

# Über die Programmierung digitaler Rechenanlagen

Autor(en): **Mühlhahn, A.**

Objekttyp: **Article**

Zeitschrift: **Bulletin des Schweizerischen Elektrotechnischen Vereins :  
gemeinsames Publikationsorgan des Schweizerischen  
Elektrotechnischen Vereins (SEV) und des Verbandes  
Schweizerischer Elektrizitätswerke (VSE)**

Band (Jahr): **55 (1964)**

Heft 24

PDF erstellt am: **21.07.2024**

Persistenter Link: <https://doi.org/10.5169/seals-916793>

## **Nutzungsbedingungen**

Die ETH-Bibliothek ist Anbieterin der digitalisierten Zeitschriften. Sie besitzt keine Urheberrechte an den Inhalten der Zeitschriften. Die Rechte liegen in der Regel bei den Herausgebern. Die auf der Plattform e-periodica veröffentlichten Dokumente stehen für nicht-kommerzielle Zwecke in Lehre und Forschung sowie für die private Nutzung frei zur Verfügung. Einzelne Dateien oder Ausdrucke aus diesem Angebot können zusammen mit diesen Nutzungsbedingungen und den korrekten Herkunftsbezeichnungen weitergegeben werden. Das Veröffentlichen von Bildern in Print- und Online-Publikationen ist nur mit vorheriger Genehmigung der Rechteinhaber erlaubt. Die systematische Speicherung von Teilen des elektronischen Angebots auf anderen Servern bedarf ebenfalls des schriftlichen Einverständnisses der Rechteinhaber.

## **Haftungsausschluss**

Alle Angaben erfolgen ohne Gewähr für Vollständigkeit oder Richtigkeit. Es wird keine Haftung übernommen für Schäden durch die Verwendung von Informationen aus diesem Online-Angebot oder durch das Fehlen von Informationen. Dies gilt auch für Inhalte Dritter, die über dieses Angebot zugänglich sind.

Zum Schlusse dieses Abschnittes sei nochmals betont, dass die Fehlerangaben auf den Grundannahmen von 0,3 m Empfindlichkeit und Ablesegenauigkeit der Druckdifferenzen und auf 10% möglichem Fehler des  $\alpha_m$ -Wertes beruhen.

### 9. Grenzen der Anwendung der thermodynamischen Methode

Die Anwendungsgrenzen sind abhängig von der gewünschten Genauigkeit und von der Festigkeitsgrenze der unter Druck durchflossenen Apparateile.

Setzt man voraus, dass die Apparaturen festigkeitstechnisch den Grenzdrücken der Fehlerfelder gewachsen sind, so können die Anwendungsgrenzen etwa wie folgt festgelegt werden, wobei man sich gemäss Fig. 10 stets auf der sicheren Seite befindet:

Wahrscheinlicher $\eta$ -Fehler	< 1%	< 2%
Peltonturbinen	100...1800 m bis 15°C	80...>2000 m bis 30°C
Francisturbinen	-N 100...1800 m bis 10°C	100...>2000 m bis 28°C
	-HE 150...1200 m bis 10°C	100...1800 m bis 28°C
Speicherpumpen	-N 150...1200 m bis 8°C	120...2000 m bis 22°C
	-HE 300...700 m bis 8°C	150...1300 m bis 22°C
Maximaler Temperaturgradient	$\approx 0,5^\circ\text{C/h}$	$\approx 1,0^\circ\text{C/h}$
Maximale Temperaturdifferenz	$\theta_u - \theta_w, \approx 15^\circ\text{C}$	$\approx 30^\circ\text{C}$

Voraussetzung für die angegebenen Grenzen ist die Erfüllung der Voraussetzungen des Abschnittes 7, wobei besonders an die Bedingung D2 erinnert sei.

### 10. Berechnung der Durchflussmenge

Die Durchflussmenge ergibt sich bei der thermodynamischen Methode dann, wenn zusätzlich eine Leistungsmessung durchgeführt wird.

