



elektronische datenverarbeitung  
(1968) 6  
S. 293–297

---

H. Kramer

**Optimierung eines Regelkreises  
mit Tischanalogrechner und  
Digitalzusatz**

# Optimierung eines Regelkreises mit Tischanalogrechner und Digitalzusatz

Mit 6 Bildern

H. Kramer, Konstanz

**Zusammenfassung:** Am Beispiel eines Regelkreises wird gezeigt, wie schon mit einem Tischanalogrechner und angeschlossenem Digitalzusatz die Optimierung einer dynamischen Funktion durchgeführt werden kann. Es wird eine Schaltung angegeben, die automatisch das Optimum sucht und dort den Rechner anhält, dem dann die optimalen Werte zu entnehmen sind.

**Summary:** A three-parameters optimization problem of functional is solved by a table-top analogue computer. The example described is a control system optimization. Successive one-parameter optimizations are used to find the optimum. An automatic iteration circuit is connected with a digital coupling system. It runs by steps to the optimal parameters and stops the computing system. The optimal values are stored by integrator accumulators.

Die Optimierung eines Regelkreises mit PID-Regler nach einem vorgegebenen Optimierungskriterium  $Op(x_w)$  ( $x_w$ : Regelabweichung) ist mit der Aufgabe identisch, das Minimum einer Hyperfläche zu finden, die durch die Funktion  $Op(x_w) = Op(V_R, K_I, T_D)$  festgelegt ist. Die Koordinaten des vierdimensionalen Raumes, in dem sich diese Fläche aufspannt, sind die Regelparameter  $V_R$ ,  $K_I$  und  $T_D$  sowie die Funktion  $Op(x_w)$ .

Wenn die Nachbildung der Regelstrecke nicht zu aufwendig ist, kann eine solche Aufgabe schon mit einem kleinen Analogrechner durchgeführt werden, den man mit einem Zusatzgerät mit digitalen Elementen koppeln kann. Für das nachfolgende Beispiel genügt bereits ein Analogrechner mit 19 Rechenverstärkern, von denen 8 als Analogspeicher verwendet werden. Eine derartige Anordnung ist als die einfachste Form eines hybriden Analogrechners zu bezeichnen.

## Aufgabe

Für eine Regelstrecke, die aus drei Verzögerungsgliedern

$$\begin{array}{lll} V_1 = 10 & V_2 = 5 & V_3 = 1 \\ T_1 = 1s & T_2 = 0,5s & T_3 = 0,1s \end{array}$$

besteht, sind die optimalen Werte  $V_R$ ,  $K_I$  und  $T_D$  eines PID-Reglers

$$y = V_R \cdot x_w + K_I \cdot \int x_w \cdot dt + \frac{1}{10} T_D \cdot \frac{dx_w}{dt}$$

(y: Stellgröße)

zu ermitteln, die im Bereich  $V_R < 1$ ,  $K_I < 1s^{-1}$  und  $T_D < 0,5s$  liegen. Die Optimierung soll so erfolgen, daß das Zeitintegral der quadratischen Regelabweichung möglichst klein wird, d.h.

$$Op(x_w) = \int x_w^2 \cdot dt \rightarrow \text{Min}$$

Die Integralfunktion  $Op(x_w)$  ist aus der Regelabweichung  $x_w$  zu errechnen, die am simulierten Regelkreis bei sprungförmiger Störung entsteht. Es genügt, die Integrationsdauer gleich der Einschwingzeit des Regelkreises zu wählen, da dann praktisch kein Beitrag mehr zum Integral geliefert wird.

## Optimierungsverfahren

Aus der Vielzahl der Optimierungsverfahren wurde das nach Gauß-Seidel ausgewählt, weil es mit kleineren Analogrechnern weniger aufwendig zu verwirklichen ist.

