

MINCAL 4
SPEICHERPROGRAMMIERTER DIGITALRECHNER
INFORMATIONSBESCHREIBUNG

- vorläufig -

HEINRICH DIETZ INDUSTRIE-ELEKTRONIK
MUELHEIM/RUHR KOELNER STRASSE 115
TELEFON 48 85 41 TELEX 08 56770

| | <u>Inhalt</u> | <u>Seite</u> |
|--|---------------|--------------|
| <u>Vorbemerkung</u> | | 2 |
| <u>1. Technische Daten</u> | | 3 |
| <u>2. Konzeption</u> | | 4 |
| 2.1) Anwendungsbereich | | 4 |
| 2.2) Wortlänge | | 4 |
| 2.3) Komprimiertes Speichern, Doppelwort- und Gleitkommarechnung | | 5 |
| 2.4) Speicherkapazität | | 5 |
| 2.5) Geschwindigkeit | | 6 |
| 2.6) Parallel-Ein/Ausgang und Puffersystem | | 7 |
| 2.7) Periphergeräte | | 7 |
| 2.8) Das Prioritätssystem | | 8 |
| 2.9) Programmbefehle | | 8 |
| 2.10) Programmierhilfen und Testmöglichkeiten | | 9 |
| 2.11) Technologie | | 9 |
| <u>3. Struktur</u> | | 10 |
| 3.1) Die zentrale Recheneinheit | | 10 |
| 3.2) Die festen Register | | 10 |
| 3.3) Die Prioritätslogik | | 11 |
| 3.4) Die Verbindung mit den Puffern | | 11 |
| 3.5) Parallel-Ein/Ausgang | | 12 |
| <u>4. Programmfunktionen</u> | | 12 |
| 4.1) Darstellung von Daten | | 12 |
| 4.2) Darstellung von Instruktionen | | 13 |
| 4.3) Adressen | | 14 |
| 4.4) Indirekte Adressierung | | 14 |
| 4.5) Adreßmodifikation | | 14 |
| 4.6) Festwertoperation | | 14 |
| 4.7) Datentausch innerhalb der festen Register | | 14 |
| 4.8) Verzweigungen | | 15 |
| 4.9) Unterprogramme | | 15 |
| <u>5. Programmbefehle</u> | | 15 |
| <u>6. Ausführungsformen</u> | | 26 |
| 6.1) Grundausführung | | 26 |
| 6.2) Schreibmaschine | | 26 |
| 6.3) Lochstreifeneinheit | | 27 |
| 6.4) Erweiterung der Speicherkapazität | | 27 |
| 6.5) Puffer vom R-Typ | | 27 |
| 6.6) Puffer vom D-Typ | | 28 |
| 6.7) Puffer vom A-Typ | | 29 |
| 6.8) Parallel-Ein/Ausgang | | 29 |
| 6.9) Standard-Konsole | | 29 |

Vorbemerkung

Diese Informationsbeschreibung soll Interessenten und künftigen Benutzern des MINCAL 4 einen Überblick über dessen Eigenschaften geben; sie ist also weder eine Programmierfibel noch ein "interface manual".

Für eine Kurzinformation genügt es zu lesen:

- | | |
|--|---|
| zur Einordnung in das Spektrum heutiger Digitalrechner | Kapitel 1 und 2 |
| zum Verstehen des inneren Aufbaus | Kapitel 3 |
| für den Programmierer | Kapitel 4 und 5 sowie Abschnitt 2.10 |
| für den Techniker, der einen MINCAL 4 richtig dimensionieren und ihn für seine Zwecke einsetzen will | Kapitel 6 sowie Abschnitte 2.6 bis 2.11 |

Weitere Informationen stehen auf Wunsch gerne zur Verfügung.

1. Technische Daten des MINCAL 4

| | |
|--|--|
| Typ: | Speicherprogrammierter Einadreß-Digitalrechner mit fester Wortlänge und Festkomma |
| Taktfrequenz: | 3.2 MHz |
| Wortzeit: | 20 µs |
| Wortlänge: | 24 bit + Vorzeichen-bit + Prüfbit |
| Speicherkapazität: | 2048 (2k) bzw. 4096 (4k) Worte (Grundausführung) erweiterungsfähig bis 32768 Worte (32 k) sämtlich direkt adressierbar |
| Befehle: | 60 |
| Programmniveaus: | 8 |
| Arbeits- und Hilfsregister: | 2 je Niveau |
| Indexregister: | 4 je Niveau |
| Bedingungseingänge: (Sensoren) | 7 gemeinsame sowie 7 weitere je Niveau |
| Puffer: | max. 8 eingebaut, für je 4096 Adressen bzw. Formatangaben |
| Parallel-Ein/Ausgang: | für 8 Gruppen zu je 4096 Adressen |
| Standard-Periphergeräte: | Elektrische Eingabe-Ausgabe-Schreibmaschine IBM 735 15 Z/s |
| | 8-Kanal-Streifenlocher 75 Z/s |
| | 8-Kanal-Streifenleser * 75 Z/s |
| Weitere Peripher-Anschlüsse: (auf Wunsch) | Drucker-, Löcher-, Fernschreibausgang Ein/Ausgänge für digitale Informationen und Schaltbefehle Analog-Ein/Ausgänge |
| Verwendete Halbleiter: | Monolithische integrierte Schaltkreise und Silizium-Epitaxial-Planartransistoren |
| Kernspeicher: | Ferritkerne mit weitem Temperaturbereich |

