\substack{\text{Anti-} \\ \text{koinzidenz}}} \quad \text{Bit } 1\dots 1, 16, 17, 30, 31)$$

ϕ_{4a}

Aus der S1-Spur wird zu jeder Wortzeit der ϕ_3 während $\underline{S2} \underline{S3}$ eine (neue) Sektoradresse in das C-Register geschrieben:

$$Cw' = \underbrace{G \underline{H} \underline{S2} \underline{S3}}_{\phi_3} S1 + \dots$$

Ende ϕ_3 steht die Operandenadresse des M,N(D)-Befehls im C-Register und läuft in ϕ_4 um.

$$Cw' = G \underline{S2} C \underline{brc} (Q1+Q2+Q3+Q4+\dots) + H C \underline{brc}$$

ϕ_3, ϕ_4 Umlauf

 nicht "füllen/
Löschen"
Umlauf ϕ_{4a}

Nach der Sektorkoinzidenz werden F und H zurückgesetzt, und der Rechner steht wieder in ϕ_1 .

2.5.7. M-Befehl

$$M \hat{=} \underline{Q1} \ Q2 \ Q3 \ Q4 \hat{=} \ 0111$$

Beim M-Befehl wird der Inhalt des Akkumulators (Multiplikator) mit dem Inhalt einer Hauptspeicherzelle (Multiplikand) multipliziert.

Die meistbedeutende Hälfte des Produktes steht im Akkumulator.

Die M-Multiplikation entspricht bis einschließlich der 64. Wortzeit der N-Multiplikation.

Nach der 64. Wortzeit stehen die wenigstbedeutenden Bit in A und die meistbedeutenden in der 2. Hälfte von A*.

In der 65. Wortzeit werden die meistbedeutenden Bits von A* nach A übertragen. Am Ende dieser Wortzeit stehen die Bits jedoch infolge des Schiftens in A* um 1 Bit, auch in A um 1 Bit verschoben.

In der 66. Wortzeit wird das Produkt durch Addition von A + A (entspricht Multiplikation von A mit 2) um 1 Bit nach links geschiftet. Danach stehen die meistbedeutenden Bits des Produktes in A.

Die P- und Addierlogik in der 65. Wortzeit entspricht der in den ungeraden Wortzeiten der N-Multiplikation.

In der 66. Wortzeit erfolgt die Addition A + A:

$$I1 = F \ G \ Q3 \ \underline{P1} \ A + \dots$$

$$I2 = G \ H \ Q3 \ \underline{P1} \ A + \dots$$

$$Aw' = H \ \underline{P1} \ (L \ I1 \ I2 + L \ \underline{I1} \ \underline{I2} + L \ I1 \ \underline{I2} + L \ \underline{I1} \ I2)$$

Wortzeitgruppen

Am Ende der 64. Wortzeit wird nach dem Sektorvergleich G gesetzt (s. Wortzeitgruppen N-Befehl S. A2-66).

$G' = \underline{G} \ H \ Q4 \ G \ a \ \underline{be} \ T3 \ K$
 | |
 M(D)-Befehl Sektorkoinzidenz nach 64 WZ

Die 65. Wortzeit ist gegeben mit
 G F H P1.

Die 66. Wortzeit ist gegeben mit
 F G H P1

Am Ende der 66. Wortzeit werden F,G,H zurückgesetzt,
 und der Rechner steht wieder in $\emptyset 1$.

$$\underline{F}' = F \ G \ T3 \ \underline{P1} + \dots$$

$$\underline{G}' = G \ H \ T3 \ Q3 \ \underline{P1}$$

$$\underline{H}' = G \ H \ T3 \ Q3 \ \underline{P1}$$

Wortzeit LGP 21	Wortzeit Beispiel	F G H P1	P5	P6	II	I2	S	Aw	Akkuinhalt am Wortzeitende		Bemerkungen
									A	A*	
1	1	1 1 0 X	X	X	(A)	(V)	X	A.T3	0 1 0 1 0 0 0 0	0 1 0 1 0 0 0 0	Multiplikator durch B-Befehl in Akku
2	2	0 0 1 0	0	0	0	P6R	1	$\sum = 0$	0 0 0 0 0 0 0 0	0 1 0 1 0 0 0 0	
3	3	0 0 1 1	0	0	A*	P5	1	$\sum.T3 = A*.I3$	0 0 1 0 0 0 0 0	0 0 0 0 0 0 0 0	
4	4	1 0 1 0	0	1	A*	P6R	0	$\sum = A*.I3$	0 1 1 0 0 0 0 0	0 0 0 0 0 0 0 0	O: Null geworden durch I3,
5	5	1 0 1 1	0	1	A*	P6P5	0	$\sum.T3 = A.I3$	0 1 0 0 0 0 0 0	0 1 1 0 0 0 0 0	P6 geht an für WZ 4 und 5
6	6	1 0 1 0	0	0	A*	P6R	0	$\sum = A*$	1 0 0 0 0 0 0 0	0 1 0 0 0 0 0 0	P6 geht aus für WZ 6 und 7
7	7	1 0 1 1	0	0	A*	P6P5	0	$\sum.T3 = A.I3$	0 0 0 0 0 0 0 0	1 1 0 0 0 0 0 0	
8	8	1 0 1 0	0	1	A*	P6R	0	$\sum = A*.I3$	1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 1	
9	9	1 0 1 1	0	1	A*	P6P5	0	$\sum.T3 = A*.I3$	0 0 0 0 0 0 0 1	1 1 1 0 0 0 0 0	
10	10	1 0 1 0	0	0	A*	P6R	0	$\sum = A*$	1 1 0 0 0 0 0 0	0 0 0 0 0 0 0 1	
11	11	1 0 1 1	0	0	A*	P6P5	0	$\sum.T3 = A*.I3$	0 0 0 0 0 0 1 1	1 1 0 0 0 0 0 0	
12	12	1 0 1 0	0	0	A*	P6R	0	$\sum = A*$	1 0 0 0 0 0 0 0	0 0 0 0 0 0 1 1	
13	13	1 0 1 1	0	0	A*	P6P5	0	$\sum.T3 = A*.I3$	0 0 0 0 0 1 1 1	1 0 0 0 0 0 0 0	
14,16...62	14	1 0 1 0	0	0	A*	P6R	0	$\sum = A*$	0 0 0 0 0 0 0 0	0 0 0 0 0 1 1 1	
15,17...63	15	1 0 1 1	0	0	A*	P6P5	0	$\sum.T3 = A*.I3$	0 0 0 0 1 1 1 1	0 0 0 0 0 0 0 0	
64	16	1 0 1 0	0	0	A*	P6R	0	$\sum = A*$	0 0 0 0 0 0 0 0	0 0 0 0 1 1 1 0	N endet hier
65	17	1 1 1 1	0	0	A*	P6P5	0	$\sum.T3 = A*.I3$	0 0 0 1 1 1 1 0	0 0 0 0 0 0 0 0	
66	18	1 1 1 0	0	0	A	A	0	\sum	0 0 1 1 1 1 0 0	0 0 0 1 1 1 1 0	M endet hier

$0.1010000 \times 0.1100000 = 0.0111100$

Dezimal bef. q $\begin{matrix} 5 & 3 \\ 8 & 2 \end{matrix}$

Multiplikationstabelle LGP 21

2.5.8. E-Befehl

$$E \hat{=} Q1 \underline{Q2} \underline{Q3} Q4 \hat{=} 1001$$

Mit dem E-Befehl können beliebige Stellen eines Wortes ausgeblendet werden. Das auszublendende Wort steht in A, die Maske (V) im Hauptspeicher und das Ergebnis wieder in A.

A bleibt "1", wenn in A und V an derselben Position eine "1" steht. Alle anderen Positionen werden "0".

Der E-Befehl entspricht einer Multiplikation ohne Berücksichtigung der einzelnen Überträge.

$$Aw' = \underbrace{F \ G \ H}_{\emptyset 4} \cdot \underbrace{Q2 \ Q3 \ Q4}_E \cdot \underbrace{V \ A}_{+ \dots}$$

d.h., nur wenn beide "1" wird eine "1" geschrieben.

2.5.9. U-Befehl

$$U \hat{=} Q1 \underline{Q2} Q3 \underline{Q4} \hat{=} 1010$$

Mit dem U-Befehl wird die Adresse im Zähler verändert, d.h. der Adreßteil des U-Befehls gibt die Hauptspeicherzelle an, aus der der nächste Befehl geholt werden soll.

Der Umlauf des C-Registers wird in $\emptyset 4$ unterbrochen und der Inhalt des R-Registers nach C geschrieben (Cw'). Die neue Information läuft danach in C um (siehe Gleichung für Cw').

In $\emptyset 2$ wurde der U-Befehl in das R-Register gebracht. In der folgenden $\emptyset 3$ braucht kein Operand gesucht zu werden.

$\emptyset 3$ wird nach einer Wortzeit beendet. G wird gesetzt.

$$G' = \underbrace{F \ G}_{\emptyset 3} \ T3 \ \underbrace{Q1 \ \underline{Q2} \ Q3}_{U(T)\text{-Befehl}} \ \underline{be} + \dots$$

└ Taste Ausführen nicht gedrückt

$$Cw' = \underbrace{F \ G \ H}_{\emptyset 4} \ T3 \ \underbrace{Q1 \ \underline{Q2} \ Q3 \ \underline{Q4}}_{U\text{-Befehl}} \ R$$

└ Inhalt des R-Registers

2.5.10. T-Befehl

$$T \hat{=} Q1 \underline{Q2} Q3 Q4 \hat{=} 1011$$

Der T-Befehl ist ähnlich dem U-Befehl. Es wird ein Sprung nach der im Adreßteil des T-Befehles angegebenen Zelle ausgeführt.

Ein Sprung erfolgt immer bei negativem Akkumulatorinhalt oder wenn bei einem -T-Befehl die PST-Taste gedrückt ist.

Der U- und T-Befehl unterscheiden sich nur im Setzen von Q4. Sind obige Bedingungen erfüllt, wird Q4 zurückgesetzt. Es folgt ein normaler Ablauf des U-Befehles (siehe 2.5.9.).

Sind die Bedingungen nicht erfüllt, bleibt Q4 gesetzt. Der Umlauf von C bleibt unverändert. Der Sprungbefehl ist unwirksam.

Die Entscheidung über einen Sprung erfolgt in Ø3.

$$Q4' = \underbrace{Q1 \underline{Q2} Q3}_{\substack{\text{T-Befehl} \\ \text{A negativ?}}} T3 A + \underbrace{Q1 \underline{Q2} Q3 T3 R}_{\text{-T-Befehl}} Tc$$

└ Signal der PST-Taste

2.5.11. H-Befehl

$$H \hat{=} Q1 \ Q2 \ \underline{Q3} \ \underline{Q4} \hat{=} 1100$$

Mit dem H-Befehl wird der Inhalt des Akkumulators (A) in eine Hauptspeicherzelle gebracht. Der Akkumulator bleibt unverändert.

Die Schreiberlaubnis W erfolgt in $\emptyset 4$

$$W = \underbrace{F \ G \ \underline{H}}_{\emptyset 4} \ \underbrace{Q1 \ Q2 \ \underline{Q3}}_{(C)H\text{-Befehl}} + \dots$$

Die Schreibinformation Vw' lautet:

$$Vw' = L (Q1 \ A + \dots)$$

$$L' = W + \dots$$

$$\underline{L'} = T3 (\underline{H} + \dots) +$$

Der Umlauf in A erfolgt über:

$$Aw' = \underline{H} \ \underline{To} \ A \ \left[\underbrace{F+G+Q1Q3Q4}_{\text{Umlauf in } \emptyset 1-\emptyset 3} + \underbrace{T3Q1Q2(Q3+Q4)+Q2(Q3+Q4)}_{\text{Umlauf in } \emptyset 4 \text{ für alle Befehle außer C}} \underline{Faf} \right]$$

2.5.12. C-Befehl

$$C \hat{=} Q1 \ Q2 \ \underline{Q3} \ Q4 \hat{=} 1101$$

Mit dem C-Befehl wird der Inhalt des Akkumulators (A) in eine Hauptspeicherzelle gebracht.

Der Akkumulator wird gelöscht.

Die Ausführung des Speicherns erfolgt wie unter 2.5.11.

Die Schreiberlaubnis W erfolgt in $\phi 4$:

$$W = \underbrace{F \ G \ H}_{\phi 4} \underbrace{Q1 \ Q2 \ \underline{Q3}}_{C(H)\text{-Befehl}} + \dots$$

Die Schreibinformation in Vw' lautet:

$$Vw' = L (Q1 \ A + \dots)$$

$$L' = W + \dots$$

$$\underline{L'} = T3 \ H + \dots$$

Der Umlauf in A wird beim Löschen unterbrochen.

$$Aw' = \underline{H} \ \underline{To} \ A \left[\underbrace{\underline{F+G+Q1Q3Q4+T3Q1Q2(Q3+Q4)+Q2(Q3+Q4)}}_{\text{Umlauf in } Q1, \phi 2, \phi 3} \ \underline{Faf} \right]$$

Umlauf in $\phi 4$ für alle Befehle außer C

2.5.13. A-Befehl

$$A \hat{=} Q1 Q2 Q3 \underline{Q4} \hat{=} 1110$$

Mit dem A-Befehl wird der Inhalt des Akkumulators (A) mit dem Inhalt einer Hauptspeicherzelle addiert. Das Ergebnis läuft in A um.

Das L-Flip Flop speichert bei der Addition in $\emptyset 4$ die Überträge. L wird am Ende einer jeden Wortzeit (ausgenommen $\emptyset 4a$) zurückgesetzt ($\underline{L}' = T3 \underline{H}$). Bei gleichzeitiger "1" in A und V wird L gesetzt, bei gleichzeitiger "0" in A und V wird L zurückgesetzt. Bei unterschiedlichem A und V bleibt die Stellung von L unverändert.

Die Eingänge der Addier-Logik werden mit I1 und I2 bezeichnet.

$$I1 = \underline{H} A + \dots$$

$$I2 = \underline{H} V + \dots$$

$$L' = \underline{T3} \underline{S} I1 I2 \underline{L} + \dots$$

$$\underline{L}' = T3 \underline{H} + \underbrace{L \underline{S} I1 I2}_{\text{unwichtig für A-Befehl. Verhindert das Rücksetzen bei einem Y-,R-,H- und C-Befehl}} \underbrace{Q2 (Q3 + \dots)}_{\text{Rücksetzer für A und V beide "0"}} + \dots$$

unwichtig für A-Befehl. Verhindert das Rücksetzen bei einem Y-,R-,H- und C-Befehl

Rücksetzer für A und V beide "0"

Das Signal S gibt die Addition an

$$\underline{S} = Q4 \underline{H}$$

T3 in der Setzbedingung für L verhindert, daß L zur T3-Zeit gesetzt wird, da dann Setzer und Rücksetzer ($\underline{L}' = T3 \underline{H}$) gleichzeitig auf L einwirken würden.

Die Schreibinformation Aw' ergibt sich dann wie folgt:

$$Aw' = (\underbrace{FG Q1Q2Q3+\dots}_{\emptyset 4 \text{ A(S)-Befehl}})(\underbrace{L\underline{I1}I2+L\underline{I1}I2+L\underline{I1}I2+L\underline{I1}I2}_{\text{Addierlogik}})+\dots$$

Die Addition negativer Zahlen erfolgt im Rechner in gleicher Weise wie die der positiven Zahlen.

Mathematisch wird sie durch Addition des Komplementes der negativen Zahlen gebildet.

Der LGP 21 bildet bei der Addition kein Komplement.

Die negativen Zahlen müssen deshalb schon als Komplementwerte gespeichert sein.

Beispiel:	-14	:	0,1110
	Kehrwert	:	1,0001
	+ "1"	:	<u>1</u>
	Komplement:		1,0010 $\hat{=}$ -14

2.5.14. S-Befehl

$$S \hat{=} Q_1 Q_2 Q_3 Q_4 \hat{=} 1111$$

Mit dem Subtraktionsbefehl wird der Inhalt (V) einer Hauptspeicherzelle vom Inhalt des Akkumulators (A) subtrahiert. Die Differenz läuft in A um.