Bei einem Wert des Kriteriums  $Op(V_R^{(0)}, K_I^{(0)}, T_D^{(0)})$  beginnend, wird zunächst ein Parameter, z. B.  $V_R$ , solange variiert, bis keine Verbesserung der Funktion  $Op(V_R, K_I^{(0)}, T_D^{(0)})$  mehr zu erreichen ist. Danach wird der nächste Parameter verändert und das Minimum von  $Op(V_R^{(1)}, K_I, T_D^{(0)})$  gesucht. Nachdem auch  $T_D$  variiert wurde, beginnt man von neuem mit der Variation von  $V_R$ . Die drei Parameter werden, zyklisch vertauscht, solange verändert, bis keine Verbesserung von  $Op$  mehr zu erreichen ist. Man ist so auf der Hyperfläche in Richtung der drei Parameter  $V_R$ ,  $K_I$  und  $T_D$  hinabgewandert und hat den tiefsten Punkt der Mulde erreicht. Natürlich findet man nach diesem Verfahren bei komplizierteren Optimierungsfunktionen nur das Minimum, auf dessen Hang der angenommene Startpunkt lag.

## Flußdiagramm

Um das Flußdiagramm leichter zu verstehen, verfolgen wir zunächst für einen Parameter den Ablauf einer Teiloptimierung. Der gewählte Parameter wird dabei mit einer konstanten Schrittweite  $\epsilon$  verändert. Nach jedem Schritt wird der neu ermittelte Wert des Kriteriums  $Op^{(i)}$  mit dem vorher berechneten  $Op^{(i-1)}$  verglichen. Ist der neue Wert nämlich größer oder gleich dem vorherigen, d.h. ist  $Op^{(i)} - Op^{(i-1)} \geq 0$  (dieser Fall soll digital durch  $\Delta = 1$  gekennzeichnet werden), so führte die Parametervariation nicht zu der gewünschten Annäherung an das Optimum.

Vom Startpunkt  $Op^{(0)}$  aus (Bild 1) kann der 1. Schritt, der durch den Zustand ( $\bar{O}\bar{P}$ ) bezeichnet ist, demnach den  $Op$ -Wert verkleinern ( $\Delta = 0$ ) (linkes Bild) oder vergrößern ( $\Delta = 1$ ) (rechtes Bild). Für  $\Delta = 0$  ist das Signum von  $\epsilon$  günstig, so daß dieser Schritt bereits zu einer Verkleinerung von  $Op$  führt. In nachfolgenden Schritten (Zustand:  $OP$ ) wird deshalb der Parameter in dieser Richtung solange variiert, bis man erstmals  $\Delta = 1$  erhält. Jetzt ist das Minimum von  $Op$  gerade durchlaufen worden. Der nachfolgende Zustand ( $\bar{O}\bar{P}$ ) setzt nun den letzten Schritt wieder zurück.

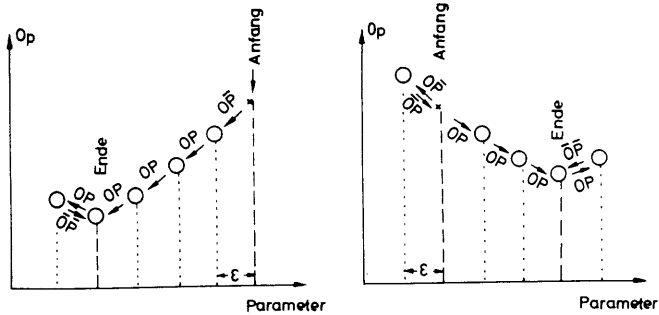


Bild 1 Teiloptimierung der Funktion  $Op$  nach einem Parameter  
Zustandsfolge:  $O\bar{P}$ ,  $(O\bar{P})$ ,  $OP$ , ...  $OP$ ,  $O\bar{P}$ .

Erhält man bereits beim ersten Schritt ( $O\bar{P}$ ) ein  $\Delta = 1$ , so wurde der Parameter in der falschen Richtung geändert. Das Signum von  $\epsilon$  ist deshalb im folgenden Schritt ( $\bar{O}P$ ) zu wechseln, um diesen ungünstigen Schritt wieder rückgängig zu machen. Mit dem richtigen Vorzeichen von  $\epsilon$  nähert man sich nun in  $\epsilon$ -Schritten ( $OP$ ) wie zuvor dem Teiloptimum von  $Op$ .

Im allgemeinen wird sich der Optimierungsablauf aus vielen solchen Teiloptimierungen zusammensetzen, wie aus dem Flußdiagramm (Bild 2) ersichtlich ist.