* auf Wunsch 300 Z/s

2. Konzeption des MINCAL 4

2.1) Anwendungsbereich

Das Bedürfnis nach rascher und vollständiger Sammlung von Informationen und ihrer unmittelbaren Auswertung, der Wunsch nach Aufdeckung unbekannter Zusammenhänge und schneller Gewinnung von Resultaten und insbesondere die verfeinerte Steuerung von Arbeitsprozessen erfordern in zunehmendem Maße den Einsatz von datenverarbeitenden Geräten.

In letzter Zeit ist vor allem der Wunsch nach kleineren Rechnern hervorgetreten, die einerseits nicht zu teuer, einfach in Bedienung und Programmierung und problemlos hinsichtlich der Wartung sind, auf der anderen Seite aber auch hinreichend schnell arbeiten, eine leistungsfähige Befehlsorganisation besitzen, sich den jeweiligen Aufgaben organisch anpassen und mit deren Vergrößerung erweitern lassen.

Das gilt - im technischen Bereich - für recht unterschiedliche Anwendungsgebiete, die von Rechenproblemen wissenschaftlicher Natur über die Sammlung und Verarbeitung von Meßdaten bis zur Steuerung von Prozessen reichen. Dank der technologischen Fortschritte, insbesondere in der Entwicklung von schnellen, kompakten Halbleiterbauteilen, auf den Gebieten der Speicher und der Peripheriegeräte sind heute digitale Rechner verfügbar, die diesen Forderungen weitgehend genügen.

Der Digitalrechner MINCAL 4 ist ein universelles, frei programmierbares Datenverarbeitungsgerät, das für den beschriebenen Anwendungsbereich zugeschnitten ist. Die Grundausführung des MINCAL 4 ist so weit reduziert, daß sie auch für kleinere Aufgaben wirtschaftlich eingesetzt werden kann; sie umfaßt jedoch von vornherein die volle Befehlsvielfalt und alle Hilfsmittel für schnelle und bequeme Programmierung, Eingabe und Ausgabe. Hinsichtlich Kapazität, Peripheriegeräten und Ein-Ausgabemöglichkeiten - vor allem für den Anschluß an Prozeßgrößen - ist der MINCAL 4 in hohem Maße ausbaufähig und flexibel.

Kapazität und Wortlänge, Arbeitsgeschwindigkeit und Befehlsvielfalt, Ein-Ausgabe-Funktionen und Technologie erschließen dem Digitalrechner MINCAL 4 viele Anwendungsmöglichkeiten, sei es als selbständiges Gerät für wissenschaftlich-technische Berechnungen, sei es für Aufgaben der Meßwertverarbeitung oder als Prozeßrechner.

2.2) Wortlänge

Der MINCAL 4 ist ein Rechner mit fester Wortlänge. Das bedeutet, daß er Informationen konstanten Formats verarbeitet, im Speicher aufhebt und mit seiner Peripherie austauscht.

Jede Programminstruktion und jeder numerische Wert erscheinen in Gestalt eines Wortes, dessen Länge und Aufbau daher gewissermaßen die "Visitenkarte" des Rechners hinsichtlich Befehlsvielfalt und Rechengenauigkeit ist.

Numerische Werte, mit denen der MINCAL 4 arbeitet, liegen in binärer Form vor; jedes Wort umfaßt 24 Binärstellen (24 bit); es können also Zahlen von 2^{-1} bis 2^{-24} verarbeitet werden. Anders ausgedrückt: Die Rechengenauigkeit beträgt 2^{-24} , bezogen auf eine numerische Operation, was einem Rechenfehler von weniger als 10^{-7} entspricht. Bedenkt man, daß z.B. technische Meßwerte häufig mit einer relativen Unsicherheit von 10^{-3} bis 10^{-4} behaftet sind, so kann man erwarten, daß die zusätzlichen Fehler, die bei anschließenden, langwierigen Rechenvorgängen auftauchen, vernachlässigbar klein sein werden.

Bei der Ein- und Ausgabe haben Zahlenwerte im allgemeinen dezimale Form. Zur Darstellung einer Dezimalstelle werden 4 Binärstellen benötigt; ein 24-bit-Wort kann also auch eine 6-stellige Dezimalzahl aufnehmen, entsprechend einer Auflösung von 10^{-6} . Und für die Aufnahme von 4 alphanumerischen Zeichen (zu je 6 bit) ist auch gerade ein Wort nötig.