Die Subtraktion erfolgt mathematisch, indem man das Komplement der Information vom Hauptspeicher (V) zum Akkumulator (A) addiert.

$$\begin{array}{rcl} \text{Beispiel 1: } A = 30 \hat{=} 11110 & +27 \hat{=} 11011 & \\ V = \underline{-27} \hat{=} \underline{00101} & -27 \hat{=} \begin{array}{l} 00100 \\ 00101 \end{array} & \begin{array}{l} \text{invers} \\ \text{Komplement} \end{array} \\ \quad \quad \quad 3 \quad 00011 \hat{=} 3 & & \end{array}$$

Das gleiche Ergebnis erhält man, indem man den Inhalt des Akkumulators (A) invertiert, zu der Information vom Hauptspeicher (V) addiert und das Ergebnis wieder invertiert.

$$\begin{array}{rcl} \text{Beispiel 2: } A = 30 \quad 30 \hat{=} 11110 & \text{invertiert: } 00001 & \\ V = \underline{-27} \quad 27 \hat{=} 11011 & & + \underline{11011} \\ \quad \quad \quad 3 & & 11100 \\ & & \text{invertiert: } 00011 = 3 \end{array}$$

Die Subtraktion wird im LGP 21 praktisch wie die Addition verwirklicht. Nur bei der Übertragsbildung (L-Flip-Flop) wird statt A A benutzt (s. auch 2.5.13).

$$S = Q_4 \underline{H} = \text{Angabe des Subtraktionsbefehles}$$

$$I_1 = A \underline{H}$$

$$I_2 = V \underline{H}$$

$$L' = \underline{T_3} \underline{H} S \underline{L} \underline{I_1} I_2$$

L Invertierung von A

$$\underline{L'} = \underline{T_3} \underline{H} + L S \underline{I_2} I_1 \underline{Q_2(Q_3+\dots)}$$

ist für den S-Befehl unwichtig

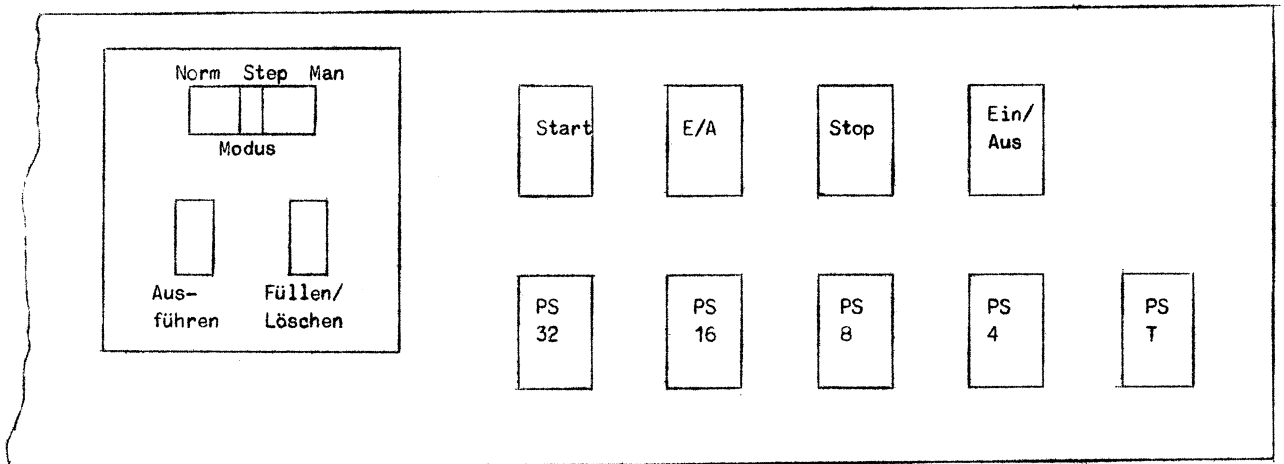
Die Schreibinformation Aw' ergibt sich dann wie folgt:

$$Aw' = (\underline{FG} \underline{Q_1 Q_2 Q_3 + \dots}) (\underline{L I_1 I_2 + L I_1 I_2 + L I_1 I_2 + L I_1 I_2})$$

\emptyset_4 (A)S-Befehl Addierlogik

2.6. Bedienungsfeld

Die Anordnung der Bedienungstasten ist in Bild 2.27 schematisch dargestellt. (Siehe auch Bild 3.37.)



1. Ein/Aus

Bild 2.24

Dieses ist der Hauptschalter, mit dem der Rechner in Betrieb gesetzt wird (Netzschalter). Die Taste leuchtet auf, wenn alle Versorgungsspannungen des Rechners vorhanden sind.

2. Stop

Das rote Feld "Stop" dient nur als Anzeige. Es leuchtet, wenn der Rechner in "Stop" steht (kein Befehlsablauf, $Kc = "1"$)

$Kc = \underbrace{F \ G \ Q2}_{\emptyset 1} \quad \underbrace{\quad}_{\text{"Stop"}}$ wird Relais B (Bedienungsfeld) nicht angesteuert.

3. E/A

Durch Drücken der E/A-Taste wird der Akkumulatorinhalt durch Unterbrechung des Umlaufes gelöscht und das jeweilige E/A-Gerät durch Zurücksetzung des entsprechenden Anwahl-Flip Flops abgewählt.

$Aw' = \underbrace{To}_{\text{E/A-Signal}} \quad \underbrace{H \ A}_{\text{Umlauf}} \quad [F + G + \dots]$

$F(n)' = T_0 + \dots$ Löschen Anwahl
!- Anwahl-Flip Flop

Die E/A-Taste leuchtet, solange ein Anwahl-Flip Flop gesetzt ist ($F_{af} \hat{=} "1"$, auch bei Manuell).

4. Start

Die Start-Taste gibt das Signal (bs) zum Beginn einer Operation. Der Stop wird hierdurch aufgehoben.

(Erklärung siehe Z-Befehl, Abschnitt 2.5.1.1.)

Die Start-Lampe leuchtet in Abwechslung mit der Stop-Lampe. Sie leuchtet, wenn Kc nicht erfüllt ist $\hat{=} 0$ Volt

$$Kc = \underbrace{F G}_{\emptyset 1}, \underbrace{Q2}_{\text{Stop}}$$

5. PST, PS4, PS8, PS16, PS32

Durch diese "Programm-Sprungtasten" hat der Bediener die Möglichkeit, manuell in ein Programm einzugreifen und verschiedene Programmteile anzurufen. Eine gedrückte Taste leuchtet.

Erklärung siehe Z-Befehl, Abschnitt 2.5.1.1. und T-Befehl, Abschnitt 2.5.

6. Modus

Der Schalter Modus ist ein Dreistellungsschalter für "Normal-Step-Manuell". Mit ihm können drei Betriebszustände ausgewählt werden:

In Stellung "Normal" durchläuft der Rechner fortlaufend automatisch das gesamte Programm entsprechend der Befehlsfolge; in Mittelstellung "Step" stoppt der Rechner nach $\emptyset 4$, führt also jeweils nur einen Befehl aus; bei "Manuell" kann der Akkumulator manuell vom Tastenfeld des Flexowriters gefüllt werden (Simulation eines I-Befehles).

6.1. Normal

Diese Schalterstellung enthält keine speziellen logischen Ausdrücke oder Operationen. Die Verriegelungen über Step und Manuell sind aufgehoben.

6.2. Step

In Stellung "Step"-Signal 01- erfolgt nach jeder Befehlsausführung ein Stop. Der Rechner bleibt in $\emptyset 1$ stehen, da das G-Flip Flop nicht gesetzt wird. Das Setzen von G wird durch Q2 entschieden (siehe auch Abschnitt 2.5.1.1.).

$$G' = \underbrace{G \ H}_{\emptyset 1} \ T3 \ K \ Q2 \ \underbrace{be \ bs}_{\text{Entscheidung}} \ Ga + \dots$$

$\underbrace{\hspace{10em}}_{\text{Faf}}$
 $\underbrace{\hspace{10em}}_{\text{nicht Start}}$

Q2 wird nach jeder Befehlsausführung zurückgesetzt.

$$\underline{Q2'} = G \ T3 \ \underbrace{01}_{\emptyset 4} \ \underbrace{Q1}_{\text{Step}} + \dots \underbrace{01 \ H'}_{\text{für M,N,D-Befehl nach Ablauf von } \emptyset 4a} + \dots$$

$\underbrace{\hspace{10em}}_{\text{alle Befehle außer M,N,D}}$

Durch Drücken der Start-Taste erfolgt der nächste Befehlsablauf (siehe auch Abschnitt 2.5.1.1. Setzen Q2).

Nach jeder Eingabe eines Zeichens erfolgt ebenfalls ein Stop:

$$\underline{Q2} = G \ T3 \ \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{\text{I-Befehl}} \ \underbrace{01}_{\text{Step}} + \dots$$

6.3. Manuell

Das Signal 01 für Step und das Signal bq für Manuell sind logisch verknüpft. Signal 01 wird "1" vor dem Signal bq, und 01 bleibt "1" während der Stellung Manuell, d.h. in Manuell erfolgt ebenfalls ein Stop wie unter 6.2. Außerdem wird das Flexowriteranwahl-Flip Flop gesetzt.

7. Füllen/Löschen

Durch Drücken der Taste Füllen/Löschen-Signal brc -, wird einmal der Akkumulatorinhalt in das R-Register geschrieben, zum anderen wird das C-Register zu bestimmten Zeiten und Befehlen gelöscht.

$$Rw' = \underbrace{A \text{ brc}}_{A \rightarrow R} + R \underbrace{\text{brc}}_{R\text{-Umlauf}} [G + \dots]$$

$$Cw' = \underbrace{\text{brc}}_{\text{Löschen}} [F \text{ S2 C } (Q_1 + \dots) + G \text{ S2 C } + G \text{ S2 C } (F + \dots) + \dots]$$

8. Ausführen

Durch Drücken der Taste "Ausführen" -Signal bespringt der Rechner von $\emptyset 1$ nach $\emptyset 3$ und führt den im R-Register stehenden Befehl aus.

$$F' = be \text{ T3} + \dots \quad (\text{Springen nach } \emptyset 3)$$

Der Rechner verharrt in $\emptyset 3$, bis die Taste "Ausführen" wieder losgelassen ist, da be in jeder Setzbedingung für das G-Flip Flop eingeschlossen ist.

Außerdem wird das Setzen aller E/A-Anwahl-Flip Flops zur Zeit $be = "1"$ verhindert.

2.7. Ein- Ausgabe-Logik (E/A-Logik)

Bei der Normalausführung des LGP 21 ist ein Flexowriter als Standard-E/A-Einheit vorgesehen. Die Ansteuerung des ersten Flexowriters erfolgt über die E/A-Steuerung I (Zchngs.-Nr. 80573). Auf dieser Karte ist noch zusätzlich die Eingabelogik für einen Tally-Leser-Anschluß.

2.7.1. Anwahl der E/A-Einheit und Unterscheidung zwischen 4- oder 6 Bit Ein-Ausgabe

Die einzelnen E/A-Einheiten werden über Flip Flops angewählt.

Das Flexowriteranwahl-Flip Flop wird zur T3-Zeit der $\emptyset 3$ durch einen (800) IO200- oder (800) PO200-Befehl bzw. durch "Manuell" gesetzt, das Tally-Leser-Anwahl-Flip Flop durch einen (800) IO000-Befehl.

$$Ff' = \underbrace{F \ G}_{\emptyset 3} \ T3 \ \underbrace{P1 \ P2 \ P3 \ P4 \ P5}_{0200} \ be \ Faf \ \underbrace{(Q1 \ Q2 \ Q3 \ Q4)}_{P\text{-Befehl}} +$$

keine andere Einheit angewählt (E/A-Verriegelung)
nicht "Ausführen"

$$+ \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{I\text{-Befehl}} + bq$$

Manuell

$$Ft' = \underbrace{F \ G}_{\emptyset 3} \ T3 \ \underbrace{P1 \ P2 \ P3 \ P4 \ P5}_{0000} \ be \ Faf \ \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{I\text{-Befehl}}$$

Es kann jeweils nur eine E/A-Einheit angewählt sein. Das Signal Faf verhindert weitere Anwahlen.

$$Faf = Ff + Ft + \dots F(n)$$

Tally-Leser weitere E/A-Einheiten
Flexowriter

Die Ein- Ausgabe kann ein 4- oder 6-Bit erfolgen. Der negative E/A-Befehl kennzeichnet 4-Bit. Die Unterscheidung erfolgt durch das K-Flip Flop. K wird zur T3-Zeit der $\emptyset 3$ bei 4-Bit Ein- Ausgabe gesetzt und bei 6-Bit zurückgesetzt.

Die Eingabe in "Manuell" erfolgt nur in 4 Bit.

$$K' = T3 \underbrace{Faf}_{\substack{\text{negativer I-} \\ \text{oder P-Befehl}}} (R + \dots) + \underbrace{bq}_{\text{Manuell}} \quad (\cong 4 \text{ Bit Mode})$$

$$\underline{K'} = F \underline{G} \underline{Faf} \underbrace{T3 \underline{R}}_{\substack{\text{kein negativer Befehl} \\ \text{P-Befehl} \quad \text{I-Befehl}}} (\underbrace{Q1 \ Q2 \ Q3 \ Q4}_{\text{P-Befehl}} + \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{\text{I-Befehl}}) \quad (= 6 \text{ Bit Mode})$$

2.7.2. Eingabe von Hand (Manuell) (s. auch 2.10.5.)

In Stellung "Manuell" erfolgt eine 4 Bit-Eingabe vom Tastenfeld des Flexowriters in den Akkumulator.

Der Rechner steht in $\emptyset 1$. In Stellung "Manuell" wird $bq = -20V$.

$Q1$ wird zur Kennzeichnung des Eingabebefehls zurückgesetzt (Simulation eines I-Befehles).

$$\underline{Q1'} = bq$$

Das Flexowriteranwahl-Flip Flop wird gesetzt:

$$Ff' = bq$$

Das K-Flip Flop wird zur Auswahl der 4 Bit-Eingabe gesetzt:

$$K' = bq$$

Beim Anschlagen einer Flexowriter-Type werden entsprechend dem jeweiligen Code die Umschaltkontakte SC6, SC8 bis SC 12 betätigt. Danach schaltet SC 7 verzögert um und legt $bq = -20V$ an die Umschaltkontakte und an JL 12. Entsprechend der Stellung von SC 6, SC 8 ... SC 12 werden die P-Flip Flops gesetzt. JL 12 leitet zur folgenden T3-Zeit durch Setzen von F die $\emptyset 3$ ein.

$$F' = Fc = JL 12 \quad \underline{Sc} \quad \underline{Sk} \quad T3$$

└─ keine unlesbaren Zeichen
└─ kein Stop Code

$$P1 = P1c \qquad P2 = P2c \qquad \text{usw. bis } P6$$

$$\underline{P1} = P1d \qquad \underline{P2} = P2d$$

Der Rechner steht in $\phi 3$ bis SC7 wieder umschaltet und -20V an JL11 legt. Hierdurch wird G gesetzt und $\phi 4$ eingeleitet.

$$G' = Gc = JL11 \quad \underbrace{F \quad G}_{\phi 3} \quad T3$$

Die P-Flip Flops werden für die Dauer der $\phi 4$ durch das i-Signal als Schiftregister geschaltet.

$$i = ic + \dots = Faf \quad \underbrace{G \quad F}_{\phi 4}$$

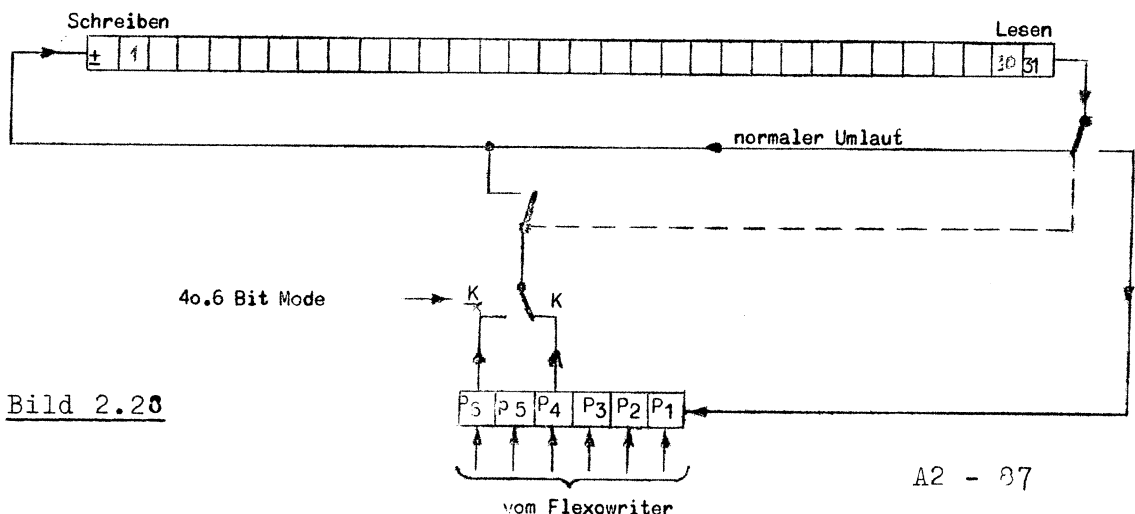
Das P-Schiftregister von P1 bis P4 wird in den Umlauf des Akkumulators eingeschlossen. Der A-Umlauf verlängert sich somit um 4 Bit (Bild

$$P1' = G \ i \ A \qquad P2' = G \ i \ P1 \qquad \text{usw. bis } P6$$

$$\underline{P1}' = G \ i \ \underline{A} \qquad \underline{P2}' = G \ i \ \underline{P1}$$

$$Aw' = Ac1 = \underbrace{F \ G}_{\phi 4} \quad Q1 \quad Faf \quad K \quad P4$$

$\phi 4$ Eingabe
Übernahme P4 (4 Bit)



Die $\emptyset 4$ wird nach einer Wortzeit beendet, und der Rechner wartet in $\emptyset 1$ auf das nächste Zeichen vom Flexowriter.

Bei den Zeichen für die Maschinenfunktionen (unten, oben, Farbe usw.) wird der SC7-Kontakt nicht betätigt. Die P-Flip Flops werden nicht gesetzt. Der Rechner bleibt in $\emptyset 1$, da $\emptyset 3$ und damit auch $\emptyset 4$ nicht gesetzt werden können.