Mit  $P_T$  Turbinenleistung [kW] und  $P_W$  Wellenantriebsleistung von Pumpen [kW] folgt:

$$102 P_T = \eta_T \gamma Q_T H_n$$

$$Q_T = \frac{102 P_T}{\eta_T \gamma H_n} \quad [\text{m}^3/\text{s}] \quad (32a)$$

und:

$$102 P_W \eta_P = \gamma Q_f H_f$$

$$Q_f = \frac{102 P_W \eta_P}{\gamma H_f} \quad (32b)$$

Die Genauigkeit der Wassermengenresultate ergibt sich aus dem Fehler  $\varphi_P$  der Leistungsmessung, dem Fehler  $\varphi_{\Delta H}$  des Nettogefälles (bzw. der Förderhöhe) gemäss Gl. (30c), und dem  $\eta$ -Fehler etwa gemäss Fig. 10 zu:

$$\varphi_Q \approx \pm \sqrt{\varphi_P^2 + \varphi_{\Delta H}^2 + \varphi_{tot}^2} \quad (32c)$$

Beispiel:

Elektrische Messinstrumente, Klasse 0,2	$\varphi_P \approx 0,2\%$
Ablesung	$\varphi_A \approx 0,2\%$
Verluste	$\varphi_{\Delta} \approx 0,1\%$
Fehler der Leistungsmessung	$\varphi_P \approx 0,3\%$
Fehler im Nettogefälle	$\varphi_{\Delta H} \approx 0,3\%$
Fehler in $\eta_{tot}$ (thermodynamische Messung)	$\varphi_{tot} \approx 1,1\%$
Dies ergibt $\varphi_Q \approx \pm \sqrt{0,3^2 + 0,3^2 + 1,1}$	$\varphi_Q \approx \pm 1,2\%$

### Literatur

- [1] Barbillon, L. und A. Poirson: Note sur un nouveau procédé de mesure du rendement des turbines hydrauliques (Méthode thermométrique). Bull. Rech. sci., industr. et Inventions 2(1921)23.
- [2] Willm, G. und P. Campmas: Mesure du rendement des turbines hydrauliques par la méthode thermométrique Poirson. Houille blanche 9(1954)4.
- [3] Willm, G.: La précision de la méthode thermodynamique à la lumière de deux années d'application industrielle. Houille blanche 11(1956)4.
- [4] Chappuis, J.: Première application en Suisse de la méthode thermodynamique à la mesure du rendement de turbines hydrauliques. Bull. Techn. de la Suisse Romande 82(1956)11.
- [5] Vaucher, R.: Thermodynamische Wirkungsgradmessungen an hydraulischen Maschinen. Escher Wyss Mitt. 32(1959)2/3.
- [6] Brand, F.L.: Die Messung des Wirkungsgrades von hydraulischen Maschinen nach dem thermodynamischen Verfahren. Voith Forsch., Konstr. -(1961)7. (Enthält komplettes Literaturverzeichnis bis 1960.)
- [7] Vaucher, R.: Rückblick auf eine vierjährige Anwendung der thermodynamischen Methode bei Wirkungsgradmessungen hydraulischer Maschinen. Bulletin SEV 54(1963)6.
- [8] Wukalowitsch, M.P.: Thermodynamische Eigenschaften des Wassers und des Wasserdampfes. VEB Verlag Technik, Berlin, 1958.
- [9] Schmidt, E.: VDI-Wasserdampf tafeln, Springer, Berlin, 1963.

### Adresse des Autors:

P. U. Weber, dipl. Pkysiker, Institut für hydraulische Maschinen an der ETH, Sonneggstrasse 3, 8006 Zürich.

## Über die Programmierung digitaler Rechenanlagen

Von A. Mühlhahn, Konstanz

681.14 - 523.8 : 518.5

### 1. Grundsätzliches

Die digitalen Rechenanlagen sind in der heutigen Zeit zu einem weitverbreiteten und unentbehrlichen menschlichen Hilfsmittel geworden. Man findet die Rechner in Universitäten, Banken, Finanzämtern, Forschungszentren, in der Flugsicherung und bei der Prozeßsteuerung in Fabriken. In den nächsten Jahren wird die Anzahl der eingesetzten Rechner noch erheblich zunehmen. Sie sind im gewissen Sinn zu einem Sinnbild der modernen Welt geworden, und viele technische Entwicklungen, wie z. B. die Raketentechnik, sind ohne den Einsatz elektronischer Rechenanlagen nicht denkbar. Diese zweifellos imponierenden Leistungen der Anlagen verführen jedoch dazu, dass häufig die Möglichkeiten der Rechner überschätzt werden.