Außer den 24 Binärstellen ist im MINCAL 4 jedem Wort eine Vorzeichenstelle zugeordnet; positive und negative Zahlen werden also echt unterschieden. Darüberhinaus ist im Kernspeicher jedes Wort mit einer weiteren Binärstelle versehen, die zur Kontrolle der Gesamtzahl aller Binärstellen dient, die eine 1 enthalten (Prüfbit). Ein Wort im Kernspeicher hat also 26 bit.

Außer für die Rechengenauigkeit ist die Wortlänge auch für die Vielfalt der Programmbefehle entscheidend. Eine Instruktion hat die Länge eines Wortes, also 24 bit. Beim MINCAL 4 ist sie außerordentlich klar aufgebaut: 6 Binärstellen charakterisieren den eigentlichen Befehl und gestatten die Unterscheidung der 60 verschiedenen Operationsarten, die der MINCAL 4 durchführen kann. Drei weitere bits ergänzen den Befehl, z.B. in seiner Anwendung auf bestimmte Register im Kernspeicher. Mit den restlichen 15 Binärstellen kann man jeden Speicherplatz, bis zur vollen Kapazität von $2^{15} = 32768$ Worten, direkt ansprechen oder jeden von jeweils 4096 Daten-Ein- oder -Ausgängen an 8 Ein/Ausgabe-Gruppen auswählen.

2.3) Komprimiertes Speichern, Doppelwort- und Gleitkommarechnung

Obwohl die Wortlänge von 24 bit für genaue numerische Verarbeitung sehr nützlich ist, wird ein Wort nur zur Hälfte ausgenutzt, wenn man z.B. einen Meßwert mit einer Auflösung von $5 \cdot 10^{-4}$ darin unterbringt. Sollen viele derartige Meßwerte gespeichert werden, so teilt man besser jeden Speicherplatz in 2 Hälften zu je 12 bit, in denen je ein Wort mit einer Auflösung von 2^{-12} steht. Informationen mit noch geringerer Länge kann man auf gleiche Weise zu mehreren in einem Wort unterbringen. Die Befehle des MINCAL 4 erlauben einfache Programmabläufe zum Zusammenfügen und Trennen der komprimiert abgespeicherten Informationen.

Auf der anderen Seite kann, insbesondere bei wissenschaftlichen Programmen, die Wortlänge nicht ausreichen, um eine genügende Rechengenauigkeit zu erzielen. Ein Ausweg ist das Arbeiten mit doppelter Wortlänge, d.h. mit zwei Worten zu je 24 bit, die eine zusammenhängende Zahl mit einer Auflösung von 2^{-48} enthalten. Die Operationen des MINCAL 4 enthalten bereits Ansätze hierfür, so daß man mit wenigen Befehlen einfache und relativ schnelle Programmabläufe gestalten kann, die das Rechnen mit doppelter Wortlänge erlauben.

Der MINCAL 4 ist ein Festkomma-Rechner; bei Multiplikation und Division ist das Komma stets am gleichen Ort, und zwar links von der Binärstelle mit der höchsten Wertigkeit. Alle numerischen Werte sind daher Brüche; die 24 bit eines Wortes haben die Wertigkeit 2^{-1} bis 2^{-24} . Wo die dadurch gegebene relative Genauigkeit zwar ausreicht, die zu verarbeitenden Zahlen aber sehr verschiedene Größenordnungen haben, empfiehlt sich das Rechnen mit gleitendem Komma. Hierbei wird eine Zahl durch eine Mantisse (zwischen 0.5 und 1) und einen Exponenten (negative Potenz von 2) dargestellt. Der MINCAL 4 ist mit Befehlen ausgerüstet, die den Aufbau verhältnismäßig einfacher Programme für Gleitkomma-Rechnung zulassen.

2.4) Speicherkapazität

Im Kernspeicher des MINCAL 4 sind - in beliebiger Anordnung - Instruktionen, Festwerte und Variablenplätze enthalten, die für das Verarbeitungsprogramm benötigt werden. Die Kapazität des Speichers ist also ein wichtiges Maß für die Leistungsfähigkeit, - aber auch ein Kostenfaktor.

Die Grundausführung des MINCAL 4 enthält eine Speichereinheit mit entweder 2048 oder 4096 Worten Kapazität. Da ein Wort beim MINCAL 4 viel "wert" ist, d.h. einen relativ großen Zahlenwert oder einen fein gegliederten Programmbebefhl enthält, lassen sich bereits mit 2048 Worten recht umfangreiche Programme beherrschen; die Erweiterung auf 4096 Worte ist nachträglich möglich.

Für größere Programme oder einen höheren Bedarf an Festwerten oder Variablenspeichern werden weitere Speichereinheiten von je 4096 Worten Kapazität vorgesehen; insgesamt sind 8 dieser Einheiten möglich, entsprechend $2^{15} = 32768$ Worten, die sämtlich direkt adressiert werden können.