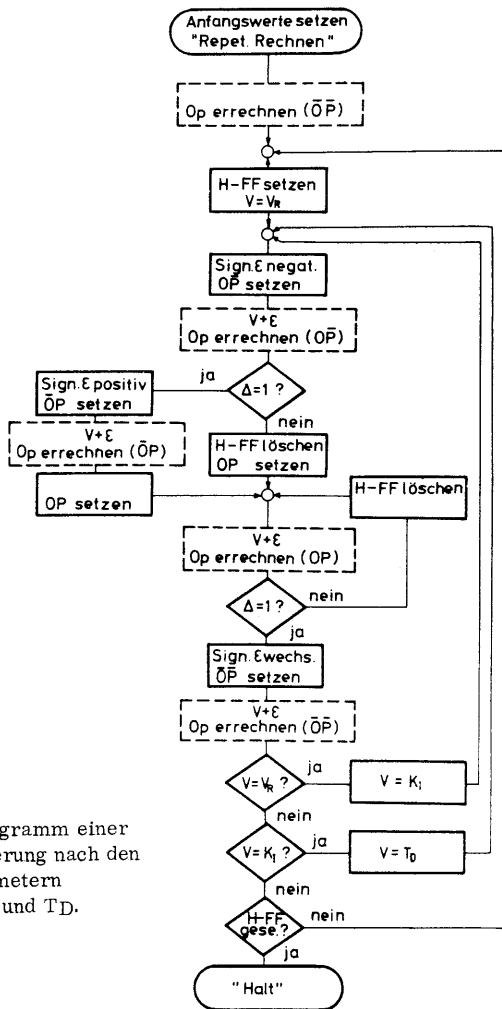


Bild 2  
Flußdiagramm einer  
Optimierung nach den  
3 Parametern  
 $V_R$ ,  $K_I$  und  $T_D$ .

Vor Beginn der Rechnung sind die Anfangswerte im analogen und digitalen Teil der Anlage einzustellen, darunter die Startwerte der Reglerparameter  $V_R^{(0)}$ ,  $K_I^{(0)}$  und  $T_D^{(0)}$ . Der Optimierungsablauf wird durch die Operation "Repetierend Rechnen" vom Analogrechner gesteuert. Im Analogrechner wird der gewählte Parameter ( $V$ ) pro Rechenschritt um  $\epsilon$  verändert. Während der eingestellten Rechenzeit ermittelt er den jeweiligen Kriteriumswert  $Op$  (im Flußdiagramm als gestrichelte Funktionsblöcke erkennbar). Während der Pausenzeit des Analogrechners trifft der Digitalzusatz die programm-gemäße Entscheidung über den nächsten Rechenschritt (ausgezogene Funktionsblöcke).

Zunächst berechnet der Analogrechner den Wert des Kriteriums am Startpunkt  $Op^{(0)}$  (Zustand:  $O\bar{P}$ ). Als Variationsparameter wird  $V = V_R$  gewählt. Mit negativem  $\epsilon$  beginnend, erfolgt über die Zustände  $O\bar{P}$ , (eventuell  $\bar{O}P$ ),  $OP$ ,  $O\bar{P}$  die besprochene Teiloptimierung dieses Parameters. Nach  $V = K_I$  folgt in einer Schleife der Optimierungsablauf für  $K_I$  und dann in einer zweiten Schleife der für  $T_D$ . Die äußerste Schleife im Diagramm sorgt dafür, daß die Variation erneut mit  $V_R$  beginnt.

Das Ende der Optimierung wird im vorliegenden Fall nicht dadurch festgelegt, daß der Unterschied zweier aufeinanderfolgender  $Op$ -Werte unter eine vorgegebene Grenze sinkt. Die festzusetzende Grenze müßte dafür auf die Schrittweite abgestimmt werden, was Informationen über die Hyperfläche voraussetzt. Hier wird das Optimierungsende nur durch die Schrittweite  $\epsilon$  festgelegt.

Das Programm soll anhalten, wenn durch die vorgegebene Schrittweite keine bessere Annäherung an das Optimum mehr möglich ist. Das ist der Fall, wenn alle günstigen Optimierungsschritte (das sind die Zustände  $O\bar{P}$  und  $\bar{O}P$ )  $\Delta = 1$  ergeben. Deshalb wird am Anfang der Rechnung und im Optimierungszyklus nach jedem neuen Beginn mit  $V_R$  ein Halt-Flipflop (H) gesetzt. Jeder erfolgreiche Optimierungsschritt setzt dieses H-Flipflop zurück. Findet während eines Zyklus kein erfolgreicher Schritt statt, so steuert der Digitalzusatz den Analogrechner in "Halt".