Die 4 Bit-Information wird von P4 in die meistbedeutende Stelle von A eingeschrieben und wandert im Laufe der $\emptyset 4$ bis zur wenigstbedeutenden Stelle (Schiften um 32 Bit des erweiterten Akkumulators), d.h. die Information von P4 steht nach der $\emptyset 4$ -eine Wortzeit- in der Position 31 des Akkumulators, P3 in 30, P2 in 29 und P1 in 28. Die alte Information aus Position 31 steht nun in 27 usw.

In der $\emptyset 4$ läuft der Inhalt von A wieder normal um. Es folgt die $\emptyset 1$ und ein weiteres Zeichen kann eingegeben werden. Dann verschiebt sich in $\emptyset 4$ das vorhergehende Zeichen in A um 4 Positionen auf die Stellen 24, 25, 26, 27, da A um 4 Bit verlängert ist. Bei jeder weiteren Eingabe verschieben sich also die bereits eingegebenen Zeichen zur meistbedeutenden Stelle hin.

Die Eingabe sei beendet. Der Rechner steht in $\emptyset 1$. Die Taste "Manuell" wird zurückgesetzt, d.h. bq wird -20V. Der Flexowriter wird abgewählt.

$$\underline{Ff}' = \underline{F} \underline{G} T3 \quad Q3 \quad Xs \quad \underline{Sr}' \quad \underline{bQ}$$

$$\quad \quad \emptyset 1 \quad \quad \uparrow \quad \quad \quad \quad \quad$$

$$\quad \quad \quad \quad \quad \text{Eingabe}$$

2.7.3. Eingabe über Flexowriter-Leser Q1 Q2 Q3 Q4 = 0100
 (siehe auch 2.10.5.)

Mit einem (800) IO200-Befehl erfolgt eine (4 Bit)
 6 Bit-Eingabe vom Flexowriterleser in den Akkumulator.

In $\emptyset 3$ wird der I-Befehl in das R-Register und in die
 Q-Flip Flops gegeben.

Das Flexowriteranwahl-Flip Flop wird gesetzt. Gleich-
 zeitig werden die P-Flip Flops zurückgesetzt.

$$Ff' = F \underline{G} \ T3 \ \underline{P1} \ \underline{P2} \ \underline{P3} \ \underline{P4} \ P5 \ \underline{be} \ \underline{Faf} \ \underline{Q1} \ Q2 \ \underline{Q3} \ \underline{Q4}$$

$$\underline{P1d = P2d = P3d = P4d = P5d = P6d =}$$

Rücksetzer P1...P6

$$= \underline{F \underline{G} \ T3} \ \underline{Q1 \ Q2 \ Q3 \ Q4} \ \underline{be} \ \underline{Faf}$$

$\emptyset 3$ I-Befehl

$\emptyset 3$ wird nach einer Wortzeit beendet, da keine Operanden-
 adresse gesucht zu werden braucht.

G wird gesetzt und der Rechner steht in $\emptyset 4$.

$$G' = F \underline{G} \ T3 \ \underline{Q1} \ Q2 \ \underline{Q3} \ \underline{Q4} \ \underline{be} \ \underline{Faf}$$

Die P-Flip Flops (alle "0") werden für die Dauer von
 $\emptyset 4$ durch das i-Signal als Schiftregister geschaltet

$$i = ic1 + \dots = Faf \ F \ G + \dots$$

Das P-Schiftregister wird in den Umlauf des Akkumula-
 tors eingeschlossen.

$$Aw' = Ac1 = F \ G \ \underline{Q1} \ Q2 \ \underline{Q3} \ \underline{Q4} \ (K \ P4 + \underline{K} \ P6)$$

$$P1' = G \ i \ A \qquad P2' = G \ i \ P1 \qquad \text{usw. bis}$$

$$\underline{P1}' = G \ i \ \underline{A} \qquad \underline{P2}' = G \ i \ \underline{P1} \qquad P4 \ \text{bzw.} \ P6$$

Das Flexowriter-One Shot Sr wird für ca. 50msec gesetzt und bringt das kcrI-Relais.

Sr* = F Q1 Ff Xs bq

! Signal vom Flexowriter
(keine Eingabe)

Ø4 dauert eine Wortzeit, danach beginnt wieder Ø1.

G' = T3 H

F' = F G H T3

Über kcrI 3-4 kann das Relais KRC anziehen. KRC hält sich mit seiner zweiten Wicklung über den Kontaktweg /+/ "Start-Rechnen" / "Stop-Lesen" / krc 3-4 /kpe 21-22 / KRC /-/

Nach 50msec wird das Sr-Signal wieder negativ und KCR I fällt ab. Dadurch kann KOC anziehen.

/+/ "Start-Rechnen" / "Stop-Lesen" / SR / krc 3-4 / kcrI 21-22 / kfb 25-26 / STC-2 / KOC /-/

KOC hält sich über:

/+/ "Start-Rechnen" / "Stop-Lesen" / SR / krc 3-4/ koc 1-2 / KOC /-/

Die Leser-Kupplung LR zieht an:

/+/ SCRT 1-2 / SBS / LKL 1-2 / "Start-Lesen" / SCD 1-2 / kdc 3-4 / SF-4 / SF-1 / koc 25-26 / krc 5-6 / kmi 1-2 / LR / - /

Wenn ein Zeichen vom Lochstreifen gelesen wird, schließen die entsprechenden Leserkontakte SR(n) und die entsprechenden Translator-Magnete LT sowie die Translator-Kupplung LTC ziehen an. Das gewünschte Zeichen wird über die Typenhebel ausgedruckt.

Durch das Anschlagen der Type werden entsprechend dem jeweiligen Code die Umschaltkontakte SC6, SC8 ... SC12 betätigt. Danach schaltet SC7 verzögert um und legt -20V an die Umschaltkontakte und an JL12.

Ff = -20V/JL32/koc28-29/krc25-26/SC7/

Entsprechend der Stellung von SC6, SC8 ... SC12 werden die P-Flip Flops gesetzt. JL12 leitet zur folgenden T3-Zeit durch Setzen von F die Ø3 ein.

$$F' = Fc = JL12 \quad \underline{SC} \quad \underline{Sk} \quad T3$$

└─ keine unlesbaren Zeichen
└─ kein Stop Code

$$\begin{array}{ll} \underline{P1^*} = \underline{P1c} & \underline{P2^*} = \underline{P2c} \\ \underline{P1^*} = \underline{P1d} & \underline{P2^*} = \underline{P2d} \end{array} \quad \text{usw. bis P6}$$

Der Rechner steht in Ø3 bis SC7 wieder umschaltet und -20V an JL11 legt. Hierdurch wird G gesetzt und Ø4 eingeleitet.

$$G' = Gc = JL11 \quad F \quad \underline{G} \quad T3$$

Die P-Flip Flops werden für die Dauer der Ø4 durch das i-Signal als Schiftregister geschaltet und das P-Schiftregister in den A-Umlauf eingeschlossen (s. Bild 2.28)

$$\begin{array}{ll} P1' = G \text{ i } A & P2' = G \text{ i } P1 \\ \underline{P1'} = G \text{ i } \underline{A} & \underline{P2'} = G \text{ i } \underline{P1} \end{array} \quad \text{usw. bis P6}$$

$$Aw' = Ac1 = F \quad G \quad \underline{Q1} \quad Q2 \quad \underline{Q3} \quad \underline{Q4} \quad (\underline{K P4} + \underline{K P6})$$

Übernahme von P4 bzw. P6

Die Ø4 wird nach einer Wortzeit beendet, und der Rechner wartet in Ø1 auf das nächste Zeichen vom Flexowriter. Beim Stop-Code öffnet der Leserkontakt SR1. Die Relais KOC und KRC und damit auch die Leserkupplung LR fallen ab.

Das Signal Xs wird OV über koc 21-22, krc 21-22 und Ff wird zurückgesetzt.

$$\underline{Ff'} = \underline{F} \quad \underline{G} \quad \underline{Q3} \quad T3 \quad \underline{Sr'} \quad Xs \quad \underline{bq}$$

Faf wird wieder OV.

Bei den Zeichen für die Maschinenfunktionen (unten, oben, Farbe usw.) wird der SC7-Kontakt nicht betätigt. Die P-Flip Flops werden nicht gesetzt. Der Rechner bleibt in $\emptyset 1$, da $\emptyset 3$ und damit auch $\emptyset 4$ nicht gesetzt werden können.

2.7.4. Links-Schiften $Q_1 Q_2 Q_3 Q_4 \hat{=} 0100$

Der I-Befehl wirkt gleichzeitig als Schifftbefehl. Der Akkumulator wird um 4 bzw. 6 Bit zur meistbedeutenden Stelle hin geschifft, und in die wenigstbedeutenden Stellen werden Nullen geschifft.

Falls nur geschifft werden soll, darf die Adresse des I-Befehls keine Ein- Ausgabeeinheit anwählen.

Der Funktionsablauf ist wie unter 2.7.3.

Die P-Flip Flops werden zur T3-Zeit der $\emptyset 3$ gelöscht. Hierdurch werden Nullen in die wenigstbedeutenden Stellen geschrieben.

$$P1d = P2d = \dots P6d = F \ G \ T3 \ \underline{Q_1} \ Q_2 \ \underline{Q_3} \ \underline{Q_4} \ \underline{be} \ \underline{Faf}$$

Das aus den P-Flip Flops gebildete Schifftregister wird in den Umlauf von A mit eingeschlossen:

$$i = ic = \underbrace{F \ G \ \underline{Q_1} \ Q_2 \ \underline{Q_3} \ \underline{Q_4}}_{\emptyset 4 \text{ Schifft-Befehl (I)}}$$

$$Aw' = Ac1 = F \ G \ \underline{Q_1} \ Q_2 \ \underline{Q_3} \ \underline{Q_4} \ (K \ P4 + \underline{K} \ P6)$$

Da Faf nicht OV wird, gilt dieser "Eingabe Zustand" nur für einen Phasenablaufzyklus und wird wie ein gewöhnlicher Befehlsablauf interpretiert.

2.7.5. Ausgabe über Flexowriter (Druckbefehl)

$$P \hat{=} Q1 \underline{Q2} \underline{Q3} \underline{Q4} \hat{=} 1000$$

Siehe auch Blockschaltbild 2.10.6.

Mit einem (800) P0200-Befehl wird das Zeichen, das durch die an den meistbedeutenden Stellen im Akkumulator stehenden (4 Bit) 6 Bit gekennzeichnet ist, ausgedruckt. Der Akkumulatorinhalt bleibt unverändert.

$\emptyset 1$ und $\emptyset 2$ verlaufen normal. $\emptyset 3$ wird auf eine Wortzeit begrenzt, da keine Operandenadresse gesucht zu werden braucht.

$$G' = \underbrace{F \ G \ T3}_{\emptyset 3} \underbrace{\underline{Q2} \ \underline{Q3} \ \underline{Q4} \ \text{be} \ \underline{Faf}}_{\text{P-Befehl}} + \dots$$

Gleichzeitig wird das Flexowriteranwahl-Flip Flop Ff gesetzt und die 4- oder 6 Bit Ausgabeunterscheidung vorgenommen (s. 2.7.1.).

$\emptyset 4$ dauert eine Wortzeit.

Die P-Flip Flops werden in $\emptyset 4$ als Schiftregister geschaltet. Während des Umlaufs von A wird sein Inhalt gleichzeitig in die P-Flip Flops geschiftet. Nur die 6 meistbedeutenden Bit werden in den P-Flip Flops gehalten.

$$i = \underbrace{F \ G \ Faf}_{\emptyset 4}$$

$$\begin{array}{lll} P1' = G \ i \ A & P2' = i \ P1 & P3' = i \ P2 \\ \underline{P1}' = G \ i \ \underline{A} & \underline{P2}' = i \ \underline{P1} & \underline{P3}' = i \ \underline{P2} \end{array} \quad \text{usw. bis } P6$$

$$Aw' = \underline{H} \ A \ \underline{To} \ \underline{Q1} \ \underline{Q3} \ \underline{Q4}$$

|
Umlauf P(H)-Befehl

Am Ende $\emptyset 4$ wird zur Kennzeichnung des P-Befehls für den weiteren Ablauf $Q3$ gesetzt.

$$Q3' = G \ T3 \ \underbrace{Q1 \ \underline{Q2} \ \underline{Q3} \ \underline{Q4}}_{\text{P-Befehl}}$$

Der eigentliche Druckvorgang beginnt in der folgenden $\emptyset 1$.

$$\underline{F}' = F \ G \ \underline{H} \ T3$$

$$\underline{G}' = G \ \underline{H} \ T3$$

Bei der 4 Bit Ausgabe werden nur die hexadezimalen Zeichen ausgedruckt. Dazu wird P5 gesetzt und P6 zurückgesetzt.

$$P1, P2, P3, P4, P5, P6 \hat{=} \underline{XXXX}10$$

beliebig

$$P5' = \underline{F} \ \underline{G} \ K \ Q3 \ Faf$$

$$\underline{P6}' = \underline{\underline{F \ G}} \ K \ Q3 \ Faf$$

$\emptyset 1$ | P-Befehl
 |
 4 Bit

Das X-Flip Flop löst den Druckvorgang aus und hält den Rechner in $\emptyset 1$ bis der Druckvorgang ausgeführt ist, wenn ein neuer Druckvorgang folgt.

$$X' = \underline{\underline{F \ G \ Tx}} \ Q3 \ T3 \ Ff \ (JL-33)(krc \ 23-24)(kfb \ 5-6) \cdot$$

$\emptyset 1$ | (P-Befehl)
 |
 Translator Kupplungstreibersignal
 ·(STC-4)(koc 3-4)

Die Translatortreiber TP1...TP6 und die Translatorkupplung LTC werden entsprechend der jeweiligen Code-Kombination der P-Flip Flops gesetzt:

$$TP1 = X \ P1$$

$$TP2 = X \ P2$$

$$TP6 = X \ P6$$

$$Tx = X \ PCP \ (\text{Translator-Kupplungstreibersignal})$$

Die Druckoperation wird ausgeführt, damit kann die Ausgabeanwahl gleichzeitig zurückgenommen werden.

$$\underline{Ff} = X$$

Hierdurch wird auch Faf wieder "1", und der Rechner befindet sich wieder in einer normalen $\emptyset 1$.

X bleibt nur eine Taktzeit lang gesetzt (10 μ sec).

$$\underline{X'} = \frac{\underline{F} \underline{G} \underline{Tx} \underline{Q3} \underline{T3} \underline{Ff} (\underline{JL-33})(\underline{krc} \underline{23-24})(\underline{kfb} \underline{5-6}) \cdot (\underline{STC-4})(\underline{koc} \underline{3-4})}{}$$

Falls mehrere Druckbefehle einander folgen, muß der Rechner solange in $\emptyset 1$ warten, bis der jeweilige Druckbefehl ausgeführt ist.

X kann erst gesetzt werden, wenn Tx und JL-33 beide "1" sind. Der Rücksetzer für die Translator-Treiber und für die Translatorkupplung wird durch STC-3 bestimmt.

Tx wird "1" über:

-20V, JL-28, koc 5-6, STC-3, JL-29.

JL-33 wird "1" über:

Null, JL-31, koc 3-4, STC-4, kfb 5-6, krc 23-24, JL-33.

2.7.6. Eingabe über Tally-Leser

$$I \hat{=} \underline{Q1} \underline{Q2} \underline{Q3} \underline{Q4} \hat{=} 0100$$

(Siehe auch Blockschaltbild 2.10.7.)

Mit einem (800) I0000-Befehl erfolgt eine (4-Bit) 6-Bit Eingabe vom Tally-Leser in den Akkumulator.

In $\emptyset 3$ wird der I-Befehl in das R-Register und in die Q-Flip Flops gegeben (s. 2.4.2.3.).

Das Tally-Anwahl-Flip Flop wird gesetzt. Gleichzeitig werden die P-Flip Flops zurückgesetzt.

$$Ft' = F \underline{G} \underline{T3} \underline{P1} \underline{P2} \underline{P3} \underline{P4} \underline{P5} \underline{be} \underline{Faf} \underline{Q1} \underline{Q2} \underline{Q3} \underline{Q4}$$

$$\underline{P1d = P2d = P3d = P4d = P5d = P6d =}$$

Rücksetzer für P1...P6

$$= \underbrace{F \ G}_{\emptyset 3} \ T3 \ \underbrace{Q1 \ Q2 \ Q3 \ Q4}_{I\text{-Befehl}} \ \underline{be} \ \underline{Faf}$$

$\emptyset 3$ wird nach einer Wortzeit beendet, da keine Operandenadresse gesucht zu werden braucht.

G wird gesetzt und der Rechner steht in $\emptyset 4$.

$$G' = F \ \underline{G} \ T3 \ \underline{Q1} \ Q2 \ \underline{Q3} \ \underline{Q4} \ \underline{be} \ \underline{Faf}$$

Die P-Flip Flops (alle "0") werden für die Dauer von $\emptyset 4$ durch das i-Signal als Schiftregister geschaltet.

$$i = ic1 + \dots = Faf \ F \ G + \dots$$

Das P-Schiftregister wird in den Umlauf des Akkumulators eingeschlossen.

$$Aw' = Ac1 = F \ G \ \underline{Q1} \ Q2 \ \underline{Q3} \ \underline{Q4} \ (K \ P4 + \underline{K} \ P6)$$

$$\begin{array}{ll} P1' = G \ i \ A & P2' = G \ i \ P1 \\ \underline{P1}' = G \ i \ \underline{A} & \underline{P2}' = G \ i \ \underline{P1} \end{array} \quad \text{usw. bis P6}$$

$\emptyset 4$ dauert eine Wortzeit, danach beginnt wieder $\emptyset 1$.