Man kann den Kernspeicher mit Instruktionen, Festwerten oder Variablen in beliebiger Weise belegen. Allerdings haben 8 Plätze des Speichers feste Funktionen (Startadresse, 4 Indexregister, Programmstand, Arbeits- und Hilfsregister); werden die 8 verschiedenen Programm niveaus des MINCAL 4 ausgenutzt, so sind $8 \times 8 = 64$ Plätze für diese Funktionen reserviert.

2.5) Geschwindigkeit

Ein weiteres, wichtiges Kriterium für die Leistungsfähigkeit eines Rechners ist seine Arbeitsgeschwindigkeit bzw. die Dauer der einzelnen Operationen.

Als Maß hierfür dient die "Wortzeit", d.h. die Zeit für Lesen, Verarbeiten und Zurückschreiben eines Wortes, wobei unter Verarbeitung eine Addition bzw. ein Transportvorgang zu verstehen ist. Jede durch einen bestimmten Programmbebefhl gekennzeichnete Operation setzt sich aus einer Reihe von Wortzeiten zusammen.

Die Wortzeit des MINCAL 4 beträgt 20 Mikrosekunden. In dieser Zeit wird z.B. ein 24-bit-Wort aus dem Kernspeicher gelesen, zu einem anderen Wort addiert und in den Kernspeicher zurückgeschrieben. Das Lesen und Schreiben eines Wortes erfolgt parallel, d.h. alle Binärstellen werden zugleich übertragen; dagegen wird das gelesene Wort bit für bit (seriell) verarbeitet. Diese Anordnung ergibt ein Optimum von Aufwand und Arbeitsgeschwindigkeit; sie bietet sich vor allem als Folge der beim MINCAL 4 vorliegenden Kombination eines langsameren, temperaturstabilen Kernspeichers mit sehr schnellen integrierten Schaltkreisen an.

Die Operationszeiten liegen zwischen 60 μ s (z.B. Addition) und 1600 μ s (Division). Aus dieser Sicht ist der MINCAL 4 als mittelschneller Rechner nach dem heutigen Stand der Technik einzustufen.

Allerdings darf man dabei nicht außer Acht lassen, daß die reelle Arbeitsgeschwindigkeit einer Datenverarbeitungsanlage nicht allein von der Zeittdauer interner Abläufe bestimmt wird, sondern auch von Geschwindigkeit der Peripheriegeräte und des Datenaustauschs mit der Umwelt. Dies gilt in besonderem Maße für Anwendungen im technischen Bereich, bei Aufgaben der direkten Meßwertverarbeitung und der Prozeßsteuerung. Wenn man berücksichtigt, daß z.B. die Aufnahme eines Meßwerts ca. 2 ms, das Lesen einer 6-stelligen Zahl mit anschließendem Zwischenraumzeichen aus einem Lochstreifen ca. 100 ms, ihre Registrierung auf einer Schreibmaschine ca. 500 ms und auf einem Fernschreiber ca. 1000 ms dauert, so bedeuten diese Zeiten das 100-, 5000-, 25000- bzw. 50000fache einer Wortzeit von 20 μ s.

Will man die reelle Arbeitsgeschwindigkeit erhöhen, so muß man dafür sorgen, daß alle langsamten Eingangs- und Ausgabevorrichtungen unabhängig voneinander und vom eigentlichen Rechner arbeiten und doch so lückenlos wie möglich vom Rechner bedient werden. Durch Anwendung eines Systems von Puffern und durch die eingegebene Prioritätslogik kommt der MINCAL 4 dieser Forderung entgegen.

2.6) Parallel-Ein/Ausgang und Puffersystem

Der Digitalrechner MINCAL 4 ist in der Lage, Daten von außen aufzunehmen bzw. nach außen abzugeben, und zwar wortparallel über je einen 25-bit-Kanal, wobei ein 15-bit-Kanal den Geber bzw. Empfänger der Information identifiziert ("Adresse"). Bei synchronem Datenaustausch stehen 20 μ s für Aufnahme oder Abgabe eines Wortes zur Verfügung; wenn die peripheren Einrichtungen dieser Geschwindigkeit nicht gewachsen sind, kann der Rechner bis zur vollständigen Übernahme bzw. Bereitstellung eines Wortes anhalten.

Um jedoch gleichzeitiges Arbeiten mit Periphergeräten, ohne Anhalten des Programms, zu ermöglichen, ist im MINCAL 4 zusätzlich ein nach Bedarf ausgebauter System von Puffern vorgesehen. Sie nehmen eine Information, d.h. ein Wort von 24 bit plus Vorzeichen, sowie eine Adresse von 12 bit Länge innerhalb einer Wortzeit auf und geben sie an das durch die Adresse ausgewählte Gerät bzw. in einer durch sie bestimmten Weise weiter, während das Programm weiterläuft. Umgekehrt kann ein Puffer, durch das Programm angestoßen, Daten von außen holen, speichern und nach vollständiger Übernahme an den Rechner abgeben.