Die optimalen Werte für die drei Parameter sind nun bis auf einen Fehlerabstand  $\epsilon$  gefunden. Falls gewünscht, kann die Optimierung von hier aus mit einer kleineren Schrittweite neu gestartet werden.

#### Schaltung des analogen und digitalen Teils

In den Bildern 3 und 4 ist dargestellt, wie das beschriebene Verfahren auf einem Analogrechner mit Digitalzusatz zu verwirklichen ist.

Bild 3 zeigt in Blockdarstellung den simulierten Regelkreis mit den drei Analogspeichern für die Regelparameter. Sie können mit den Schaltern  $S_2$  und  $S_3$  nacheinander um  $\epsilon$  verändert werden.  $S_1$  steuert das Vorzeichen von  $\epsilon$ . Nach jedem Schritt wird das aus  $x_w$  errechnete Optimierungskriterium mit dem vorherigen Wert verglichen. Die Entscheidung über diese Differenz  $\Delta$  erhält der Digitalzusatz. Der Rechenakt T setzt nun die Flipflops dem Flußdiagramm entsprechend um. Die von den Flipflops angesteuerten Schalter  $S_1$  bis  $S_3$  wählen für den nächsten Rechenschritt den gewünschten Parameter und das Vorzeichen von  $\epsilon$  aus. Schließlich steuert  $S_4$  den Analogrechner in "Halt".

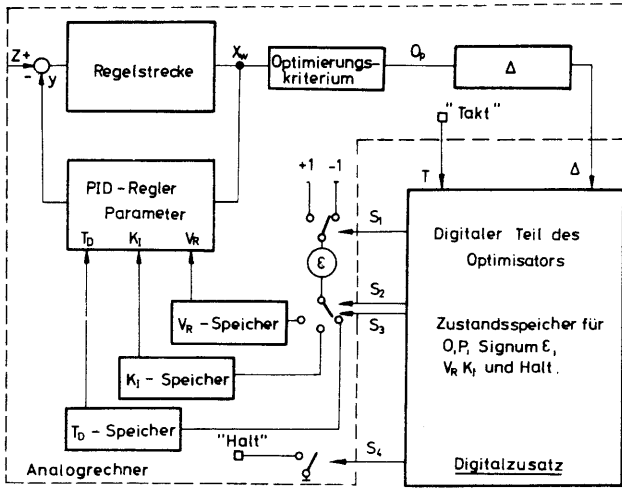


Bild 3 Blockschaltung des Optimisators

Die detaillierte Rechenschaltung des analogen und digitalen Teils ist in Bild 4 wiedergegeben. Die Regelstrecke mit Verzögerungsgliedern ist durch drei rückgeführte Integrierer nachgebildet. Beim simulierten PID-Regler ist der D-Anteil der Regelstrecke entnommen. Um eine Variation der Reglerparameter  $V_R$ ,  $K_I$  und  $T_D$  auf elektrische Weise zu ermöglichen, werden ihre Werte über Multiplizierer in der Schaltung wirksam. Die momentanen Werte liegen hierzu ständig in drei Speicherkreisen vor. In einem Speicherkreis wird der einmal vorgegebene Wert während der Pausenzeit des Analogrechners von einem komplementären Speicher ( $\bar{S}_p$ ) übernommen und während der nachfolgenden Rechenzeit gehalten. Am Beginn der Rechenzeit übernimmt ein normaler Speicher ( $S_p$ ) den Wert und hält ihn während der nächsten Pausenzeit. Zu diesem fortwährend umlaufenden Wert addiert der komplementäre Speicher während der Repetierpause das angebotene  $\epsilon$ .