Zur Synchronisierung des Tally-Lesers mit dem Rechner-takt wird das Flip Flop Di gesetzt.

$$Di' = \underbrace{F \ G}_{\emptyset 1} \ Ft \ \underline{Xp}$$

| └─ astabiler Multivibrator
Tally-Anwahl-Flip Flop

Die Ansteuerung des Tally-Leser erfolgt über

$$R_p = D_t = D_i X_p$$

| |
| | Tally Multivibrator
| | Tally-Leser Kupplungstreiber Signal

$$\underline{R_p} = T_r$$

|
| Signal für die Tally-Leser Kontakte

Solange D_i gesetzt ist, verläuft das Signal R_p , $\underline{R_p}$, T_r bzw. D_t im Takte des Tally-Multivibrators X_p (4,5msec).

Der 4,5 msec Impuls gibt das Treibersignal für die Kupplung.

Sobald R_p "1" wird, geht der Rechner zur T3-Zeit durch Setzen von F in $\emptyset 3$:

$$F_c = \underline{F} \underline{G} T_3 \underline{S_c} \underline{C_d} \underline{S_k} F_t R_p$$

| | |
| | | kein Maschinenzeichen
| | | kein Code delete
| | | kein Stop Code

$$F' = \underline{F_c} + \dots$$

Die Tally-Leser Kupplung zieht an (4,5msec Impuls). Bevor jedoch das nächste Zeichen erreicht wird (nach ca. 4msec), werden entsprechend der Lage der Tally-Leser Kontakte über das Signal $T_r = \underline{R_p} = -20V$ die entsprechenden P-Flip Flops gesetzt.

$$P_1 = P_{1c}$$
$$\underline{P_1} = P_{1d} \quad \text{usw. bis } P_6$$

Sobald $R_p = -20V$ wird (nach 4,5msec), geht der Rechner zur T3-Zeit durch Setzen von G in $\emptyset 4$

$$G_c = \underline{F} \underline{G} T_3 F_t \underline{R_p}$$

|
| $\emptyset 3$

$$G' = \underline{G_c} + \dots$$

Die P-Flip Flops werden für die Dauer der $\emptyset 4$ durch das i-Signal als Schiftregister geschaltet und das P-Schiftregister in den A-Umlauf eingeschlossen (s. Bild 2.28).

$$\begin{array}{ll}
 P1' = G \text{ i } A & P2' = G \text{ i } P1 \\
 \underline{P1}' = G \text{ i } \underline{A} & \underline{P2}' = G \text{ i } \underline{P1} \quad \text{usw. bis } P6
 \end{array}$$

$$\begin{array}{l}
 Aw' = Ac1 = \underbrace{F \ G \ Q1 \ Q2 \ Q3 \ Q4}_{\emptyset 4} \underbrace{(K \ P4 + K \ P6)}_{\substack{\text{I-Befehl} \\ \text{Übernahme von } P4 \text{ bzw. } P6}}
 \end{array}$$

Die Phase 4 wird nach einer Wortzeit beendet und der Rechner wartet in $\emptyset 1$ auf das nächste Zeichen vom Leser.

Wird ein Stop Code, ein Code delete oder ein Maschinenzeichen gelesen, wird F zur T3-Zeit der $\emptyset 3$ zurückgesetzt und der Rechner steht in $\emptyset 1$.

$$\begin{array}{l}
 \underline{F}' = \underline{F_d} = \underbrace{F \ G}_{\emptyset 3} \text{ T3 } (\underbrace{Sc}_{\text{Stop Code}} + \underbrace{Cd}_{\text{Code delete}} + \underbrace{Sk}_{\text{Maschinenzeichen}}) \text{ Ft}
 \end{array}$$

Durch einen Stop Code oder durch Drücken der E/A-Taste wird die Tally-Leser-Anwahl zurückgenommen.

$$\begin{array}{l}
 \underline{F_t}' = \underbrace{F \ G}_{\emptyset 1} \text{ T3 } \underbrace{Sc}_{\text{Stop Code}} \underbrace{Xp}_{\text{E/A-Taste}} + \text{To}
 \end{array}$$

Hierdurch wird auch Faf wieder "1" und damit Di zurückgesetzt:

$$\underline{D1}' = \underline{Faf}$$

2.8. Bedeutung der Buchstaben in den logischen Gleichungen

2.8.1. Flip Flops

- F,G,H Phasen-Flip Flops. Sie stellen $\emptyset 1 \dots \emptyset 4a$ dar.
($\emptyset 1 = \underline{F} \underline{G} \underline{H}$; $\emptyset 2 = \underline{F} \underline{G} \underline{H}$; $\emptyset 3 = \underline{F} \underline{G} \underline{H}$;
 $\emptyset 4 = \underline{F} \underline{G} \underline{H}$; $\emptyset 4a = \dots H$)
- Q1...Q4 Befehls-Flip Flops. Sie werden zur Speicherung des Befehles für die Befehlsausführung benutzt (Schiftregister).
Mehrfach ausgenutzt wird:
Q1 als Sprungindikator
Q2 als Stopindikator
Q3 als Ausgabebefehlindikator.
- P1...P6 Spuradreß- und E/A-Flip Flops. Sie werden zur Speicherung der Spuradresse und bei einem Ein-/Ausgabe-Befehl zur Speicherung der Information benutzt (Schiftregister).
Mehrfach ausgenutzt wird:
P1 zur Unterscheidung geradzahliger und ungeradzahliger Wortzeiten in $\emptyset 4a$ beim M,N,D-Befehl.
P5 zur Speicherung des Vorzeichens des Operanden (Divisor, Multiplikand) vom Hauptspeicher beim M,N,D-Befehl.
P6 zur Speicherung des Vorzeichens des Dividenden bzw. des laufenden Restes und des Multiplikators.
- K Entscheidungs-Flip Flop für Sektorübereinstimmung in $\emptyset 1$ und $\emptyset 3$.
Übertrags-Flip Flop in $\emptyset 2$ bei der Addition von 1 zum C-Register.
Übertrags-Flip Flop in $\emptyset 4$ für die Addition von 1 zum C-Register bei einem R-Befehl.
Entscheidungs-Flip Flop bei M,N,D-Befehl für die Sektorübereinstimmung in der 64. Wortzeit.
Als Indikator für 4 oder 6 Bit Mode bei der Ein-/Ausgabe oder Links-Schift-Befehl.

L Übertrags-Flip Flop beim A,S,M,N,D-Befehl.
Es zwingt beim Speicher-Befehl eine "0" in
die Spacer-Bit-Stelle.

F_f, F_t Anwahl-Flip Flop für E/A-Geräte.

2.8.2. Logische Ausdrücke

i Schaltet die P-Flip Flops als Schift-Register
zur Übernahme der Spuradresse oder eines Zei-
chens bei der Ein-/Ausgabe.

α_{11} Schaltet die Q-Flip Flops als Schift-Register
zur Übernahme des Befehls-Code.

J1 Addierer-Eingang (vom A-Register usw.).

J2 Addierer-Eingang (vom Hauptspeicher usw.).

B3 Arithmetischer Übertragsausdruck.

$\alpha_5 - \alpha_8$ Überlauf-Begriffe

S Entscheidungsüber Addition - Subtraktion.

W Schreiberlaubnis bei einem Schreibbefehl.

r1 Eingangssignal zu dem P-Flip-Flop-Schift-
Register.

Kc Gibt den blockierten Zustand an und steuert
die Stop- und Start-Anzeigelampe.

Faf Zeigt den Ein/Ausgabe-Zustand an.

T3 Vorzeichenzeit ($T_3 = S_1 S_3$).

Index a Verriegelung für die Setzseite eines Flip Flops
Index b Verriegelung für Rücksetzseite eines Flip Flops

Index c Setzer für ein Flip Flop
Index d Rücksetzer für ein Flip Flop

2.8.3. Schreibsignale:

Aw' Schreibsignal für das A-Register
(Akkumulator)

Cw' Schreibsignal für das C-Register
(Zählregister)

Rw' Schreibsignal für das R-Register
(Befehlsregister)

Vw' Schreibsignal für den Hauptspeicher

2.8.4. Lesesignale

A Lesesignal vom A-Register
A* Lesesignal vom A*-Register
C Lesesignal vom C-Register
R Lesesignal vom R-Register
V Lesesignal vom Hauptspeicher
S1 Lesesignal Sektoradresse
S2 Lesesignal Sektor-Spuradreßzeit
S3 Lesesignal Spur-Befehls-T3-Zeit

2.8.5. Bedienungsfeld

be Taste "ausführen" gedrückt
bq Modus-Taste in "Manuell"
brc Taste "Füllen/Löschen"

bs	Taste "Start"
O1	Modus-Taste in "Step"
Tb1-Tb4	Signal der PS-Tasten
Tc	Signal der PST-Taste
To	Signal der E/A-Taste

2.8.6. Ein-/Ausgabe (E/A-Steuerung 1, 80573)

Cd	Code delete
Di	Synchronisierungs-Flip Flop für Tally-Leser
Dt	Tally-Leser Kupplungstreibersignal
Ff	Flexowriteranwahl-Flip Flop
Ft	Tally-Leser-Anwahl-Flip Flop
JL 11	Ende Eingabezyklus Flexowriter (Setzsignal für Ø4)
JL 12	Anfang Eingabezyklus Flexowriter (Setzsignal für Ø3)
JL 29	Rücksetzer für die Translatortreiber
P1*...P6*	Zeicheninformation vom Flexowriter
Rs1...Rs6	Zeicheninformation vom 1. Tally-Leser
Rp	One Shot, legt Dauer des Kupplungstreiberimpulses beim 1. Tally-Leser fest
Sc	Stop Code
Sk	Maschinenfunktion (Farbe, oben, unten usw.)
Sr	One Shot, gibt den Flexowriter zur Eingabe frei
Sr*	Setzbedingung für das Sr-One Shot
Tp1...Tp6	Treibersignale für die Translator-Magnete des Flexowriters

Tr1	Impuls zu den Abfühlkontakten des Tally-Lesers
Tx	Flexowriter Kupplungstreibersignal
X	Steuer-Flip Flop für die Ausgabe zum Flexowriter
Xp	Astabiler Multivibrator, gibt die Dauer des Tally-Leser-Zyklus an
Xs	Signal vom Flexowriter, gibt Ende des Eingabezyklus an
B5	Definiert Drucken in Ø1 bei 4 Bit-Mode.

2.9. Logische Funktionsgleichungen

$$F' = \underbrace{\underline{F} \underline{G} \underline{H} \underline{T3}}_{\emptyset 4a} P1 + \text{Ende WZ 3 in } \emptyset 4a + \underbrace{\underline{F} \underline{G} \underline{H} \underline{T3} \underline{Q1}}_{\emptyset 2} + \text{gehe in } \emptyset 3 \text{ beim Z-Befehl} + \text{Sprungindikator}$$

$$+ \text{"ausführen"} + \text{gehe in } \emptyset 3 \text{ (v. E/A-Gerät)}$$

$$+ \underline{be} \underline{T3} + \underline{Fc}$$

$$\text{"ausführen"}$$

$$\underline{F}' = \underbrace{\underline{F} \underline{G} \underline{H} \underline{T3}}_{\emptyset 4} + \text{Ende } \emptyset 4 + \underbrace{\underline{F} \underline{G} \underline{T3} \underline{P1}}_{\emptyset 4a} + \text{WZ 66, D, M-Befehl} +$$

$$+ \underline{WZ 64 (Ende) N-Befehl} + \text{Stop Code, Code Delete}$$

$$+ \underline{\underline{F} \underline{H} \underline{T3} \underline{Q4}} \underline{K} + \underline{Fd}$$

$$\text{Sektorvergleich}$$

$$\text{N-Befehl}$$

$$G' = \underbrace{\underline{F} \underline{G} \underline{T3}}_{\emptyset 3} + \text{gehe in } \emptyset 4 \text{ bei Eingabe} +$$

$$\underline{\underline{Q1} \underline{Q2} \underline{Q3} \underline{Q4}} \underline{be} \underline{\underline{Faf}} +$$

$$\text{I-Befehl} \quad \text{E/A-Verriegelung}$$

$$\text{nicht "ausführen"}$$

$$+ \text{gehe in } \emptyset 4 \text{ nach Sektorvergleich} +$$

$$+ \underline{\underline{F} \underline{G} \underline{H} \underline{T3}} \underline{K} \underline{Ga} \underline{be} +$$

$$\text{Sektorvergleich}$$

$$+ \text{gehe in } \emptyset 2 \text{ nach Sektorvergleich, falls kein Stop} +$$

$$+ \underline{\underline{G} \underline{H} \underline{T3}} \underline{K} \underline{Ga} \underline{Q2} \underline{be} \underline{bs} +$$

$$\text{Sektorvergl.} \quad \underline{\underline{Faf}} \quad \text{kein Stop} \quad \text{nicht "ausführen"}$$

$$+ \text{Ende WZ 64, D,M-Befehl} +$$

$$+ \underline{\underline{G} \underline{H} \underline{T3}} \underline{K} \underline{Ga} \underline{Q4} \underline{be} +$$

$$\text{Sektorvergleich} \quad \underline{\underline{Faf}} \quad \text{D,M-Befehl} \quad \text{nicht "ausführen"}$$

$$+ \text{gehe in } \emptyset 4 \text{ nach 1 WZ, U,T-Befehl} +$$

$$+ \underline{\underline{F} \underline{G} \underline{T3}} \underline{\underline{Q1} \underline{Q2} \underline{Q3}} \underline{be} +$$

$$\text{U,T-Befehl} \quad \text{nicht "ausführen"}$$

$$+ \text{gehe in } \emptyset 4 \text{ nach 1 WZ bei Z,P-Befehl} \quad \text{Ende Eingabezyklus}$$

$$+ \underline{\underline{F} \underline{G} \underline{T3}} \underline{\underline{Q2} \underline{Q3} \underline{Q4}} \underline{be} \underline{\underline{Faf}} + \underline{Gc} \quad \text{(gehe in } \emptyset 4)$$

$$\text{Z,P-Befehl} \quad \text{E/A-Verriegelung}$$

$$\text{nicht "ausführen"}$$

Phasensteuerung (80570)

$$\underline{G}' = \underbrace{\underline{G} \ \underline{H} \ T3}_{\emptyset 4} + \text{Ende } \emptyset 4 \text{ nach 1 WZ} + \underbrace{\underline{F} \ \underline{G} \ T3}_{\emptyset 2 \text{ oder WZ 67 beim D-Befehl}} + \text{Ende } \emptyset 2 \text{ nach 1 WZ oder Ende D-Befehl} +$$

$$+ \text{ Ende WZ 66 beim M-Befehl} + \underbrace{\underline{G} \ \underline{H} \ \underline{P1} \ T3 \ Q3}_{\emptyset 4a} + \underline{Gd} \quad \text{M-Befehl}$$

gehe in $\emptyset 4a$ nach 1 WZ $\emptyset 4$ beim M,N,D-Befehl

$$\underline{H}' = \underbrace{\underline{F} \ \underline{G} \ \underline{H} \ T3}_{\emptyset 4} \underbrace{\underline{Q1} \ \underline{Q2} \ \alpha 9}_{(3 + Q4)} \quad \text{M,N,D-Befehl}$$

$$\underline{H}' = \underbrace{\underline{F} \ \underline{G} \ \underline{H} \ T3}_{\emptyset 4a} + \text{Ende WZ 67 beim D-Befehl} + \text{Ende WZ 64 beim N-Befehl} + \underline{H} \ T3 \ K \ \underline{Q4} +$$

$$+ \text{ Ende WZ 66 beim M-Befehl} + \beta 6 \quad \text{= G H T3 Q3 P1}$$

$$\beta 6 = \underbrace{\underline{G} \ \underline{H} \ T3 \ Q3 \ \underline{P1}}_{\emptyset 4a} \quad \text{M-Befehl}$$

$$\underline{Kc} = \underbrace{\underline{F} \ \underline{G} \ \underline{Q2}}_{\emptyset 1} \quad \text{Stop (Anzeige) Stop-Indikator}$$

$$\underline{Rw}' = \underbrace{\underline{F} \ \underline{G} \ \underline{H} \ V}_{\emptyset 2} + \text{übernehme V in } \emptyset 2 + \underbrace{\underline{G} \ \underline{H} \ V \ \underline{Q1} \ \underline{Q2} \ \alpha 9}_{\emptyset 4} + \text{übernehme V in } \emptyset 4 \text{ beim M,N,D-Befehl} + \dots \quad \text{Hauptspeicher-information} \quad \text{M,N,D-Befehl} \quad \text{...(Q3+Q4)}$$

$$+ \underline{A} \ \underline{brc} \quad \text{übernehme A wenn "füllen/löschen" gedrückt} + \quad \text{füllen/löschen} + \quad \text{Akkumulatorinhalt}$$

$$+ \underline{R} \ \underline{brc} \left[\underbrace{\underline{G}}_{\emptyset 1, \emptyset 3} + \underbrace{\underline{H}}_{\emptyset 4a} + \underbrace{\underline{F}}_{\emptyset 4} \left(\underline{Q1} + \underline{Q2} + \underline{Q3} \ \underline{Q4} \right) \right] \quad \text{Umlauf von R in } \emptyset 1, \emptyset 1, \emptyset 4a \text{ und in } \emptyset 4 \text{ bei allen Befehlen außer M,N,D-Befehl}$$

nicht füllen/löschen R-Umlauf

I-Befehl Z,B,Y,R-Befehl P,E,U,T,H,C,A,S-Befehl

Phasensteuerung (80570)

$$Vw' = \underbrace{L}_{\text{addiere 1 zum Zähler im C beim R-Befehl}} \left(\underbrace{Q1 \ Q4 \ K \ C}_{\text{R-Befehl}} + \underbrace{Q1 \ Q4 \ K \ C}_{\text{R-Befehl}} + \right)$$

schreibe 0 zur Spacerzeit Addition von 1

$$+ \underbrace{Q1 \ A}_{\substack{\text{übernehme A beim C,H-Befehl} \\ \text{A-Inhalt} \\ \text{C,H-Befehl}}} + \underbrace{Q4 \ A}_{\substack{\text{übernehme A beim Y-Befehl} \\ \text{A-Inhalt} \\ \text{Y-Befehl}}}$$

$$W = \underbrace{F \ G \ H}_{\emptyset 4} \underbrace{Q1 \ Q2 \ Q3}_{\text{Schreiberlaubnis in } \emptyset 4 \text{ beim H,C-Befehl}} +$$

$$+ \underbrace{F \ G \ S2}_{\emptyset 4} \underbrace{Q1 \ Q2 \ Q3}_{\substack{\text{Schreiberlaubnis zur Adreßzeit in } \emptyset 4 \text{ beim Y,R-Befehl} \\ \text{Y,R-Befehl} \\ \text{Adreßzeit}}}$$

$$r1 = \underbrace{F \ C}_{\emptyset 1} + \underbrace{H \ C}_{\emptyset 4} + \underbrace{F \ H \ R}_{\emptyset 3}$$