Die Grundausführung des MINCAL 4 enthält stets einen Puffer; Plätze für 7 weitere Puffer sind vorbereitet. Je nach Art der angeschlossenen Periphergeräte sind die Puffer in der Ausführung unterschiedlich. Ein Puffer vom R-Typ gestattet den Anschluß von maximal 8 seriellen Registratur- und 8 seriellen Eingabegeräten, die durch einen Teil der Adresse ausgewählt werden; die restlichen Stellen der Adresse bestimmen das Format der Ein- und Ausgabe. Eine andere Ausführung (D-Typ) gestattet die serielle Ein- und Ausgabe digitaler, dezimaler Informationen und Ausgabe von Schaltbefehlen freiprogrammierbarer Dauer; es können deren bis zu 4096 Informationsgeber bzw. -empfänger angeschlossen werden. Eine dritte Puffer-Ausführung (A-Typ) arbeitet z.B. in Verbindung mit einem Analog-Digital-Umsetzer und Meßstellenumschaltern, mit deren Hilfe man maximal 4096 analoge Meßstellen erfassen kann.

Jeder Puffer ist durch eine Adresse gekennzeichnet. Nach vollständiger Abgabe bzw. Aufnahme einer Information kann er eine Programmunterbrechung auslösen, um so schnell wie möglich wieder bedient zu werden.

2.7) Periphergeräte

Zur Grundausrüstung des MINCAL 4 gehört eine elektrische Eingabe-Ausgabe-Schreibmaschine (IBM 735), die ca. 15 Zeichen je Sekunde druckt, sowie ein 8-Kanal-Streifenleser und ein 8-Kanal-Streifenlocher mit ca. 75 Zeichen je Sekunde Lese- bzw. Lochgeschwindigkeit. Diese für die Programmierung des Rechners und für die Daten-Ein- und -Ausgabe unbedingt notwendigen Geräte sind - über Anpassungseinheiten - an den ersten Puffer angeschlossen.

Weitere Registratur- und Eingabegeräte (Schreibmaschinen, Fernschreiber, Locher, Leser) können - unter Zwischenschaltung entsprechender Anpassungseinheiten - an den gleichen Puffer angeschlossen werden, wobei sie vom Programm zu getrennten Zeiten oder gleichzeitig (dann aber mit den gleichen Daten und im gleichen Format) arbeiten. Ist eine gleichzeitige, aber unabhängige Betriebsweise erwünscht, so ist der Anschluß an getrennte Puffer vom R-Typ notwendig.

Zu den Periphergeräten, die auf Wunsch erhältlich sind, zählt ein binärer Analog-Digital-Umsetzer mit 4 V Meßbereich und einer Auflösung von 2^{-12} , der anstehende analoge Meßwerte in ca. 2 ms digitalisiert. Andere Meßbereiche und spezielle Filter, die u.U. längere Meßzeiten erfordern, können vorgesehen und vom Programm angewählt werden.

Anpassungseinheiten für andere Periphergeräte (z.B. Lochkartengeräte) und Datenanschlüsse (z.B. Analogausgang) sind in Vorbereitung bzw. werden nach Bedarf geliefert.

2.8) Das Prioritätssystem

Zeitsparende Programme in Verbindung mit Periphergeräten, die über das Puffersystem des MINCAL 4 angeschlossen sind, erfordern die Möglichkeit von Programmunterbrechungen. Das gleiche gilt für den häufig vorkommenden Fall, daß während eines laufenden Programms dem Rechner von außen Daten angeboten werden, die sofort übernommen werden müssen, oder ganz allgemein dann, wenn mehrere Programme, die langsamere Periphergeräte benutzen, gleichzeitig laufen.

Das Prioritätssystem des MINCAL 4 sieht 8 Programm niveaus vor, in denen gearbeitet werden kann; sie werden von außen über 8 Startleitungen (oder auch vom Programm her) erregt. Wird ein höheres Niveau gestartet als das gerade laufende, so wechselt das Programm nach Ablauf der jeweiligen Operation, um nach beendetem Vorrangprogramm das unterbrochene Programm an der gleichen Stelle fortzusetzen.

Prioritätsstarts werden z.B. von den Puffern ausgelöst, wenn eine Datenabfrage oder -ausgabe beendet ist, aber auch von außerhalb, wenn ein Informationsaustausch mit externen Datenträgern erwünscht ist. Jeder Programmbeginn und jeder Wiederauflauf eines angehaltenen Programms wird ebenfalls durch einen Prioritäts-Start bewirkt.