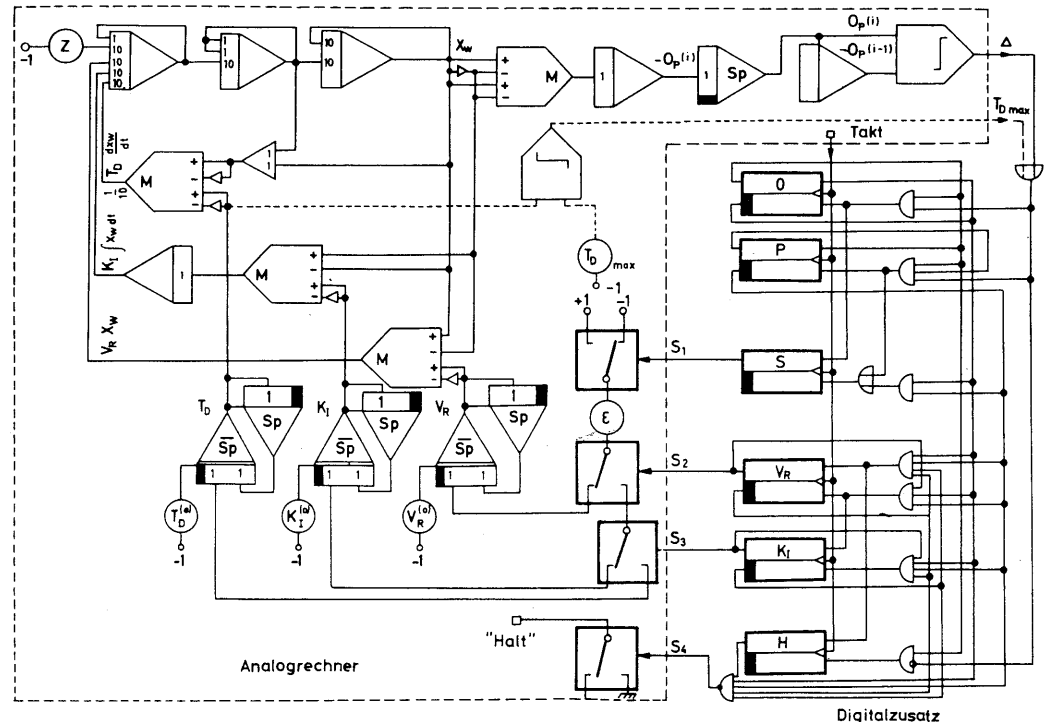


Bild 4 Analoge und digitale Rechenschaltung

Das Zeitintegral der quadratischen Regelabweichung des jeweiligen Rechenschritts  $Op^{(i)}$  wird ebenfalls gespeichert und liegt bis zum Ende des nächsten Rechenschritts am nachfolgenden Integrierer vor. So ist es möglich, die  $Op$ -Werte zweier aufeinanderfolgender Rechnungen zu vergleichen und mit einem Komparator die Entscheidung über  $\Delta = 1$  oder  $\Delta = 0$  zur Steuerung des Digitalzusatzes zu fällen.

Für die Benutzer kleinerer Analogrechner sei erwähnt, daß sich die Ausgangsverstärker aller Multiplizierer erübrigen, wenn man das Multiplizieretzwerk unter Berücksichtigung seiner Eingangswertigkeit direkt auf den Summenpunkt des nachfolgenden Verstärkers schaltet und das Vorzeichen eines Eingangs wechselt. Der analoge Schaltungsteil ist so bereits mit 19 Rechenverstärkern zu erfüllen.

Im digitalen Teil der Schaltung sind folgende Zustände der Optimierung durch Flipflops festgelegt:

- O: } kennzeichnen die Zustände  $\bar{O}\bar{P}$ ,  $\bar{O}P$ ,  $O\bar{P}$
- P: } und  $OP$  im Ablauf einer Teiloptimierung
- S: = 1 Signum von  $\epsilon$  ist positiv
- $V_R$ : = 1 variiere  $V_R$
- $K_I$ : = 1 variiere  $K_I$
- bzw.:  $T_D = \bar{V}_R \cdot \bar{K}_I$ : variiere  $T_D$
- H: = 1 bisher kein erfolgreicher Optimierungsschritt

Die Verknüpfung der Flipflops mit "UND"- und "ODER"-Gliedern und ihre Steuerung entsprechend der  $\Delta$ -Entscheidung des Analogrechners ist aus dem spezifizierten Flußdiagramm in Bild 5 zu entnehmen. In den digitalen Funktionsblöcken sind dort die Flipflop-Umsetzungen eingetragen, die in Bild 2 verbal ausgedrückt sind.

Zum Beispiel ist nach dem Zustand ( $O\bar{P}$ ) und ( $\Delta = 0$ ) die Anweisung "H-FF löschen", "OP setzen" zu erfüllen. Dazu sind die Flipflop-Umsetzungen  $H: = 0$  und  $P: = 1$  auszuführen. Der neue Zustand ist so durch  $\bar{H} = P: = O \cdot \bar{P} \cdot \bar{\Delta}$  beschrieben (lies: Setze  $\bar{H}: = 1$  und  $P: = 1$ , wenn  $O = 1$  "UND"  $\bar{P} = 1$  "UND"  $\bar{\Delta} = 1$ ).