Eine elektronische Rechenanlage ist kein Wunderwesen, das auf jede beliebig gestellte Frage eine Antwort bereit hat. Stelle man sich vor, es sollte eine Differentialgleichung gelöst werden. Man wird in den meisten Fällen die Lösung durch ein Näherungsverfahren berechnen. Soll diese Arbeit von einer ungeschulten Kraft ausgeführt werden, so wird man dieser die Formeln und die dazugehörigen Konstanten übergeben mit einer Anweisung, wie gerechnet werden soll. Diese Anweisung muss um so genauer und ausführlicher ausfallen, je weniger die Hilfskraft von der Materie versteht. Man kann aber die Berechnung auch von einer elektronischen Rechenanlage durchführen lassen. Diese hat jedoch gar kein Verständnis von den zu bearbeitenden Problemen. Infolgedessen muss der Programmierer mühsam einen Plan

erstellen, der der Anlage bis in die kleinsten Einzelheiten vorschreibt, wie sie zu arbeiten hat. Insbesondere muss dem Rechner für jeden der möglichen Fälle genau mitgeteilt werden, wie er sich verhalten soll. Jeder dieser Fälle muss vom Programmierer im voraus bedacht und durchdacht werden. Der Rechner führt letzten Endes nur Stück für Stück das aus, was ihm durch das Programm vorgeschrieben wird, er ist also völlig unselbständig, und man muss ihm jegliche eigene Intelligenz absprechen. Vor allem ist es unsinnig, von einem Elektronengehirn zu sprechen, da man von einem Gehirn doch im allgemeinen eine gewisse Selbständigkeit erwarten darf. Ohne ein Programm ist ein Rechner ein totes Werkzeug aus Drähten, Widerständen und Transistoren.

Dafür ist aber der Rechner in der Lage, mit einer unvorstellbaren Geschwindigkeit zu arbeiten. Zwar kann eine Rechenanlage nichts tun, was jeder Benutzer dieses Rechners nicht auch selbst tun könnte, sie kann es jedoch unvergleichlich schneller, und auf dieser Geschwindigkeit beruht die grosse Bedeutung der Anlagen. Als Beispiel, bei dem es auf die Geschwindigkeit des Rechners besonders ankommt, sei der Abschuss von Weltraumraketen genannt. Nach dem Start einer Rakete berechnet die Anlage aus den eintreffenden Positionsmeldungen die Abweichungen vom geforderten Kurs, errechnet die nötigen Kursänderungen und korrigiert fortlaufend deren Bahn. Eine Berechnung auf konventionelle Art wäre in diesem Fall unmöglich, da sich die Rakete schon zu weit von der vorberechneten Bahn entfernt hätte, bevor überhaupt die Rechnung richtig beginnen würde.

## 2. Die Arbeitsweise

Zur Mitteilung an den Rechner steht eine Befehlsmenge zur Verfügung, die je nach Typ und Grösse der Anlage etwa 16...250 Befehle umfassen kann. Ein Befehl ist dabei eine exakt definierte Anweisung an den Rechner, die einen bestimmten Vorgang (Operation), z. B. die Addition von zwei Zahlen, auslöst. Ein Programm besteht aus einer Folge von Befehlen, und programmieren heisst, die Lösung eines Problems in eine dem Rechner verständliche Form, d. h. in eine Befehlsfolge zu übertragen.

Um an späterer Stelle ein kleines Programm vorführen zu können, ist es nötig, in grossen Zügen die Arbeitsweise eines Rechners zu beschreiben.