In seiner Struktur ist der MINCAL 4 für das Arbeiten mit Vorrangprogrammen konzipiert. Die Anzahl der Steuer-, und Rechenwerke ist auf ein Minimum reduziert, und da am Ende einer Operation deren Inhalt (Programm-Stand und Dateninhalt) ohnehin in festen Plätzen des Kernspeichers enthalten ist, erfolgt der Niveauwechsel unverzögert, d.h. sofort mit der nächsten Operation des Vorrangprogramms.

Für jedes Programm niveau ist ein getrennter Satz von Plätzen mit fester Funktion vorgesehen (Startadresse, 4 Indexregister, Programmstand, Arbeits- und Hilfsspeicher); der Rechner bedient sich automatisch der Register des jeweiligen Niveaus.

Falls mehrere Puffer oder externe Einrichtungen ein gemeinsames Programm niveau starten, muß der Rechner die Herkunft des Starts ermitteln; dies geschieht über Bedingungseingänge (7 gemeinsame bzw. weitere 7 je Niveau, die nur in der betreffenden Vorrangebene wirksam sind). Die Bedingungseingänge, die auf Programmverzweigungen wirken, bilden auch ein bequemes Hilfsmittel zur Modifikation von Verarbeitungsabläufen.

2.9) Programmbefehle

Der Digitalrechner MINCAL 4 besitzt - unabhängig vom Ausbauzustand - stets eine Operationslogik für 60 verschiedene Befehle.

Sie umfassen Vorzeichenbehandlung, Schiften um eine angegebene Zahl von Binärstellen in beiden Richtungen und Linksschiften zwecks Normierung; Transportoperationen zwischen den festen, niveauzugeordneten Registern und beliebigen Plätzen im Kernspeicher; Addition, Subtraktion und logische Operationen; Multiplikation, Division sowie spezielle Befehle zur Umwandlung von Dezimal- in Binärzahlen und umgekehrt; ferner Programmverzweigungen, die entweder unbedingt oder abhängig von Bedingungseingängen, von Vorzeichen oder vom Inhalt der festen Register ablaufen; Verzweigungen in Unterprogramm, Prioritätswechsel, Anhalten und Programmende; außerdem Ein-Ausgabebefehle in Verbindung mit externen Kreisen, insbesondere den Puffern.

Bei allen numerischen, logischen und Transportoperationen kann die Adresse über eines der 4 Indexregister (additiv) verändert werden; anstelle der Adresse kann man einen Festwert von 15 Binärstellen Länge angeben. Bei Transportoperationen ist indirekte Adressierung möglich, d.h. es wird ein Speicherplatz benutzt, dessen Adresse in dem Platz steht, der in der Instruktion angegeben ist.

Die Gruppe der Ein/Ausgabebefehle unterscheidet zwischen der Auslösung von Eingabe- und Ausgabevorgängen und der Übernahme des Pufferinhalts; Anhalten oder Weiterlaufen des Programms sowie die Prioritätsstart-Berechtigung ist programmierbar. In der Adresse ist die Auswahl der Puffereinheit sowie des angeschlossenen Datengebers bzw. -empfängers enthalten; sie ist über Indexregister modifizierbar.

In Verbindung mit Puffern vom R-Typ enthält die Adresse alle Hinweise für die Geräteauswahl und das Ein/Ausgabeformat (Vorzeichen, Zeichendarstellung, Stellenzahl und Endzeichen). Diese Art von Instruktionen erleichtert die Programmierung außerordentlich; mit einem einzigen Befehl kann ein Wort in beliebigem Format aufgenommen bzw. ausgegeben werden.

2.10) Programmierhilfen und Testmöglichkeiten

Trotz ihres übersichtlichen Aufbaus ist die binäre Instruktionsform ("Maschinensprache") des MINCAL 4 nicht das geeignete Mittel, um längere Programme bequem, schnell und frei von Irrtümern zu formulieren. Daher wurde für den MINCAL 4 eine "symbolische Sprache" mit mnemotechnischen Bezeichnungen für die einzelnen Befehle ausgearbeitet.

Jedem Benutzer des MINCAL 4 steht ein Umwandlungsprogramm zur Verfügung, das ein symbolisch geschriebenes Programm in Maschinensprache übersetzt. Dieses Programm übersetzt nicht nur die Befehle; es ordnet auch Daten- und Verzweigungsadressen, Festwertdefinitionen und Variablenplätze, die als "Merkmale" programmiert werden, automatisch zu und erleichtert dadurch die Programmierarbeit weiter.

Das endgültige Programm liegt in Form eines 8-Kanal-Lochstreifens vor, der die Instruktionen, Festwerte und Variablenplätze in der Reihenfolge der Kernspeicheradressen enthält (je Wort 4 Zeichen und ein Vorzeichen bzw. Zwischenraum). Dieser Programmstreifen wird eingelesen und der Kernspeicher mit seinem Inhalt geladen.