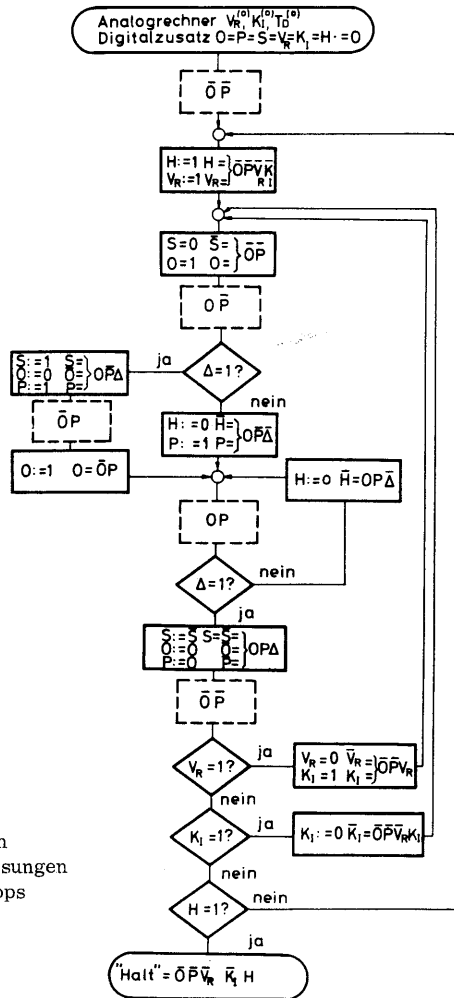


Bild 5  
Flußdiagramm  
mit Setzanweisungen  
für die Flipflops

Aus Bild 5 sind für die 6 Flipflops folgende Umsetzanweisungen zu entnehmen:

$$\begin{aligned}
 O: &= \bar{O} \cdot \bar{P} + \bar{O} \cdot P & \bar{O}: &= O \cdot P \cdot \Delta + O \cdot \bar{P} \cdot \bar{\Delta} \\
 P: &= O \cdot \bar{P} \cdot \Delta + O \cdot P \cdot \bar{\Delta} & \bar{P}: &= O \cdot P \cdot \Delta \\
 S: &= O \cdot \bar{P} \cdot \Delta + O \cdot P \cdot \bar{\Delta} & \bar{S}: &= \bar{O} \cdot \bar{P} + O \cdot P \cdot \Delta \\
 V_R: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot \bar{K}_I & \bar{V}_R: &= \bar{O} \cdot \bar{P} \cdot V_R \\
 K_I: &= \bar{O} \cdot \bar{P} \cdot V_R & \bar{K}_I: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot K_I \\
 H: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot \bar{K}_I & \bar{H}: &= O \cdot \bar{P} \cdot \bar{\Delta} + O \cdot P \cdot \bar{\Delta}
 \end{aligned}$$

Ohne auf allgemeine Rechenregeln einzugehen, lassen sich einige der Anweisungen leicht vereinfachen.

$$\text{z.B. } P: = O \cdot \bar{P} \cdot (\Delta + \bar{\Delta}) = O \cdot \bar{P} = 0$$

So erhält man schließlich die Anweisungstabelle

$$\begin{aligned}
 O: &= \bar{O} & \bar{O}: &= O \cdot \Delta \\
 P: &= O & \bar{P}: &= O \cdot P \cdot \Delta \\
 S: &= O \cdot \Delta & \bar{S}: &= \bar{O} \cdot \bar{P} + O \cdot P \cdot \Delta \\
 V_R: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot \bar{K}_I & \bar{V}_R: &= \bar{O} \cdot \bar{P} \cdot V_R \\
 K_I: &= \bar{O} \cdot \bar{P} \cdot V_R & \bar{K}_I: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot K_I \\
 H: &= \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot \bar{K}_I & \bar{H}: &= O \cdot \bar{\Delta}
 \end{aligned}$$

nach der der digitale Teil des Optimisators aufgebaut ist. Das oberste "UND"-Glied der Schaltung realisiert z.B. die Anweisung  $\bar{O}: = O \cdot \Delta$ .