Ein Rechner besteht im wesentlichen aus Befehlswerk, Rechenwerk, Ein- und Ausgabewerk und Speicher. Ein Programm wird im allgemeinen schriftlich fixiert. Es ist nun aber bei den grossen Rechenanlagen nicht so, dass sich der Programmierer an den Rechner stellt und mittels irgendwelcher Hebel oder Schalter dem Rechner Befehl für Befehl mitteilt, was dieser als nächstes tun soll. Die Programme mit den zugehörigen Konstanten werden vielmehr mit Lochstreifen, Lochkarten oder Magnetband in den Speicher gelesen, und dem Rechner wird dann mitgeteilt, wo er mit der Verarbeitung beginnen soll.

Es gibt verschiedene Arten von Speicher (Trommelspeicher, Kernspeicher), in denen die Information in Form von elektrischen oder magnetischen Zuständen aufgezeichnet ist. Jede Speicherstelle ist jeweils zweier Zustände fähig, so dass die Information binärverschlüsselt im Speicher steht. Eine Anzahl Speicherstellen stellt ein Zeichen dar. Mehrere Zeichen sind zu einer Einheit, die man «Wort» nennt, zusammengefasst, von denen jede eine Adresse — eine Art Haus-

nummer — hat. Wir wollen der Einfachheit halber annehmen, dass jedes Wort einen Befehl oder eine Zahl aufnehmen kann. Es genügt nun, dem Rechner die Anfangsadresse des Programms mitzuteilen. Die Anlage überträgt dann einen Befehl nach dem anderen aus dem Speicher in das Befehlswerk, entschlüsselt den Befehl und führt ihn aus. Im allgemeinen werden die Befehle in der Reihenfolge verarbeitet, in der sie im Speicher stehen. Eine Ausnahme von dieser Regel ermöglichen die sog. Sprungbefehle. Ein Sprungbefehl teilt dem Rechner mit, dass er als nächstes nicht den folgenden Befehl ausführen, sondern an einer Stelle weiterarbeiten soll, die der Sprungbefehl angibt. Die Ausführung eines Sprungbefehls wiederum kann von bestimmten Bedingungen abhängig gemacht werden (bedingte Sprungbefehle), z. B. davon, ob die sich im Rechenwerk befindende Zahl gleich 0 ist.

Das Rechenwerk dient zur Ausführung von Rechenoperationen, zu denen vor allem die 4 Grundrechenarten gehören.

## 3. Der Befehlscode

Ein Befehl besteht im wesentlichen aus dem Operationsteil, der die auszuführende Operation, und dem Adressteil, der die Adresse des Operanden angibt. (Man bezeichnet diesen Typ eines Rechners als Einadressmaschine.) Der Befehl «ADDIERE 100» bedeutet z. B., dass der Rechner zu der sich im Rechenwerk befindenden Zahl den Inhalt der Speicherzelle mit der Adresse 100 addieren soll. Der Operationsteil wird intern auch binärverschlüsselt dargestellt.

Es sollen nun einige Befehle näher besprochen werden (die Klammer enthält die jeweilige Abkürzung, unter der der Befehl in dem Programm auftaucht):

### a) Arithmetische Befehle:

Addition (ADD)  
Subtraktion (SUB)  
Division (DIV)  
Multiplikation (MUL)  
Quadratwurzel (W)

Die arithmetischen Befehle dienen zur Ausführung arithmetischer Operationen. Der erste Operand befindet sich jeweils im Rechenwerk, der Adressteil des Befehls gibt an, wo der zweite Operand steht. Eine Ausnahme bildet der Befehl W, er besitzt nur einen Operanden, der sich in der durch den Adressteil bezeichneten Speicherzelle befinden muss. Das Ergebnis aller arithmetischen Operationen steht im Rechenwerk.

### b) Transportbefehle:

Die Transportbefehle dienen zum Transport der Information aus dem Speicher in das Rechenwerk (B) und umgekehrt (C). Im ersten Fall gibt die Adresse die Speicherzelle an, deren Inhalt in das Rechenwerk gebracht werden soll, und im zweiten Fall, in welche Speicherzelle der Inhalt des Rechenwerkes gebracht werden soll. Die vorherigen Inhalte werden dabei überschrieben.