Der MINCAL 4 enthält fest verdrahtete Steuerfunktionen zum Einlesen, Auslochen, Ausdrucken und manuellen Eingeben von Programmen sowie zum manuellen Erstellen, Duplizieren und Übersetzen von Programmstreifen, wobei Schreibmaschine, 8-Kanal-Locher und 8-Kanal-Leser benutzt werden. Eingabe und Registrierung auf der Schreibmaschine erfolgt oktal, d.h. es werden jeweils 3 bits eines Wortes zusammengefaßt und mit Hilfe der Ziffern 0...7 ausgedrückt. Diese Darstellung entspricht exakt dem Aufbau der Programmstruktionen und ist daher leicht zu interpretieren.

Alle Bedienungs- und Anzeigeelemente für Programmieren und Testen sind in der Frontplatte der Zentraleinheit enthalten: Anzeigen für den Inhalt der Werke, Anzeigen und Bedienungselemente für das Prioritätssystem, die Puffer, die Bedingungseingänge und Prüfschaltungen; ferner die Bedienung für den Programmierbetrieb, für Einzelanwahl beliebiger Speicherplätze sowie der festen Register; Adreßstop, Vorwahl- und Starttasten für Einzelschrittbetrieb usw.

Jedem Benutzer steht ein Satz diagnostischer Programme zur Verfügung, mit deren Hilfe die einzelnen Funktionsgruppen des MINCAL 4, die Befehlsabläufe usw. getestet werden können.

2.11) Technologie

Der Digitalrechner MINCAL 4 ist vorwiegend aus monolithischen integrierten Schaltkreisen aufgebaut, die sich durch kompakte Bauweise, kurze Schaltzeiten und hohe Zuverlässigkeit auszeichnen. Bei der Anpassung an die Peripherie werden, wo der Leistungspegel dies erfordert, Schaltungen aus diskreten Silizium-Halbleitern verwendet.

Der Kernspeicher des MINCAL 4 enthält Ferritkerne mit weitem Arbeitstemperaturbereich.

Alle Schaltungen sind auf steckbaren, gedruckten Platten aus Epoxydharz-Glasgewebe angeordnet; Rechenkörper, Stromversorgung, Kernspeichereinheiten und periphere Baugruppen sind als 19"-Einschübe auf Teleskopschienen in einem Schrank enthalten.

3. Die Struktur des MINCAL 4

3.1) Die zentrale Recheneinheit

Der MINCAL 4 enthält 2 Werke mit je 24 Flipflops, die als Schieberegister ausgeführt sind. Das C-Werk übernimmt wortweise (bit-parallel) die Informationen aus dem Kernspeicher und gibt sie in gleicher Weise in den Speicher zurück. Die Adresse des Kernspeicherplatzes wird von 15 Binärstellen des A-Werks bestimmt, das im übrigen auch zum zwischenzeitlichen Aufheben von Daten benutzt wird. Die internen Verarbeitungsoperationen bestehen aus einem fortlaufenden Austausch von Daten zwischen Kernspeicher, C- und A-Werk, wobei die Übertragung von C nach A und zurück sowie der Werke in sich selbst seriell erfolgt und von einer Verknüpfungslogik bestimmt wird.

Zu Anfang jeder Operation werden Befehl und Ergänzung mit 9 bits in das B-Werk übertragen, wo sie bis zum Ende der Operation verbleiben.

Die Stellen 1 bis 24 der Werke entsprechen den Binärstellen 2^{-1} bis 2^{-24} ; für Vorzeichen- und Prüfbetrag sind dem C-Werk zwei Flipflops zugeordnet, die jedoch nicht Bestandteile des Schieberegisters sind.

3.2) Die festen Register

Die 3 Werke A, B, und C haben nur während der Dauer einer Operation Bedeutung, wobei der Ablauf von ihrem Inhalt und ihrer Verknüpfung abhängt. Am Ende der Operation sind alle für den weiteren Programmverlauf wichtigen Informationen in festen Plätzen des Kernspeichers enthalten, die bei der nächsten Operation nach Bedarf erneut abgefragt werden. Diese "festen" Register sind vom Standpunkt der Programmierung allein interessant. Je Niveau sind folgende 8 Register vorgesehen:

| Adresse | symbol. Bez. | Funktion |
|---------|--------------|-----------------|
| Ø | Ø | Startadresse |
| 1 | 1 | Indexregister 1 |
| 2 | 2 | Indexregister 2 |
| 3 | 3 | Indexregister 3 |
| 4 | 4 | Indexregister 4 |
| 5 | N | Programmstand |
| 6 | W | Arbeitsregister |
| 7 | X | Hilfsregister |

Im Platz Ø steht die Adresse der ersten Instruktion eines Programms, während Register N die Adresse der jeweils folgenden Instruktion enthält. Das Arbeitsregister W nimmt das Ergebnis der jeweiligen Operation auf; eventuelle Nebenergebnisse stehen im Hilfsregister X. Die 4 Indexregister dienen zur additiven Adressmodifikation.