Die Anweisungen für die DA-Schalter  $S_1$  bis  $S_3$  ergeben sich ohne weiteres aus den Zuordnungen in Bild 4, wenn berücksichtigt wird, daß sich die Schaltkontakte des benutzten Geräts für  $S_i = 1$  in Ruhestellung befinden.

Die Optimierung ist schließlich beendet, und  $S_4$  steuert den Analogrechner in "Halt", wenn während eines Optimierungszyklus das H-Flipflop nicht zurückgesetzt wurde. Somit ist: "Halt" =  $\bar{S}_4 = \bar{O} \cdot \bar{P} \cdot \bar{V}_R \cdot \bar{K}_I \cdot H$ .

Es ist möglich, daß die Hyperfläche in dem vorgegebenen Bereich der Parameter keine Mulde hat, sondern nach einer Seite offen ist. In diesem Fall ist die Variation eines Parameters zu beschränken. In vorliegendem Beispiel ist deshalb  $T_D$  bei  $T_{D \max} = 0,5$  zu begrenzen. Wird  $T_{D \max}$  überschritten, so erhält der digitale Teil diese Information über einen Komparator. Wie in Bild 4 punktiert eingefügt, wirkt die Überschreitung wie  $\Delta = 1$  im digitalen Teil des Optimisators. Der letzte Schritt wird zurückgesetzt und die Optimierung mit dem nächsten Parameter fortgeführt.

### Durchführung der Optimierung

Geeignete Anfangswerte für  $V_R$ ,  $K_I$  und  $T_D$  (z.B.  $V_R = 0,2$ ,  $K_I = 0,2 \text{ s}^{-1}$ ,  $T_D = 0,2 \text{ s}$ ), welche natürlich im stabilen Bereich des Regelkreises liegen müssen, werden in die Speicher übertragen. Alle Flipflops sind auf Null zu stellen. Am Potentiometer ( $z$ ) wird die Höhe der Störung so eingestellt, daß  $x_w$  und der Wert des Kriteriums im Spannungsbereich des Analogrechners liegen. Für die gegebenen Anfangswerte erhält man mit  $z = 0,3$  eine ausreichende Aussteuerung. Dieser Wert kann in der Nähe des Optimums bis ca. 0,6 erhöht werden.

Man überzeugt sich leicht, daß bei Integrationsgeschwindigkeitskonstanten von  $10 \text{ s}^{-1}$  bereits eine Rechenzeit von ca. 3 s genügt, um den Op-Wert zu errechnen. Sind auf dem zur Verfügung stehenden Rechner größere Geschwindigkeitskonstanten wählbar, so ist die Rechenzeit im gleichen Verhältnis zu kürzen.

Dem Takt (T) für den digitalen Teil ist insofern Beachtung zu schenken, als die Umsetzung der Flipflops während der Haltezeit der Integrierer und Speicher erfolgen muß. Bei Rechnern, die durch mechanische Relais gesteuert werden, kann es vorkommen, daß ein am Ende der Rechenzeit entnommenes Taktsignal die Flipflops, und somit die Schalter  $S$ , schneller umsetzt als die Steuerschalter der Integrierer und Speicher. So können Fehlschaltungen eintreten, die dafür sorgen, daß das gesuchte Optimum nie ganz erreicht wird. Der Rechner hält nicht an. Durch geeignetes Verzögern des Taktimpulses im Digitalzusatz mit einem Zeitglied ist der richtige Zeitpunkt der Umsetzungen einstellbar.

Nun kann der Analogrechner durch "Repetierend Rechnen" gestartet werden. Die Flipflops zeigen den Ablauf der Optimierung an. Ist das Wertetripel nicht mehr zu verbessern, so geht der Rechner selbsttätig in "Halt". Wird danach eine bessere Annäherung an das Optimum gewünscht, so ist Potentiometer ( $\epsilon$ ) entsprechend zu verkleinern. Durch Löschen aller Flipflops startet der Rechner von neuem. Die optimalen Werte für  $V_R$ ,  $K_I$  und  $T_D$  im vorgegebenen Bereich liegen schließlich in den drei Speichern vor und können gemessen werden.