### c) Bedingte Sprungbefehle:

Der Adressteil enthält die Adresse, auf die gesprungen werden soll. Die Ausführung des Sprunges hängt von Bedingungen ab, und zwar können folgende Aussagen als Bedingungen auftreten:

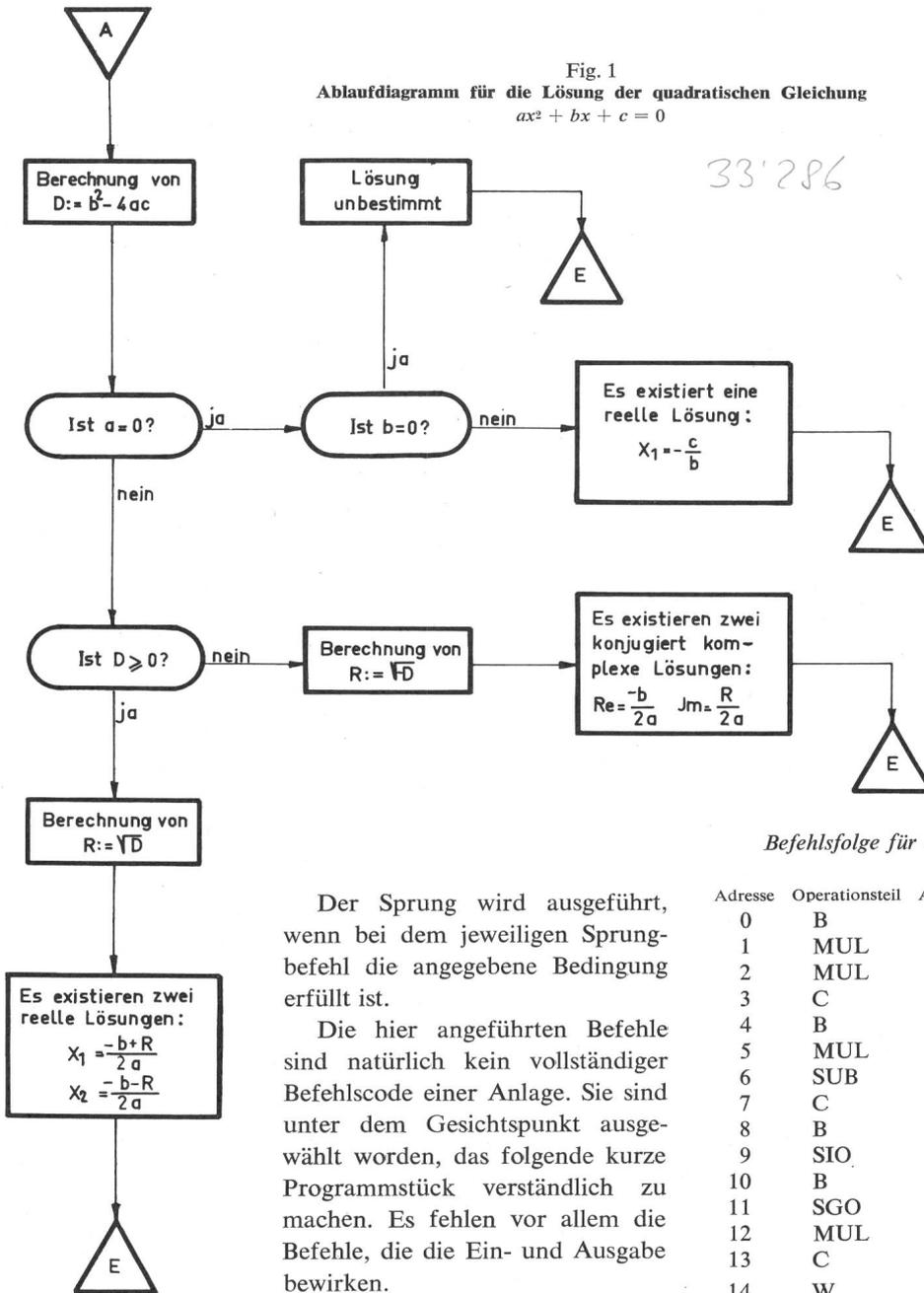
1. Die Zahl im Rechenwerk ist identisch 0 (SIO),
2. Die Zahl im Rechenwerk ist verschieden von 0 (SNO),
3. Die Zahl im Rechenwerk ist grösser oder gleich 0 (SGO),
4. Die Zahl im Rechenwerk ist kleiner oder gleich 0 (SKO).