übernehme C in $\emptyset 1$ übernehme C in $\emptyset 4a$ übernehme R in $\emptyset 3$

$$Faf = Ff + Ft + F(n)$$

Flexowriter-Anwahl Tally-Anwahl usw.

$$K' = bQ_{\text{Manuell}} + T3 \underbrace{Faf}_{\emptyset 1, \emptyset 2, \emptyset 4} (\underbrace{F + G + R}_{\text{Setze K zu jeder T3-Zeit in } \emptyset 1, \emptyset 2, \emptyset 4 \text{ und}} + \underbrace{Q1 + Q2 + Q3 + Q4}_{\text{jeder Befehl außer +I-Befehl}}) \cdot$$

$$\cdot (\underbrace{F + G + R}_{\text{bei einem -Befehl, nicht beim +E/A-Befehl}} + \underbrace{Q1 + Q2 + Q3 + Q4}_{\substack{\emptyset 1, \emptyset 2, \emptyset 4 \\ \text{jeder Befehl außer +P-Befehl}}})$$

$$\underline{K}' = (\text{siehe Rechenwerk 80571, Seite A2-113})$$

Phasensteuerung(80570)

Schalte die P-Flip Flops in 04 als Schiftregister für

$$Ac1 = \underbrace{F}_{04} \underbrace{G}_{I\text{-Befehl}} \underbrace{Q1}_{n.U.} \underbrace{BL}_{E/A\text{-Mode}} \left(\underbrace{Faf}_{I\text{- und Schiftbefehl}} + \underbrace{Q1 Q2 Q3 Q4}_{I\text{- und Schiftbefehl}} \right) \cdot$$

4- oder 6-Bit Mode

$$\bullet \left(\underbrace{K}_{\text{übernehme P4}} \underbrace{P4}_{-(I)\text{Befehl}} + \underbrace{K}_{\text{übernehme P6 (6 Bit)}} \underbrace{P6}_{+(I)\text{Befehl}} \right)$$

Schalte die P-Flip Flops in 04 als Schiftregister bei einem I-Befehl

$$ic = \underbrace{F}_{04} \underbrace{G}_{n.U.} \underbrace{BL}_{E/A\text{-Mode}} \left(\underbrace{Faf}_{I\text{- und Schiftbefehl}} + \underbrace{Q1 Q2 Q3 Q4}_{I\text{- und Schiftbefehl}} \right)$$

$$P1d = P2d = P3d = P4d = P5d = P6d =$$

Rücksetzer für P's beim I, Schiftbefehl zur T3-Zeit 03

$$= \underbrace{F}_{03} \underbrace{G}_{I\text{- und Schiftbefehl}} \underbrace{T3}_{E/A\text{-Verriegelung}} \underbrace{Q1 Q2 Q3 Q4}_{\text{nicht "ausführen"}} \underbrace{be}_{E/A\text{-Verriegelung}} \underbrace{Faf}_{\text{nicht "ausführen"}}$$

E/A-Verriegelung

$$P1b = P5a = P6a = ia = Ga = \underbrace{Faf}_{E/A\text{-Verriegelung}}$$

Vorzeichenzeit

$$T3 = S1 S3$$

$$Aw' = \frac{\begin{matrix} \text{übernehme V beim B-Befehl, und V.A beim E-Befehl} \\ \text{F G H} \\ \text{Ø4} \end{matrix}}{\text{B,E-Befehl}} \frac{\begin{matrix} \text{Q2 Q3 Q4} \\ \text{V (Q1 + A)} \end{matrix}}{\text{+}}$$

$$+ \frac{\begin{matrix} \text{A-Umlauf in Ø1...Ø3, in Ø1...Ø4 beim P,H,U,T,Y,R,Z-Befehl} \\ \text{H A Tø} \end{matrix}}{\text{nicht Ø4a E/A nicht Ø1,Ø2 Ø3}} \left[\frac{\begin{matrix} \text{F} \\ \text{+} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{G} \\ \text{+} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{Q1 Q3 Q4} \\ \text{P,H-Befehl} \end{matrix}}{\text{+}}$$

$$+ \text{ und beim D,N,M-Befehl außer Vorzeichen} \\ + \frac{\begin{matrix} \text{T3 Q1 Q2} \\ \text{D,N,M-Befehl} \end{matrix}}{\text{+}} \frac{\begin{matrix} \alpha_9 \\ \text{(Q3+Q4)} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{Q2 (Q3 + Q4)} \\ \text{U,T,Y,R,Z-Befehl} \end{matrix}}{\text{+}} \left. \frac{\text{Faf}}{\text{E/A-Verriegelung}} \right] +$$

$$+ \text{ Schiftregister in Ø4 bei Eingabe} \quad + \\ + \text{ Ac1} \quad +$$

$$+ \text{ übernehme die Addierlogik} \\ + \beta_1 \left[\frac{\begin{matrix} \text{L I1 I2} \\ \text{s. nächste Gleichung} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{L I1 I2} \\ \text{Addierlogik} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{L I1 I2} \\ \text{+} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{L I1 I2} \\ \text{+} \end{matrix}}{\text{+}} \right]$$

$$\beta_1 = \frac{\begin{matrix} \text{Ø4a keine Vorzeichenzeit} \\ \text{H T3} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{Ø4a gerade WZ} \\ \text{H P1} \end{matrix}}{\text{+}}$$

$$+ \text{ WP 67, D-Befehl} \quad + \quad \text{Ø4 A,S-Befehl} \\ + \frac{\begin{matrix} \text{F G H} \\ \text{Ø4} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{F G Q1 Q2 Q3} \\ \text{A,S-Befehl} \end{matrix}}{\text{+}}$$

$$Cw' = \frac{\begin{matrix} \text{Adressenumlauf für alle Befehle außer U} \\ \text{F S2 C} \\ \text{Ø3,Ø4} \end{matrix}}{\text{Adreßzeit}} \frac{\begin{matrix} \text{brc (Q1+Q2+Q3+Q4)} \\ \text{nicht Fü/Lö kein U-Befehl} \end{matrix}}{\text{+}}$$

$$+ \text{ Adressenumlauf in Ø1, Ø3} \quad + \quad \text{C-Umlauf in Ø4a} \\ + \frac{\begin{matrix} \text{G S2 C} \\ \text{Ø1, Ø3} \end{matrix}}{\text{Adreßzeit}} \frac{\begin{matrix} \text{brc} \\ \text{nicht Fü/Lö} \end{matrix}}{\text{+}} + \frac{\begin{matrix} \text{H C} \\ \text{Ø4a} \end{matrix}}{\text{+}} \frac{\begin{matrix} \text{brc} \\ \text{nicht Fü/Lö} \end{matrix}}{\text{+}}$$

$$+ \text{ C-Umlauf außer Adresse, außer Vorzeichenzeit, außer -Z-Befehl} \\ + \frac{\begin{matrix} \text{G S2 C} \\ \text{Ø2,Ø4} \end{matrix}}{\text{nicht Adreßzeit}} \frac{\begin{matrix} \text{brc (F + T3 + R)} \\ \text{nicht Fü/Lö kein -Befehl} \end{matrix}}{\text{+}}$$

Rechenwerk (80571)

$$+ \underbrace{Q1 + Q2 + Q3 + Q4}_{\text{alle Befehle außer Z}} + \underbrace{Faf}_{\text{E/A-Verriegelung}} + \underbrace{To}_{\text{E/A-Tastegelung}} +$$

$$+ \text{Addiere 1 zur Adresse in C in } \emptyset 2 +$$

$$+ \underbrace{F}_{\emptyset 2} \underbrace{G}_{\text{Adreßzeit}} \underbrace{H}_{\text{Adreßzeit}} \underbrace{S2}_{\text{Adreßzeit}} \underbrace{C}_{\text{Adreßzeit}} \underbrace{K}_{\text{Adreßzeit}} + \underbrace{F}_{\emptyset 2} \underbrace{G}_{\text{Adreßzeit}} \underbrace{H}_{\text{Adreßzeit}} \underbrace{S2}_{\text{Adreßzeit}} \underbrace{C}_{\text{Adreßzeit}} \underbrace{K}_{\text{Adreßzeit}} +$$

Addiere 1

$$+ \text{Vorzeichenumlauf } \emptyset 1, \emptyset 3 + \text{Übernehme die 2te Sektoradresse}$$

$$+ \underbrace{G}_{\emptyset 1, \emptyset 3} \underbrace{T3}_{\text{nicht FÜ/Lö}} \underbrace{C}_{\text{nicht FÜ/Lö}} \underbrace{brc}_{\text{nicht FÜ/Lö}} + \underbrace{G}_{\emptyset 1, \emptyset 3} \underbrace{H}_{\emptyset 1, \emptyset 3} \underbrace{S1}_{\text{2te Sektoradresse}} \underbrace{S2}_{\text{2te Sektoradresse}} \underbrace{S3}_{\text{2te Sektoradresse}} +$$

$$+ \text{Setze das Überlaufbit zur Vorzeichenzeit } \emptyset 4 \text{ A,S-Befehl} +$$

$$+ \underbrace{F}_{\emptyset 4} \underbrace{G}_{\emptyset 4} \underbrace{T3}_{\emptyset 4} \underbrace{Q1}_{\text{A,S-Befehl}} \underbrace{Q2}_{\text{A,S-Befehl}} \underbrace{Q3}_{\text{A,S-Befehl}} (\alpha 5 + \alpha 6 + \alpha 7 + \alpha 8) +$$

normale Addierlogik s. A2-115

$$+ \text{Übernehme R in } \emptyset 4 \text{ beim U-Befehl} +$$

$$+ \underbrace{F}_{\emptyset 4} \underbrace{G}_{\emptyset 4} \underbrace{H}_{\emptyset 4} \underbrace{T3}_{\emptyset 4} \underbrace{Q1}_{\text{U-Befehl}} \underbrace{Q2}_{\text{U-Befehl}} \underbrace{Q3}_{\text{U-Befehl}} \underbrace{Q4}_{\text{U-Befehl}} \underbrace{R}_{\text{U-Befehl}} +$$

$$+ \text{Setze das Überlaufbit zur Vorzeichenzeit } \emptyset 4 \text{ beim D-Befehl in WP 2}$$

$$+ \underbrace{F}_{\emptyset 4a, 2.WP} \underbrace{G}_{\emptyset 4a, 2.WP} \underbrace{H}_{\emptyset 4a, 2.WP} \underbrace{T3}_{\emptyset 4a, 2.WP} \underbrace{Q3}_{\text{Division}} \underbrace{P1}_{\text{gibt Überlauf an}} \underbrace{P6}_{\text{gibt Überlauf an}} \underbrace{A}_{\text{gibt Überlauf an}} +$$

siehe vorhergehende Zeile

$$+ \underbrace{F}_{\emptyset 4a, 2.WP} \underbrace{G}_{\emptyset 4a, 2.WP} \underbrace{H}_{\emptyset 4a, 2.WP} \underbrace{T3}_{\emptyset 4a, 2.WP} \underbrace{Q3}_{\text{Division}} \underbrace{P1}_{\text{gibt Überlauf an}} \underbrace{P6}_{\text{gibt Überlauf an}} \underbrace{A}_{\text{gibt Überlauf an}} +$$

$$I1 = \underbrace{H}_{\text{nicht } \emptyset 4a} \underbrace{A}_{\text{nicht } \emptyset 4a} + \underbrace{D\text{-Befehl}}_{\text{Übernehme in WP 2 den A (Dividend)}}: \underbrace{F}_{\text{WZ 2}} \underbrace{G}_{\text{WZ 2}} \underbrace{P1}_{\text{WZ 2}} \underbrace{Q3}_{\text{D-Befehl}} \underbrace{A}_{\text{D-Befehl}} +$$

$$\text{und Vorzeichen (P6)}$$

$$+ \underbrace{F}_{\text{WZ 2}} \underbrace{G}_{\text{WZ 2}} \underbrace{P1}_{\text{WZ 2}} \underbrace{S1}_{\text{= T3}} \underbrace{S3}_{\text{= T3}} \underbrace{Q3}_{\text{D-Befehl}} \underbrace{P6}_{\text{Vorzeichen}} +$$

keine Sektorübereinstimmung in $\emptyset 1$ und $\emptyset 3$ +

$$+ \begin{array}{cccccccccccc} \underline{G} & \underline{H} & \underline{S3} & \underline{S2} & \underline{S1} & r1 & \underline{Faf} & + & \underline{G} & \underline{H} & \underline{S3} & \underline{S2} & \underline{S1} & r1 & \underline{Faf} & + \\ \emptyset 1, \emptyset 3 & \text{Sektorzeit} & & & & & \text{E/A-Verriegel. } \emptyset 1, \emptyset 3 & \text{Sektorzeit} & & & & \text{E/A-Verriegel.} & & & \text{E/A-Verriegel.} & \\ & & & & & & \text{=C in } \emptyset 1; \text{=R in } \emptyset 3 & & & & & \text{=C in } \emptyset 1; \text{=R in } \emptyset 3 & & & & \\ & & & & & & \text{Sektorvergleich} & & & & & & & & & \end{array}$$

+ 6 Bit Mode bei +P-Befehl +

$$+ \begin{array}{cccccccccccc} \underline{F} & \underline{G} & \underline{T3} & \underline{R} & \underline{Q1} & \underline{Q2} & \underline{Q3} & \underline{Q4} & \underline{Faf} & + \\ \emptyset 3 & & & & \text{Druckbefehl} & & & & & \\ & & & & \text{kein negativer Befehl} & & & & & \end{array}$$

+ 6 Bit Mode bei +I-Befehl

$$+ \begin{array}{cccccccccccc} \underline{F} & \underline{G} & \underline{T3} & \underline{R} & \underline{Q1} & \underline{Q2} & \underline{Q3} & \underline{Q4} & \underline{Faf} \\ \emptyset 3 & & & & \text{I-Befehl} & & & & \text{E/A-Verriegelung} \\ & & & & \text{kein -Befehl} & & & & \end{array}$$

Setzer für K' = siehe A2-109

für Spacerbit beim H,C-Befehl + erlaubt Schreiben des 1. Bits +

$$\underline{L}' = \begin{array}{cccc} \underline{W} & + & \underline{F} & \underline{G} & \underline{Q2} & + \\ \text{Schreiberlaubnis} & & \emptyset 4 & & \text{Y,R-Befehl} & \end{array}$$

+ nicht Vorzeichen und gerade WZ bei M,N,D

$$\begin{array}{ccc} \beta 3 & (\alpha 5 & + & \alpha 7) \\ = \underline{T3} + \underline{H} \underline{Q3} \underline{P1} & = \underline{S} \underline{I1} \underline{I2} \underline{L} & = \underline{S} \underline{I1} \underline{I2} \underline{L} \\ & \text{Addition} & \text{Subtraktion} \end{array}$$

T3-Zeit $\emptyset 1... \emptyset 4$, D-Befehl, ungerade WZ M,N-Befehl +

$$\underline{L}' = \begin{array}{ccc} \beta 3 & + & \\ \underline{T3} (\underline{H} + \underline{Q3} + \underline{P1}) & & \end{array}$$

+ Übertragslogik

$$\begin{array}{ccc} \underline{Q2} (\underline{Q1} + \underline{Q3}) & (\alpha 6 & + & \alpha 8) \\ \text{D,M,N,A,S - Befehl} & = \underline{S} \underline{I1} \underline{I2} \underline{L} & = \underline{S} \underline{I1} \underline{I2} \underline{L} \\ & \text{Subtraktion} & \text{Addition} \end{array}$$