Das behandelte Optimierungsproblem wurde auf einem Telefonen-Tischanalogrechner RAT 740/741 gerechnet, der zur digitalen Steuerung mit einem Digitalzusatz DEX 100 gekoppelt war (Bild 6). Die Geräte wurden freundlicherweise von der Fachabteilung Analogrechner, AEG-Telefunken Konstanz, zur Verfügung gestellt, wofür sich der Verfasser bedanken möchte.

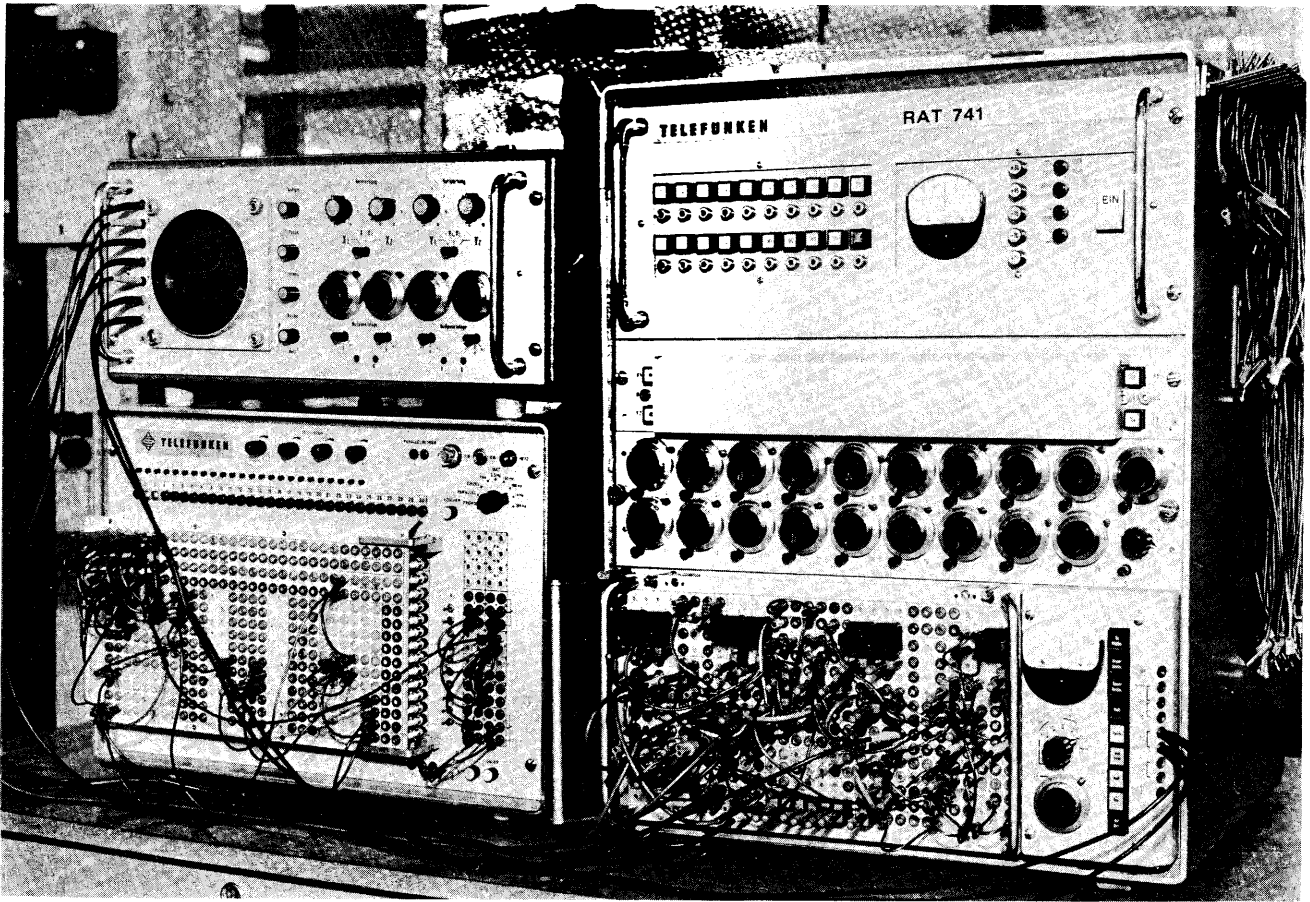


Bild 6 AEG-TELEFUNKEN-Tischanaloglechner RA 741 mit Zweistrahloszillograph OMS 800 und Digitalzusatz DEX 100