dem codierten Programm durch Sprungbefehle realisiert werden. Das Diagramm beginnt bei A und endet in einem der Zweige bei E. Welcher Zweig durchlaufen wird, hängt von den gegebenen Koeffizienten ab.

In dem folgenden codierten Programmstück steht in der ersten Spalte die Adresse des Speicherplatzes, auf dem der Befehl oder die Konstante steht. Dann folgt entweder der Befehl, aufgeteilt in Operations- und Adressteil oder bei Konstanten der Zahlwert; die letzte Spalte enthält Erläuterungen. Es wird angenommen, dass sich die Konstante a in Zelle 100, b in Zelle 101, c in Zelle 102, 4 in Zelle 103, 2 in Zelle 104, -1 in Zelle 105 befindet. Die Speicherzellen 106...109 werden als Zwischenspeicher benutzt, das Ergebnis steht in den Zellen 109 und 110. Die Frage, wie das Programm in den Speicher gekommen ist, soll hier nicht behandelt werden. Diese Aufgabe wird von besonderen Eingabeprogrammen übernommen.

(Durch das Zuordnungszeichen := in den Erläuterungen und im Ablaufdiagramm wird das links stehende Symbol durch den rechts stehenden Ausdruck definiert.)

Fig. 1  
Ablaufdiagramm für die Lösung der quadratischen Gleichung  $ax^2 + bx + c = 0$



Befehlsfolge für die Lösung der quadratischen Gleichung  $ax^2 + bx + c = 0$ .

Adresse	Operationsteil	Adressteil	Erläuterungen (RW=Rechenwerk)
0	B	100	a → RW
1	MUL	102	ac → RW
2	MUL	103	4. a. c → RW
3	C	107	4. a. c → 107
4	B	101	b → RW
5	MUL	101	b <sup>2</sup> → RW
6	SUB	107	D := b <sup>2</sup> - 4ac → RW
7	C	107	D → 107
8	B	100	a → RW
9	SIO	40	Sprung nach 40, wenn a = 0
10	B	107	D → RW
11	SGO	27	Sprung nach 27, wenn D ≥ 0
12	MUL	105	- D → RW
13	C	107	- D → 107
14	W	107	R := sqrt(D) → RW
15	C	107	R → 107
16	B	100	a → RW
17	MUL	104	2a → RW
18	C	108	2a → 108
19	B	101	b → RW
20	DIV	108	b/2a → RW
21	MUL	105	- b/2a = Re → RW
22	C	109	Realteil → 109
23	B	107	R → RW
24	DIV	108	R/2a → RW
25	C	110	Imaginärteil → 110
26	HALT		2 komplexe Lösungen
27	W	107	R := sqrt(D) → RW
28	C	107	R → 107
29	SUB	101	R - b → RW
30	DIV	104	(R - b)/2 → RW
31	DIV	100	x <sub>1</sub> = (R - b)/2a → RW

Der Sprung wird ausgeführt, wenn bei dem jeweiligen Sprungbefehl die angegebene Bedingung erfüllt ist.

Die hier angeführten Befehle sind natürlich kein vollständiger Befehlscode einer Anlage. Sie sind unter dem Gesichtspunkt ausgewählt worden, das folgende kurze Programmstück verständlich zu machen. Es fehlen vor allem die Befehle, die die Ein- und Ausgabe bewirken.