Subtrahiere + bei D-Befehl bei Vorzeichengleichheit

$$\underline{S} = \begin{array}{ccccccc} \underline{H} & \underline{Q4} & + & \underline{H} & \underline{Q3} & \underline{P5} & \underline{P6} & + & \underline{H} & \underline{Q3} & \underline{P5} & \underline{P6} & + \\ \emptyset 4a & & & \emptyset 4a & \text{D-Befehl} & & & & \emptyset 4a & \text{D-Befehl} & & & \\ & & & & \text{Vorzeichengleichheit} & & & & & & & & \end{array}$$

M,N-Befehl WZ 2,3 + D-Befehl WZ 67

$$\begin{array}{ccc} \underline{F} & \underline{H} & \underline{Q3} & + & \underline{F} & \underline{G} & \underline{H} \\ \text{WZ 2,3} & & \text{M,N-Befehl} & & \text{WZ 67} & & \end{array}$$

Rechenwerk (80571)

$$\alpha_5 = \begin{array}{cccc} \underline{S} & I_1 & I_2 & \underline{L} \\ \vdots & & & \vdots \\ \text{A-Befehl} & & \text{Addierer-Eingänge} & \end{array}$$

$$\alpha_6 = \begin{array}{cccc} S & I_1 & \underline{I_2} & L \\ \vdots & & & \vdots \\ \text{S-Befehl} & & \text{Addierer-Eingänge} & \end{array}$$

$$\alpha_7 = \begin{array}{cccc} S & \underline{I_1} & I_2 & \underline{L} \\ \vdots & & & \vdots \\ \text{S-Befehl} & & & \end{array}$$

$$\alpha_8 = \begin{array}{cccc} \underline{S} & \underline{I_1} & \underline{I_2} & L \\ \vdots & & & \vdots \\ \text{A-Befehl} & & & \end{array}$$

$$\beta_3 = \underline{T_3} + H \quad Q_3 \quad \underline{P_1}$$

Schalte die P-Flip Flops als Schiftregister

$$i = ia \underbrace{G \ H \ S2 \ S3}_{\substack{\text{Faf} \\ \emptyset 1, \emptyset 3 \quad \text{Spurzeit}}} + ic \quad \text{s. Phasensteuerung A2-110}$$

$$P1' = \underbrace{G}_{\emptyset 1, \emptyset 3} \quad i \quad \underbrace{r1}_{\emptyset 1=C, \emptyset 3=R} \quad + \quad \underbrace{G \ i \ A}_{\emptyset 4} \quad +$$

$$+ \quad \text{ungerade WZ, M, N, D-Befehl} \quad + \quad \text{Setzer von Eingabegerät}$$

$$+ \quad \underbrace{H}_{\emptyset 4a} \quad \underline{P1} \quad T3 \quad + \quad P1c$$

$$\underline{P1}' = \underbrace{G}_{\emptyset 1, \emptyset 3} \quad i \quad \underbrace{r1}_{\emptyset 1=C, \emptyset 3=R} \quad + \quad \underbrace{G \ i \ A}_{\emptyset 4} \quad +$$

$$+ \quad \text{gerade WZ, M, N, D-Befehl} \quad + \quad \text{Rücksetzer von Eingabegerät} \quad +$$

$$+ \quad \underbrace{H}_{\emptyset 4a} \quad P1 \quad T3 \quad + \quad P1d \quad +$$

$$+ \quad \text{1. WZ, M, N, D-Befehl}$$

$$+ \quad \underbrace{G}_{\emptyset 4} \quad P1 \quad T3 \quad \underline{P1b} \quad \underline{\text{Faf}}$$

$$P2' = \underbrace{i \quad P1}_{\text{Schiftsignal}} \quad + \quad \text{Setzer von Eingabegerät} \quad +$$

$$P2c$$

$$\underline{P2}' = \underbrace{i \quad \underline{P1}}_{\text{Schiftsignal}} \quad + \quad \text{Rücksetzer v. Eingabegerät} \quad +$$

$$P2d$$

$$P3' = \underbrace{i \quad P2}_{\text{Schiftsignal}} \quad + \quad \text{Setzer v. Eingabegerät} \quad +$$

$$P3c$$

$$\underline{P3}' = \underbrace{i \quad \underline{P2}}_{\text{Schiftsignal}} \quad + \quad \text{Rücksetzer v. Eingabegerät} \quad +$$

$$P3d$$

Kommandowerk (80572)

$$P4' = \begin{array}{l} \text{übernehme P3} \quad + \quad \text{Setzer v. Eingabegerät} \\ \text{i P3} \quad + \quad P4c \\ \text{Schiftsignal} \end{array}$$

$$P4' = \begin{array}{l} \text{übernehme P3} \quad + \quad \text{Rücksetzer v. Eingabegerät} \\ \text{i P3} \quad + \quad P4d \\ \text{Schiftsignal} \end{array}$$

$$P5' = \begin{array}{l} \text{übernehme P4} \quad + \quad \text{Setzer v. Eingabegerät} \\ \text{i P4} \quad + \quad P5c \quad + \\ \text{Schiftsignal} \end{array}$$

$$+ \begin{array}{l} \text{Vorzeichen von V bei M,N,D-Befehl} \quad + \quad \text{Setze P5 bei 4 Bit Ausgabe} \\ \text{G H T3 V P5a} \quad + \quad \beta 5 \\ \text{\(\emptyset\}4 \quad \text{Hauptsp.} \quad \text{Faf} \quad \text{(E/A-Verriegelung)} \quad \text{siehe weiter unten} \end{array}$$

$$P5' = \begin{array}{l} \text{übernehme P4} \quad + \quad \text{Rücksetzer v. Eingabegerät} \quad + \\ \text{i P4} \quad + \quad P5d \quad + \\ \text{Schiftsignal} \end{array}$$

$$+ \begin{array}{l} \text{Vorzeichen v. V bei M,N,D-Befehl} \\ \text{G H T3 V P5a} \\ \text{\(\emptyset\}4 \quad \text{Hauptsp.} \quad \text{Faf} \quad \text{(E/A-Verriegelung)} \end{array}$$

$$P6' = \begin{array}{l} \text{übernehme P5} \quad + \quad \text{Setzer v. Eingabegerät} \quad + \\ \text{i P5} \quad + \quad P6c \quad + \\ \text{Schiftsignal} \end{array}$$

$$+ \begin{array}{l} \text{Vorzeichen von A in \(\emptyset\}4 \text{ bei M,N,D-Befehl} \quad + \\ \text{F G H T3 A P6a} \quad + \\ \text{\(\emptyset\}4 \quad \text{Akku} \quad \text{Faf} \quad \text{(E/A-Verriegelung)} \end{array}$$

$$+ \begin{array}{l} \text{Vorzeichen v. A (laufender Rest) D-Befehl} \quad + \quad \text{Multiplizierbit beim N,M-Befehl} \\ \text{H T3 P1 Q3 A} \quad + \quad \text{H T3 P1 Q3 A*} \\ \text{\(\emptyset\}4a \quad \text{D-Befehl} \quad \text{\(\emptyset\}4a \quad \text{M,N-Befehl} \end{array}$$

Kommandowerk (80572)

$P6' =$ übernehme P5 + Rücksetzer v. Eingabegerät +
 i P5 + P6d +
 Schiftsignal

+ Vorzeichen v. A in Ø4 bei M,N,D-Befehl +
 + F G H T3 A P6d +
 Ø4 Akku Faf (E/A-Verriegelung)

+ Vorzeichen v. A (laufender Rest) D-Befehl +
 + H T3 P1 Q3 A +
 Ø4a D-Befehl

+ Multiplizierbit beim M,N-Befehl +
 + H T3 P1 Q3 A* +
 Ø4a M,N-Befehl

+ Rücksetzer bei 4 Bit Ausgabe
 + β 5
 s. weiter unten

$Q1' =$ übernehme R + Ø4 Z-Befehl, PS-Tasten Koinzidenz oder
 α11 R + G H T3 Q1 Q2 Q3 Q4 Faf To
 s. weiter unten Ø4 Z-Befehl E/A nicht gedrückt
 E/A-Verriegelung

- Z-Befehl, Überlaufbit
 (R C + P1 Tb1 + P2 Tb2 + P3 Tb3 + P4 Tb4)
 Überlauf PS-Tasten Koinzidenz
 - Z-Befehl

$Q1' =$ übernehme R + Manuell, Eingabe + Rücksetzer nach Z-Befehl
 α11 R + bq + G H T3 Q1
 s. weiter unten Ø2,Ø4

$Q2' =$ übernehme Q1 + beende Stop +
 α11 Q1 + F G H bs +
 s. weiter unten Ø1 "Start"

+ Setzer für Q2 für alle Befehle außer Z0000, Z0100
 + F G T3 Q1 Q2 (Q1+Q3+Q4+P1+P2+P3+P4+P5+P6)
 Ø4 nicht "Step" alle Befehle außer Z0000, Z0100

Kommandowerk (80572)

$$\underline{X'} = \underset{\substack{F \\ \emptyset 3, \emptyset 4}}{+} + \underset{\substack{G \\ \emptyset 2}}{+} + \underline{T3} + \underset{\substack{Tx \\ \text{Translator-} \\ \text{Kupplung}}}{+} + \underset{\substack{JL33 \\ \text{Translator} \\ \text{Cam}}}{+} + \underline{Q3} + \underset{\substack{\text{Ausgabe-} \\ \text{Indikator}}}{+} + \underline{Ff}$$

Flexowriter
nicht ange-
wählt

Drucke P1-Bit

$$Tp1 = X \quad P1$$

Drucke P2-Bit

$$Tp2 = X \quad P2$$

Drucke P3-Bit

$$Tp3 = X \quad P3$$

Drucke P4-Bit

$$Tp4 = X \quad P4$$

Drucke P5-Bit

$$Tp5 = X \quad P5$$

Drucke P6-Bit

$$Tp6 = X \quad P6$$

Drucke

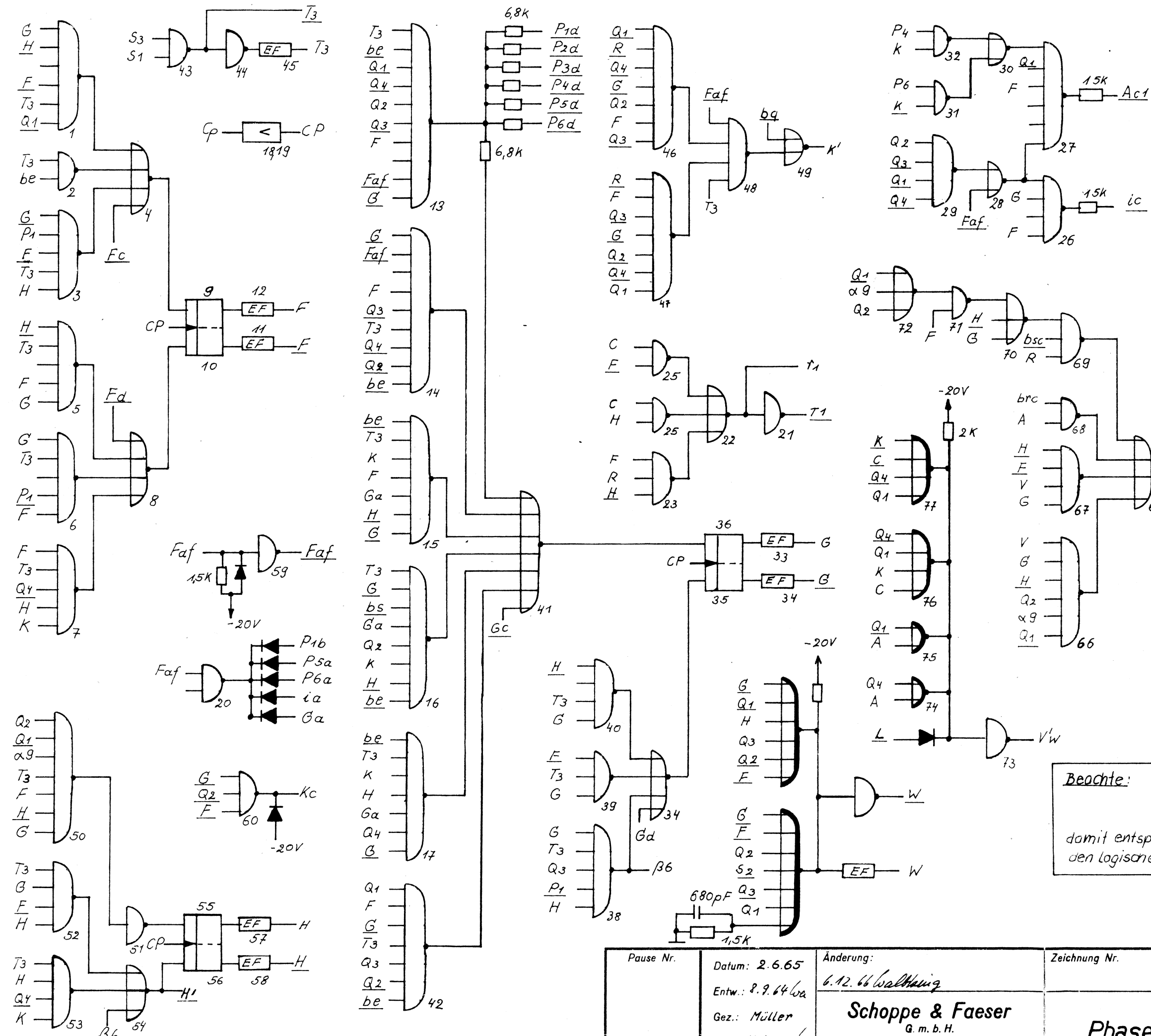
$$Tx = X \quad = \quad JL7$$

Maschinenezeichen

$$Sk = \underline{P5} \quad \underline{P6} \quad K$$

4 Bit Modus

Dieses B... wird bei Änderungen der Uhrzeit... 24. Jan. 1967



Flip-Flop

Inverter

UND

Nor
als

ODER

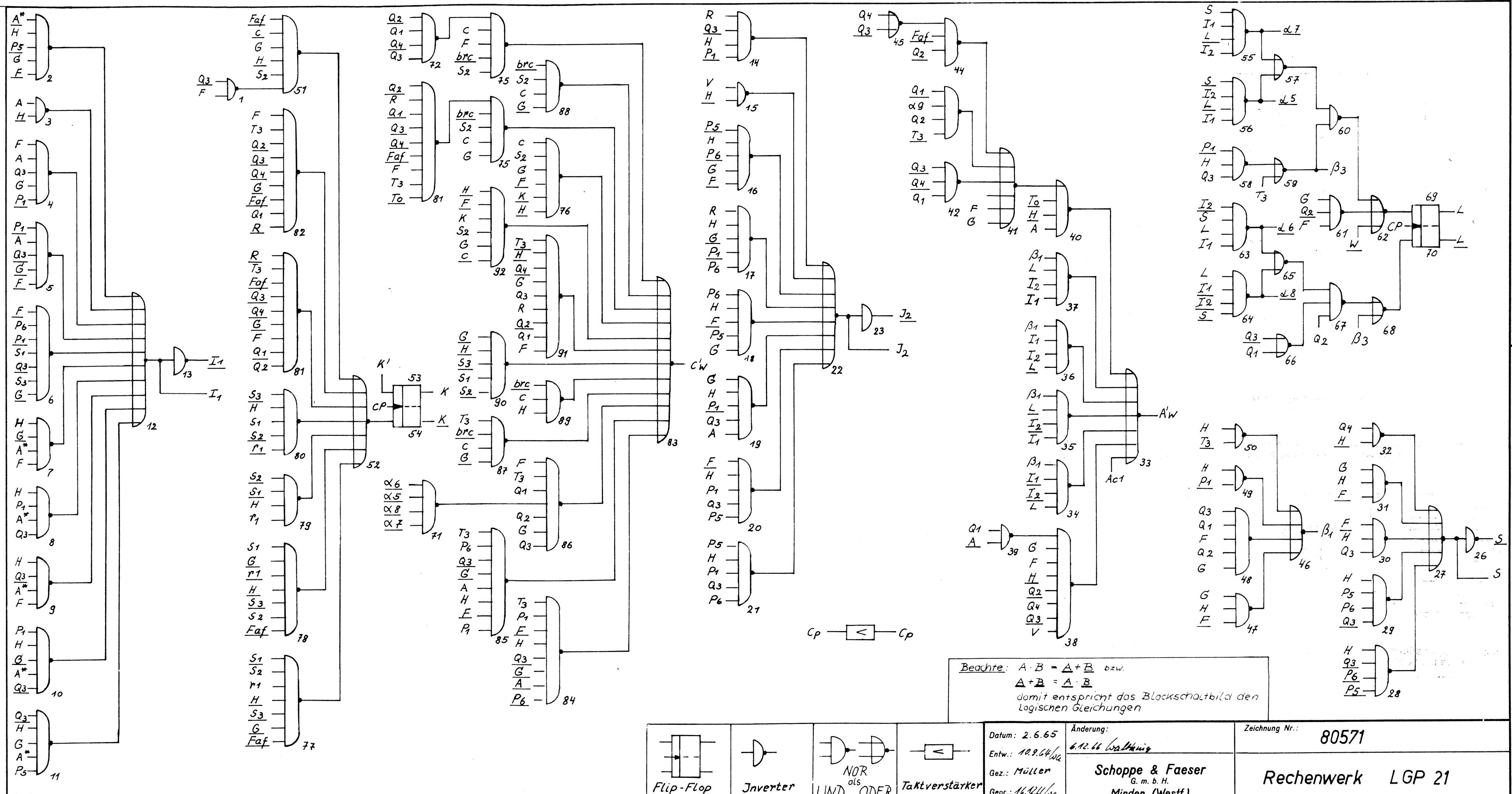
Taktverstärker

Emitterfolger

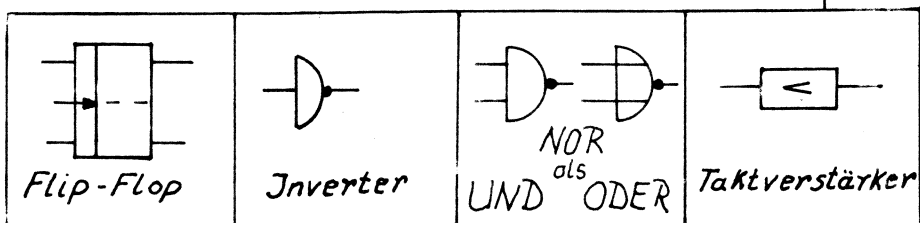
Diodengekopp.
Inverter

Beachte:
 $A \cdot B = \overline{A+B}$ bzw.
 $A+B = \overline{A \cdot B}$
 damit entspricht das Blockschaltbild den logischen Gleichungen

Pause Nr.	Datum: 2.6.65	Änderung: 6.12.66 <i>Walther</i>	Zeichnung Nr. 80570
	Entw.: 8.9.64 <i>wa</i>		
	Gez.: Müller	Schoppe & Faeser G. m. b. H.	Phasensteuerung LGP 21
	Gepr. 16.12.66 <i>wa</i>	Minden (Westf.)	

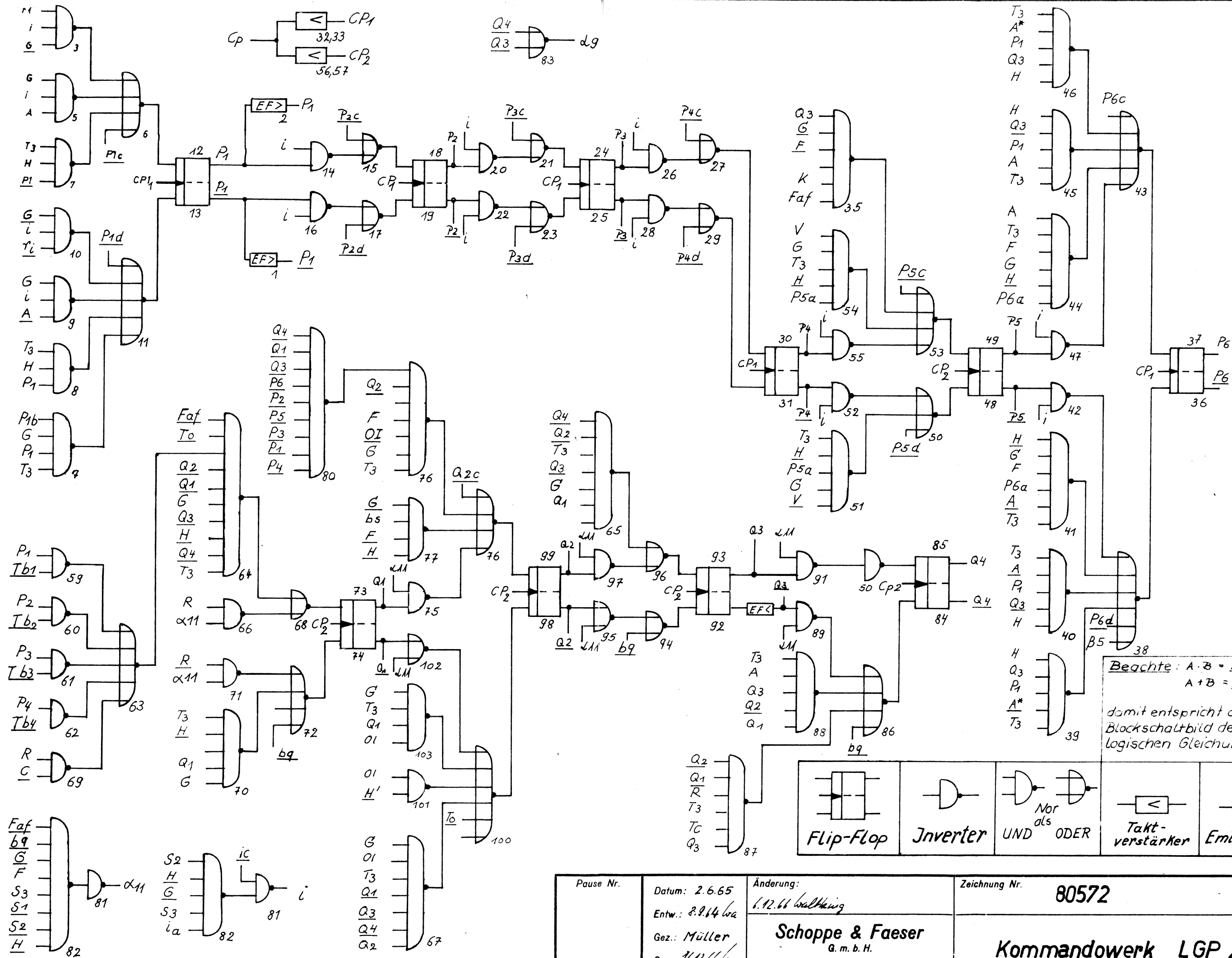


Beachte: $A \cdot B = \overline{A + B}$ bzw.
 $\overline{A + B} = A \cdot B$
damit entspricht das Blockschaubild den
logischen Gleichungen

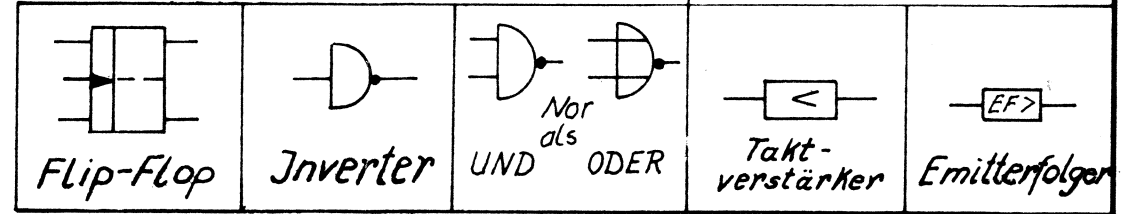


Datum: 2.6.65
Entw.: 10.9.64/wa
Gez.: Müller
Gepr.: 16.12.64/wa
Änderung:
6.12.66 Walthaus
Schoppe & Faeser
G. m. b. H.
Minden (Westf.)

Zeichnung Nr.: **80571**
Rechenwerk LGP 21



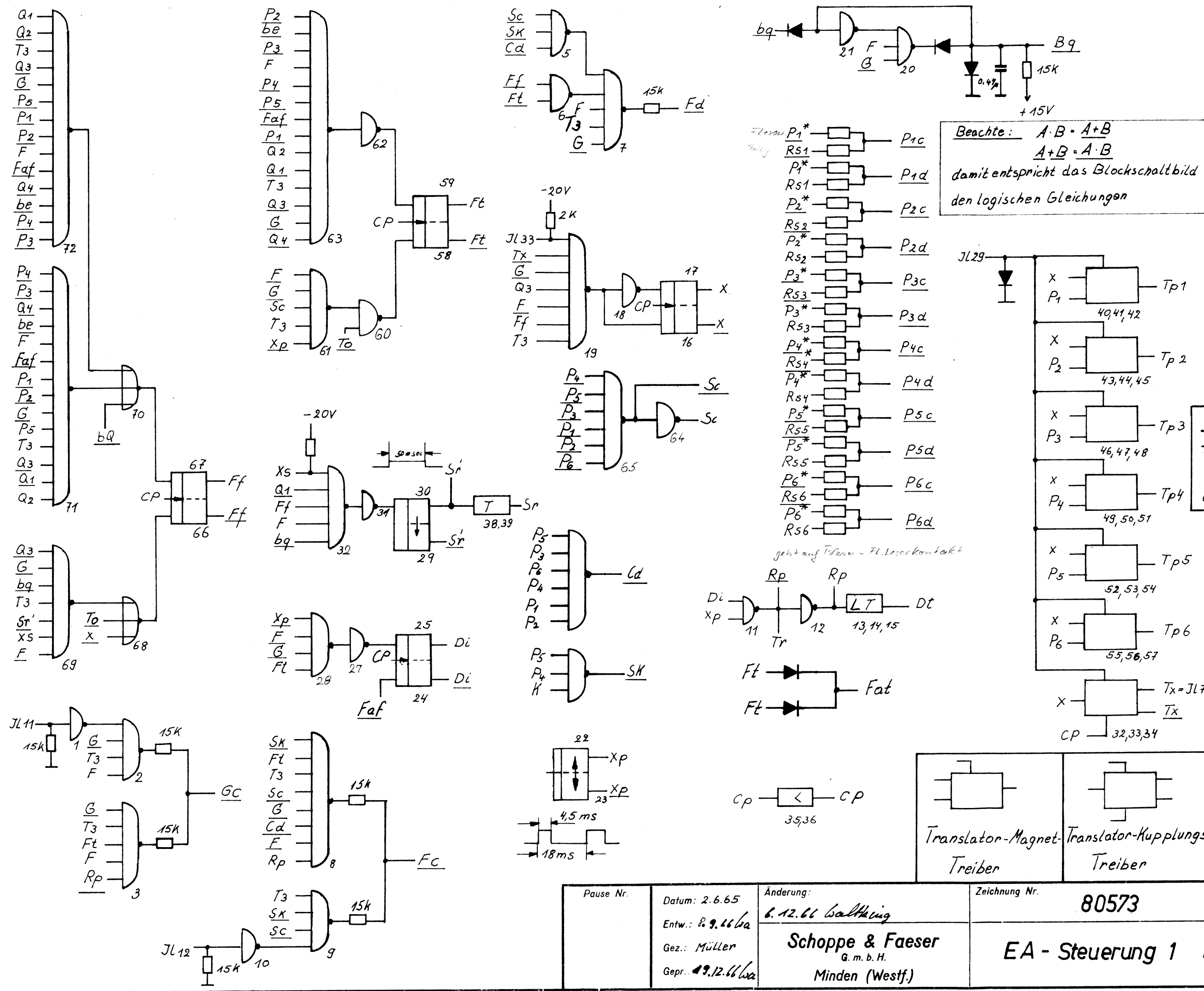
Beachte: $A \cdot B = A + B$
 $A + B = A \cdot B$
damit entspricht das
Blockschaltbild den
logischen Gleichungen



Pause Nr.	Datum: 2.6.65 Entw.: 2.9.64 wa Gez.: Müller Gepr.: 16.12.66 wa	Änderung: 1.12.66 Walkweg	Zeichnung Nr. 80572
Schoppe & Faeser G. m. b. H. Minden (Westf.)			Kommandowerk LGP 21

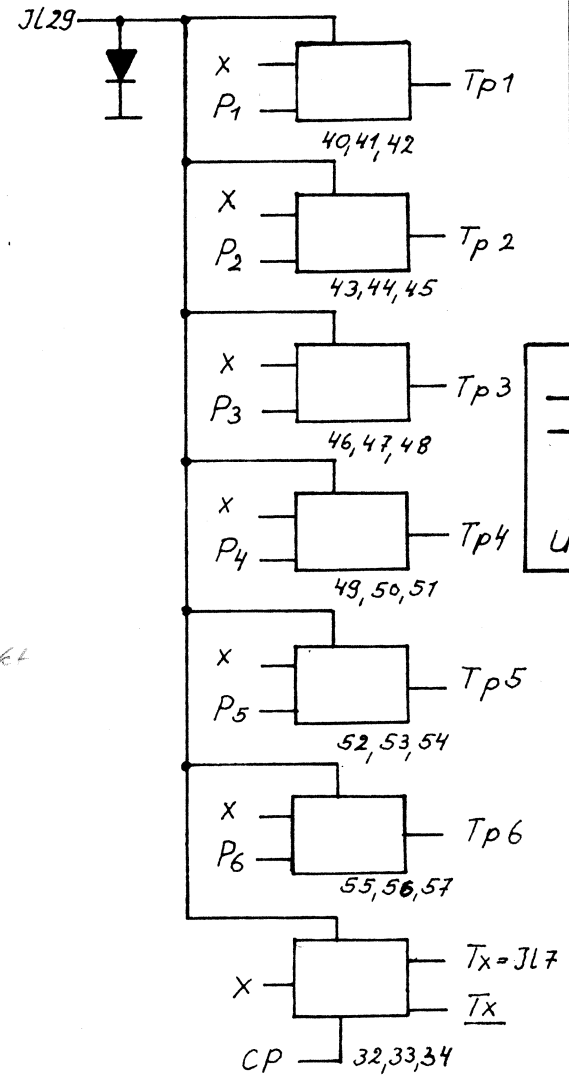
Diese Kopie ist nur für den Gebrauch
 im Rahmen der Auftragsbearbeitung
 und darf nicht weiterverbreitet werden.
 24. Jan. 1967

Diese Bauelemente sind bei Änderungen
 der Grundschaltung nicht erfaßt
 Angelegt am: 24. Jan. 1967

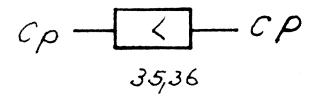
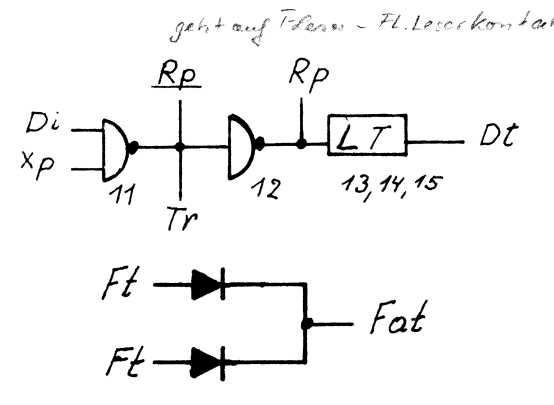
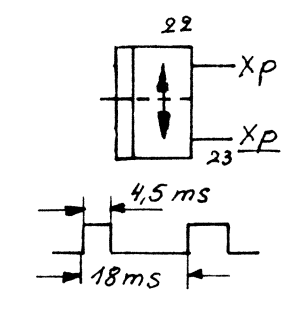
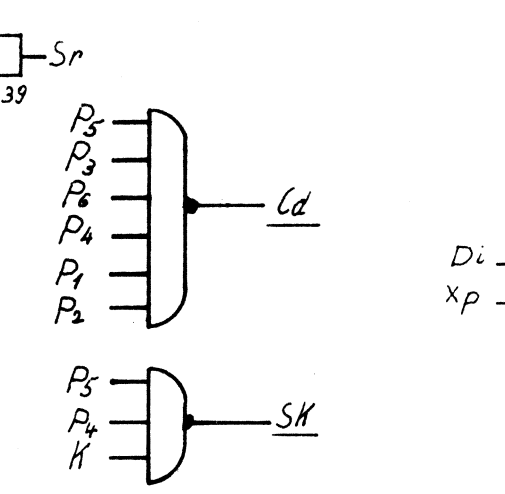
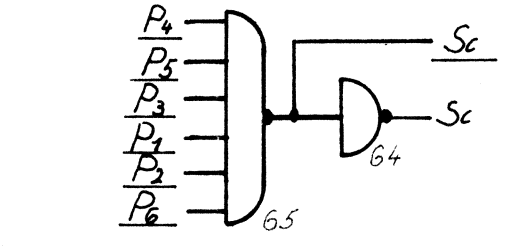
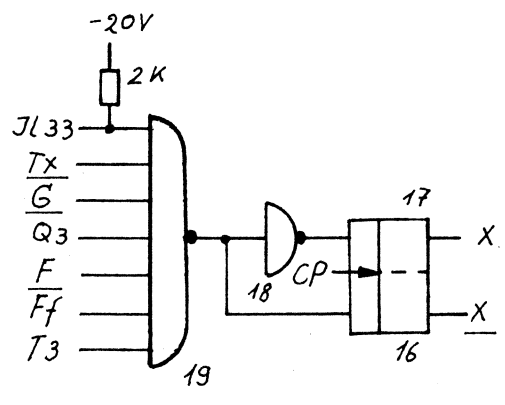
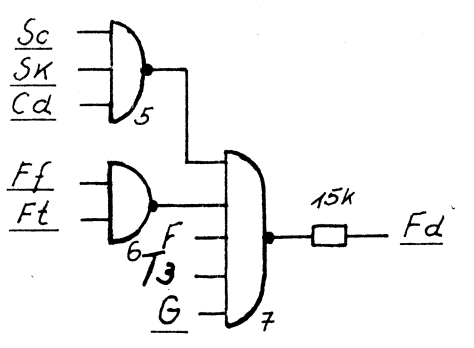


Beachte: $A \cdot B = \overline{A+B}$
 $A+B = \overline{A \cdot B}$
 damit entspricht das Blockschaltbild
 den logischen Gleichungen