SEV 33 286

#### 4. Beispiel: Lösung der quadratischen Gleichung $ax^2 + bx + c = 0$ mit reellen Koeffizienten

Mit den obigen Befehlen soll nun als Beispiel die Lösung der quadratischen Gleichung mit beliebigen Koeffizienten vorgeführt werden. In Wirklichkeit wird natürlich niemand für einen so einfachen Fall eine Rechenanlage benützen, da man die Ausrechnung schneller selbst durchführt, ehe man umständlich ein Programm dafür erstellt. Das Beispiel soll nur zeigen, wie in einem konkreten Fall ein Problem aufbereitet werden muss, damit ein Rechner es bearbeiten kann.

Das Ablaufdiagramm (Fig. 1) stellt den logischen Ablauf des Programms dar. Die ovalen Gebilde bedeuten Abfragen, von denen der Verlauf des Programms abhängt und die in

Adresse	Operationsteil	Adressteil	Erläuterungen (RW = Rechenwerk)
32	C	109	$x_1 \rightarrow 109$
33	B	107	$R \rightarrow RW$
34	ADD	101	$R + b \rightarrow RW$
35	MUL	105	$-R - b \rightarrow RW$
36	DIV	104	$(-R - b)/2 \rightarrow RW$
37	DIV	100	$x_2 = (-R - b)/2a \rightarrow RW$
38	C	110	$x_2 \rightarrow 110$
39	HALT		zwei reelle Lösungen
40	B	101	$b \rightarrow RW$
41	SNO	43	Sprung nach 43, wenn $b \neq 0$
42	HALT		Lösung unbestimmt
43	B	102	$c \rightarrow RW$
44	DIV	101	$c/b \rightarrow RW$
45	MUL	105	$-c/b = x_1 \rightarrow RW$

Adresse	Operationsteil	Adressteil	Erläuterungen (RW = Rechenwerk)
46	C	109	$x_1 \rightarrow 109$
47	HALT		eine reelle Lösung
.	.	.	.
.	.	.	.
.	.	.	.
100	a		
101	b		
102	c		
103	4		
104	2		
105	-1		

Adresse des Autors:  
Dipl.-Math. Adolf Mühlhahn, Telefunken AG, D-775 Konstanz (Deutschland).

## Die Selektionierung von Kurzschlussauswirkungen in Mittelspannungsnetzen

Von H. Bühler, Küssnacht

621.316.13.027.6

Ein zweckmässiges und wirtschaftliches Kurzschluss-Schutz-System für Mittelspannungsnetze kann mit Primärauslösern aufgebaut werden. Mit dem zunehmenden Ausbau und der Ringbildung steigen aber die Anforderungen an die zeitliche Staffelfähigkeit dieses Systems. Von der Zeitgenauigkeit, d. h. von der kürzesten Staffelfzeit, die eine selektive Auslösung garantiert, hängt es ab, ob man sich für ein anderes System entscheiden muss oder ob man eine kleinere Anzahl Schutzstrecken bildet und damit die Ausdehnung der Störung auf ein grösseres Gebiet in Kauf nimmt.

Dieser Beitrag zeigt den Weg, den das Elektrizitätswerk Küssnacht eingeschlagen hat, nachdem es vor das obige Problem gestellt wurde, und die aus Versuchen ermittelten aufschlussreichen Ergebnisse.

### 1. Das Mittelspannungsnetz des EW Küssnacht

Die Energieverteilung des Elektrizitätswerkes Küssnacht erstreckt sich über ein Gebiet von rund 12,5 km<sup>2</sup>, wozu im heutigen Ausbau 28 Transformatorstationen mit einer total installierten Leistung von 13 MVA in Betrieb stehen. Der Energieumsatz für das Jahr 1963 betrug 23 GWh bei einer maximalen Last von 5 MW. An diesen Werten ist die Industrie mit weniger als 15 % beteiligt.