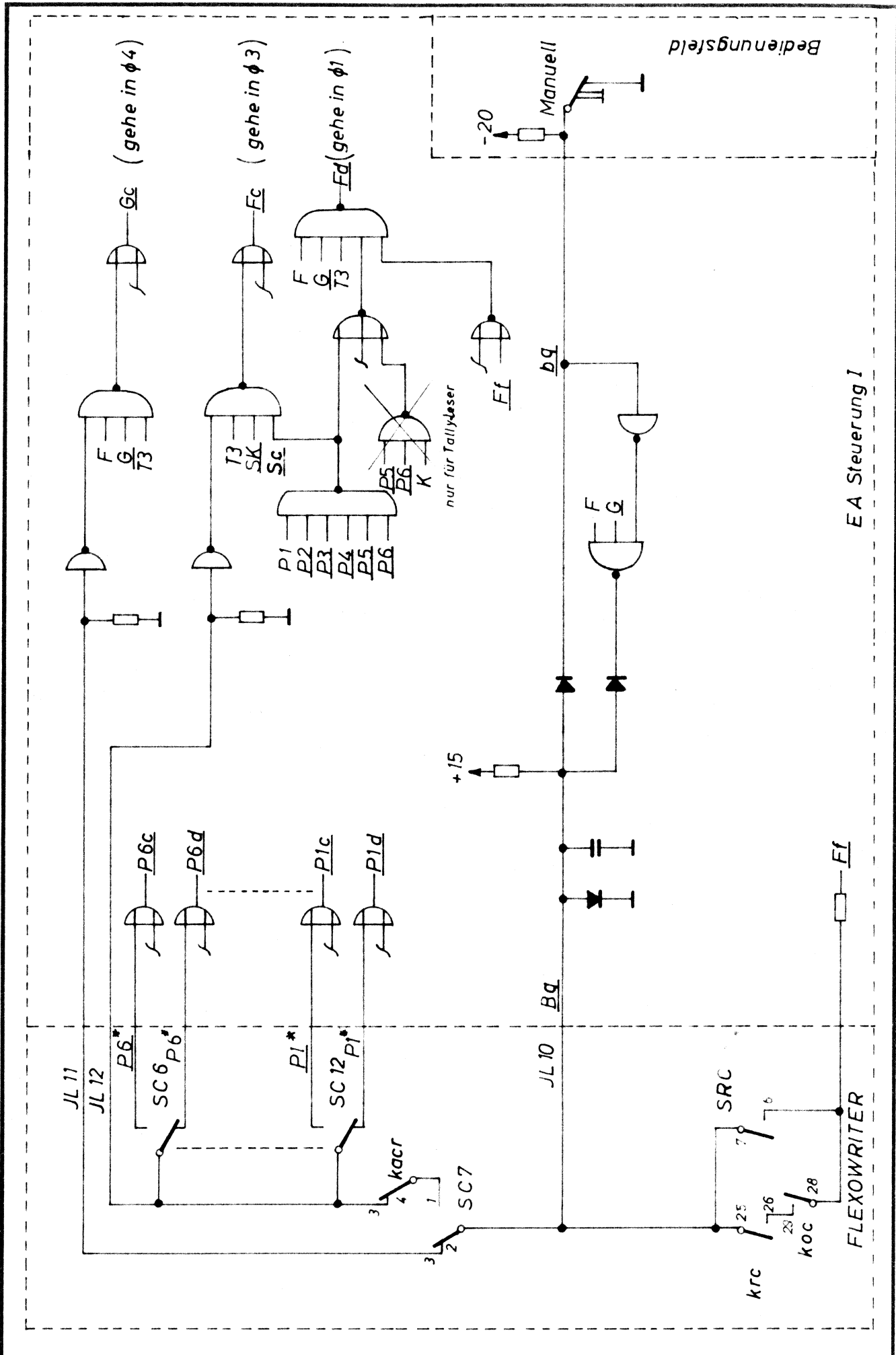
- Flexion Tally
- P_1^* \rightarrow P_{1c}
 - RS_1 \rightarrow P_{1d}
 - P_1^* \rightarrow P_{2c}
 - RS_2 \rightarrow P_{2d}
 - P_2^* \rightarrow P_{3c}
 - RS_3 \rightarrow P_{3d}
 - P_3^* \rightarrow P_{4c}
 - RS_4 \rightarrow P_{4d}
 - P_4^* \rightarrow P_{5c}
 - RS_5 \rightarrow P_{5d}
 - P_5^* \rightarrow P_{6c}
 - RS_6 \rightarrow P_{6d}



	Flip-Flop
	astabiler Multivibrat.
	monostabiler Multivibrat.
	Inverter
	NOR als UND ODER
	Taktverst.
	Leistungstreiber
	Translator-Magnet-Treiber
	Translator-Kupplungs-Treiber



Pause Nr.	Datum: 2.6.65	Änderung:	Zeichnung Nr.
	Entw.: R. 9. 66/6a	6.12.66 Walkthling	80573
	Gez.: Müller	Schoppe & Faeser	EA - Steuerung 1 LGP 21
	Gepr.: 49.12.66/6a	G. m. b. H.	
		Minden (Westf.)	



EA Steuerung I

P. 12. 1966
wa

<p>Schoppe & Faeser G. m. b. H. Minden (Westf.)</p>	<p>Eingabe über Flexowriter Blockschaltbild</p>	<p>Blattzahl: 2 Blatt: 1 A2 - 131</p>
--	---	---

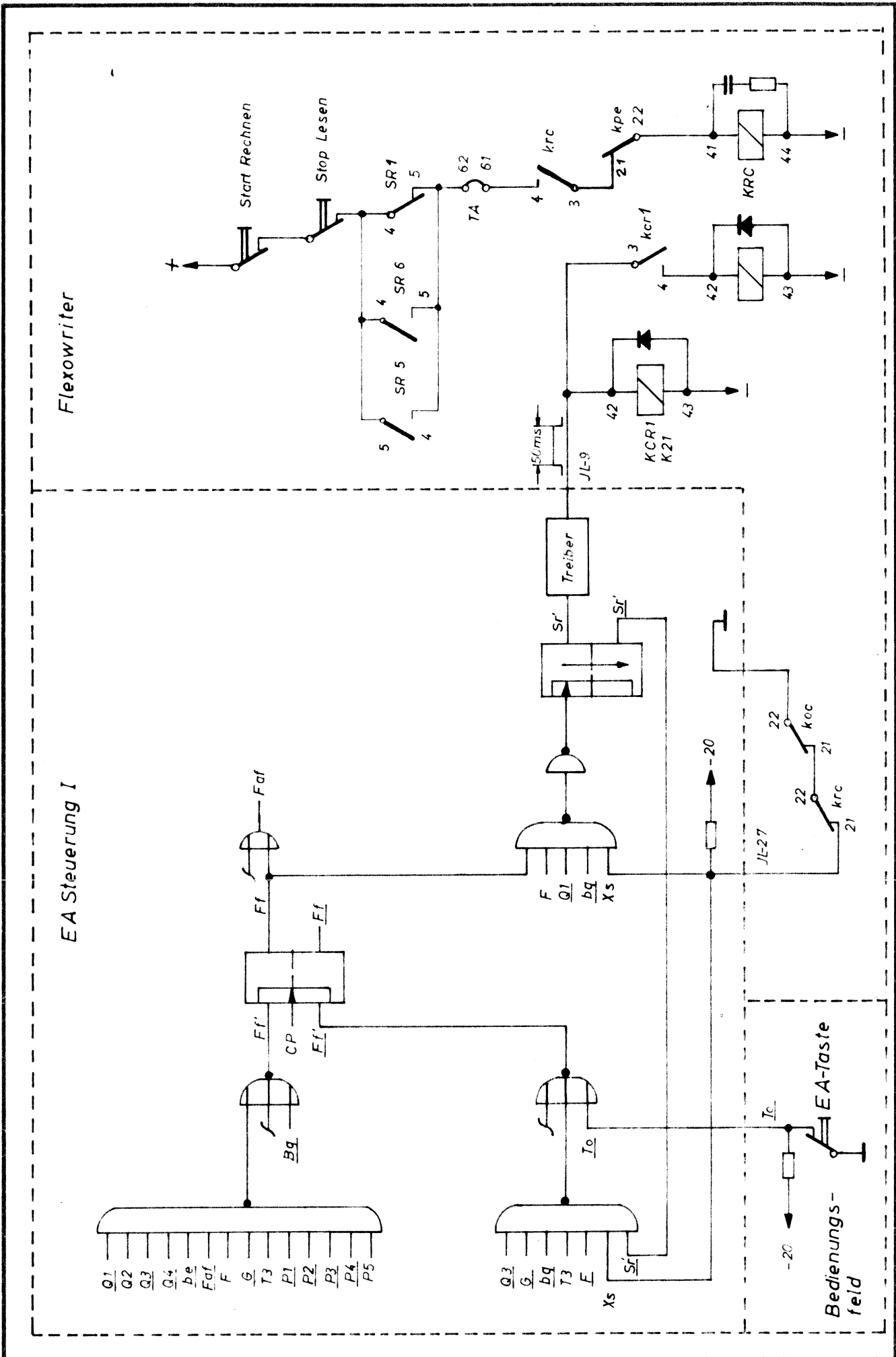
8.12.1966
wa

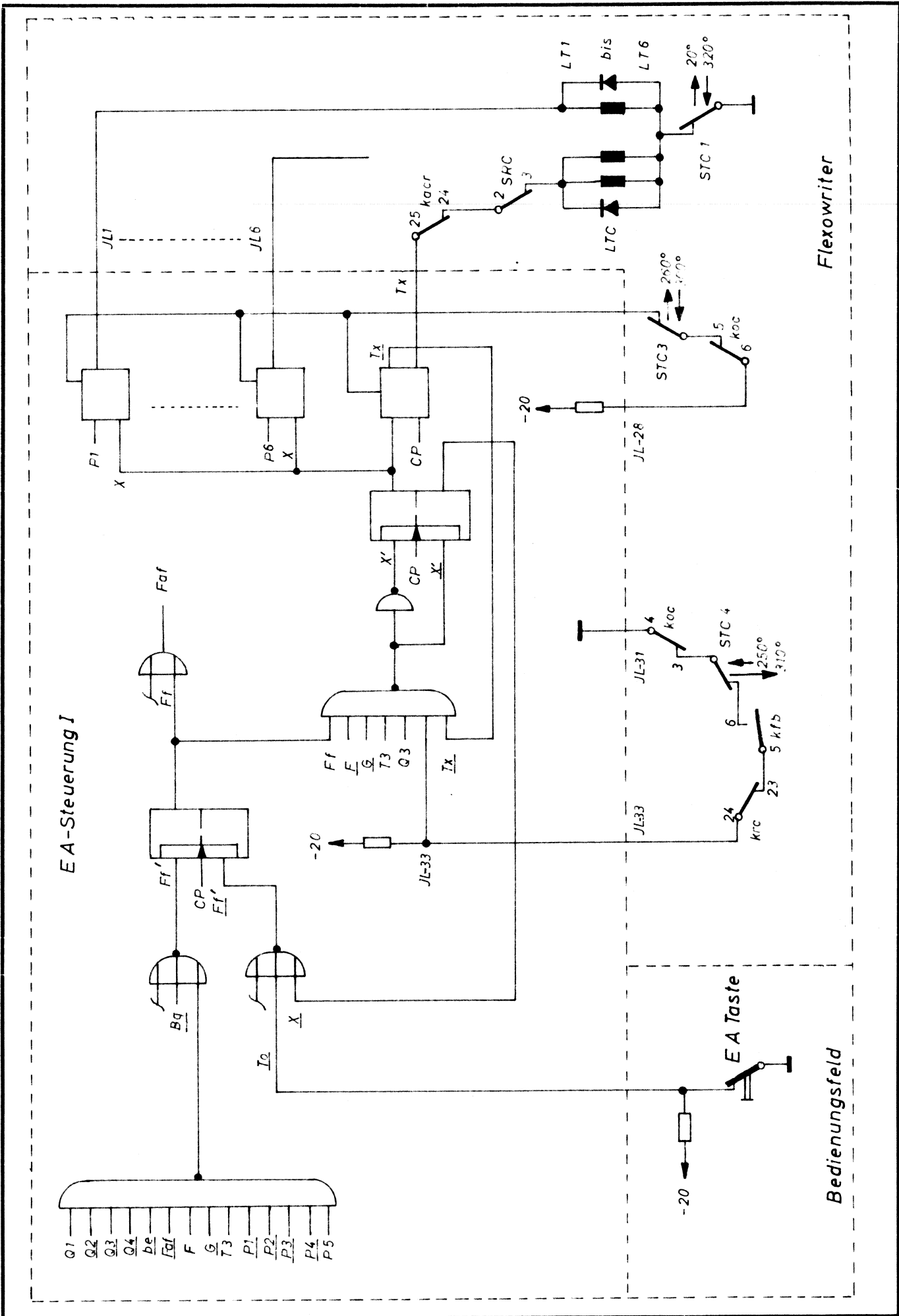
Schoppe & Faeser
G. m. b. H.
Minden (Westf.)

Eingabe über Flexowriter
Blockschaltbild

Blattzahl: 2
Blatt: 2

A2 - 133





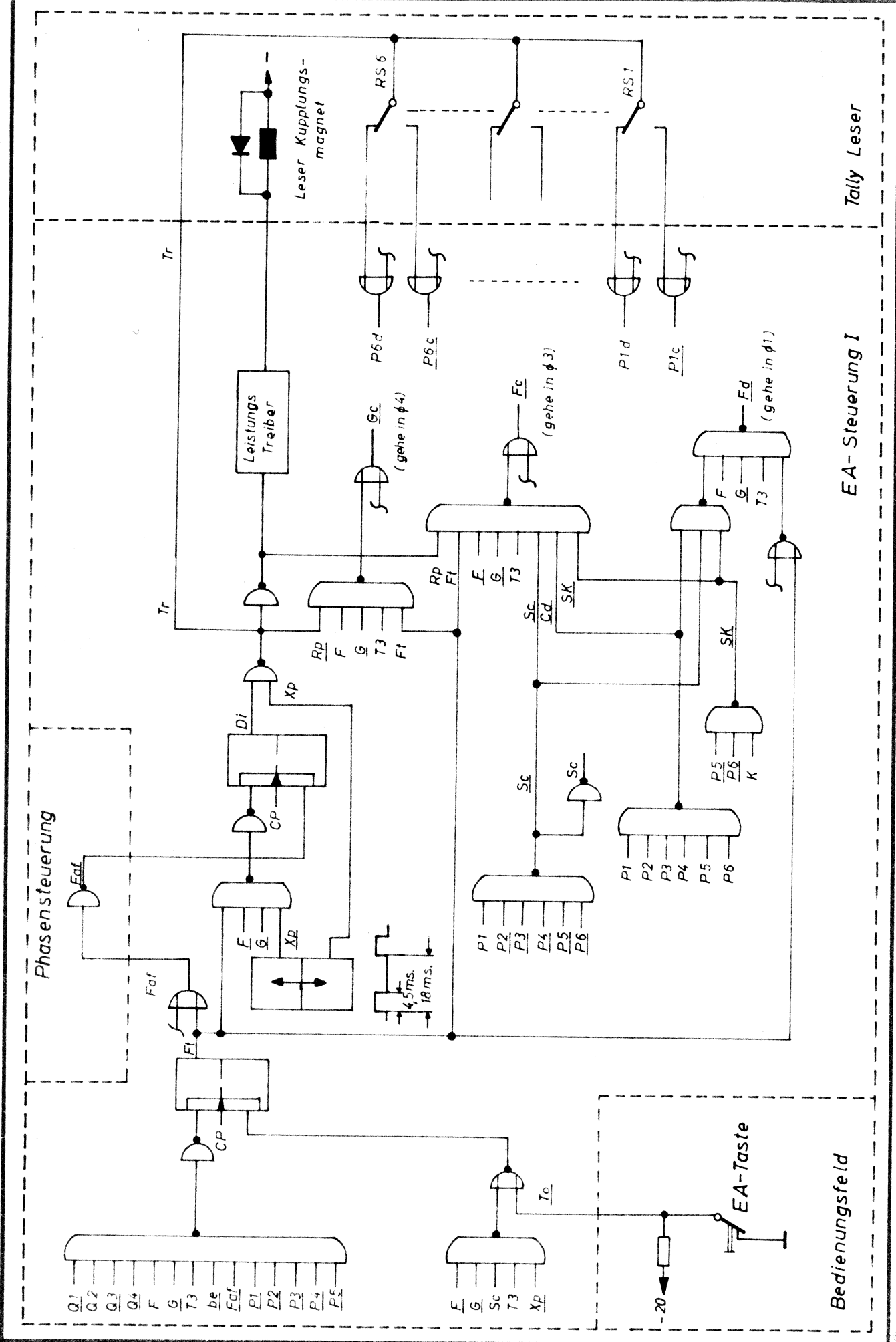
P.12.1966
wa

Schoppe & Faeser
 G. m. b. H.
 Minden (Westf.)

Ausgabe über Flexowriter
 Blockschaltbild

A2-135

P. 12.1966
wa



Schoppe & Faeser
G. m. b. H.
Minden (Westf.)

Eingabe über Tally Leser
Blockschaltbild

A2 - 137