Fig. 1 zeigt den vermascht aufgebauten 16-kV-Netzteil Dorf; er kann von zwei Seiten mit Energie versorgt werden. Normalerweise erfolgt die Einspeisung in der Meßstation Vordererb (Sammelschiene A) über eine Doppelfreileitung vom Unterwerk Herrliberg. Um über die Lastverteilung in diesem vermascht betriebenen Netz eine Kontrolle zu haben, sind in allen Abgängen der Station Vordererb Stromwandler eingebaut. Durch eine Wahl-Fernmessung können in der zentral gelegenen Mess- und Transformatorstation Dorf jederzeit die Belastungen der Kabel kontrolliert werden; diese Abgänge sind zudem durch thermische Auslöser gegen Überstrom geschützt.

Die zweite, voll leistungsfähige Einspeisemöglichkeit besteht in der Meßstation Dorf (Sammelschiene B) über das Seekabel vom Unterwerk Thalwil. Eine automatische Netzumschaltung wird eingeleitet, sobald die Spannung in der Meßstation Vordererb länger als 5 s ausfällt. Die Rückschaltung erfolgt, wenn die Sammelschiene A in Vordererb wieder 10 min die normale Betriebsspannung geführt hat.

Bei einem Spannungsunterbruch öffnen sofort die Spaltschalter 4, so dass ein sog. Baumnetz entsteht. Sowohl die Umschaltung als auch die Rückschaltung in den Normal-

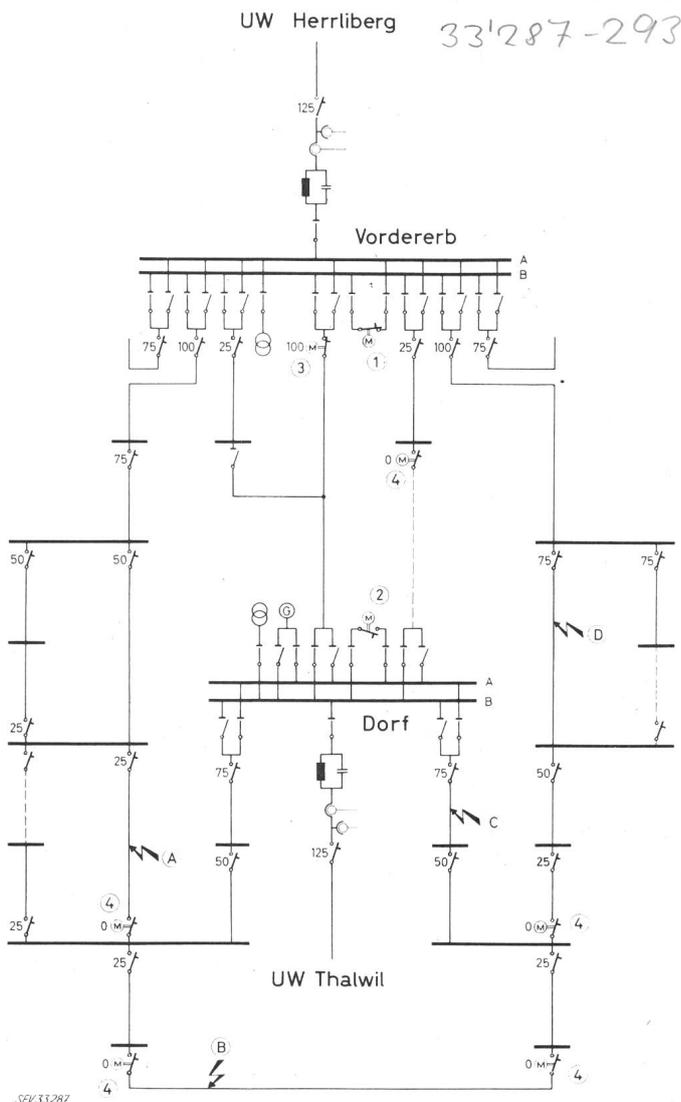


Fig. 1  
16-kV-Netzteil Dorf des EW Küssnacht  
1 Kuppelschalter Vordererb; 2 Kuppelschalter Dorf; 3 Netzverbindungs-schalter; 4 Spaltschalter (mit Momentauslösung); A, B, C, D Kurzschluss-Stellen; 25, 50 usw. Staffelfzeiten in 1/100 s  
— bestehende Leitungen - - - - - projektierte Leitungen