Der Rechner MINCAL 4 arbeitet stets mit dem Satz von Registern, die durch das jeweilige Programm-Niveau bestimmt werden. Die ersten 64 Adressen des Kernspeichers sind mit den festen Registern belegt, wobei (in oktaler Schreibweise) die "1er-Adresse" die Art des Registers (siehe oben), die "10er-Adresse" das Niveau bezeichnen, also:

| | |
|------------------|-----------------------|
| Adressen ØØ...Ø7 | Register zum Niveau Ø |
| Adressen 1Ø...17 | Register zum Niveau 1 |
| ... | ... |
| Adressen 7Ø...77 | Register zum Niveau 7 |

Niveau 7 ist dasjenige mit der höchsten Prioritätsstufe.

Jedes dieser Register ist über seine Adresse erreichbar, wie jeder andere Speicherplatz auch. Bei einer Reihe von Befehlen kann ein Register (des betreffenden Niveaus) jedoch auch speziell angesprochen werden, und zwar durch Einsetzen der 1er-Adresse in die Ergänzungsstelle der Instruktion. Im Übrigen werden insbesondere die Register \emptyset , N, W und X von der Operationslogik selbsttätig angewählt.

3.3) Die Prioritätslogik

Jeder Programmablauf wird beim MINCAL 4 durch einen Prioritätsstart begonnen, d.h. einer der 8 Starteingänge, die den 8 Niveaus zugeordnet sind, erhält ein Signal. Läuft kein Programm oder keines mit höherem Rang, so wird das gestartete Programm sofort begonnen; andernfalls wird der Start gespeichert, bis kein Programm mit höherem Rang mehr läuft.

Der Programmablauf in einem Niveau wird unterbrochen

durch Start eines höheren Niveaus, wobei die laufende Operation noch zu Ende geführt wird; bei Anhalten oder Ende des Vorrangprogramms läuft das unterbrochene Programm mit der nächsten Instruktion automatisch weiter,

durch einen Halt-Befehl im betreffenden Programm; mit einem neuen Start wird das Programm von der unterbrochenen Stelle an weiterverfolgt,

durch einen End-Befehl; der nächste Start bewirkt, daß das Programm von vorn beginnt, d.h. mit der im Register \emptyset ausgegebenen Instruktionsadresse.

Statt über einen Starteingang, kann jedes Programm auch durch einen Befehl ("Prioritätswechsel"), der in einem anderen Programmniveau liegt, gestartet werden.

3.4) Die Verbindung mit den Puffern

Der Datenaustausch mit Periphergeräten erfolgt beim MINCAL 4 vorzugsweise über Puffer, die Informationen entweder vom zentralen Rechenteil übernehmen und nach außen weitergeben oder von der Peripherie aufnehmen und dem Rechner übermitteln.

Es sind maximal 8 Puffer vorgesehen, die durch die 3 höchsten Stellen der 15-bit-Adresse ausgewählt werden.

Der Dateninhalt der Puffer wird seriell in das C-Werk bzw. in umgekehrter Richtung übertragen. Der Adreßteil nimmt seriell die letzten 12 Binärstellen des A-Werks auf (die programmierte Adresse); dadurch wird das Periphergerät selektiert bzw. das Ausgabeformat bestimmt.

Jeder Puffer besitzt einen Startausgang, der nach Belieben mit einem der Prioritätsstart-Eingänge verbunden wird und am Ende eines Abfrage- bzw. Ausgabezyklus' das betreffende Niveau erregt (falls der auslösende Befehl mit Startberechtigung programmiert war). Für den Fall, der mehrfachen Benutzung eines Starteingangs wird eine zweite Ausgangsleitung mit einem Bedingungseingang verbunden, um den Puffer zu identifizieren.

3.5) Parallel-Ein/Ausgang

Zur Übertragung von Informationen in schnelle externe Geräte bzw. weitere, parallel arbeitende Puffer und aus diesen zurück in den Rechner kann der Parallel-Ein/Ausgang benutzt werden.

Der Eingabe- und der Ausgabe-Parallelkanal (mit je 25 Anschlüssen) wirken auf die 24 Binärstellen und die Vorzeichenstelle des C-Werks. Zur Identifikation des Datengebers bzw. -empfängers dient ein Adreßkanal (mit 12 Anschlüssen), der an die 12 letzten Stellen des A-Werks angeschlossen ist, sowie ein Gruppenkanal (mit 8 Anschlüssen), den die 3 höheren Stellen des A-Werks steuern.

Die Befehle für den Parallel-Ein/Ausgang entsprechen in ihrem Aufbau den Operationen für Ein/Ausgabe über die eingebauten Puffer.

4. Die Programmfunctionen des MINICAL 4

4.1) Darstellung von Daten

Ein 24-bit-Wort des MINICAL 4 kann eine 24-stellige Binärzahl aufnehmen; ein zusätzliches bit kennzeichnet das Vorzeichen:

| | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 2^{-1} | | | | | | | | | | | | | | | | | | | | | | | | |

2-24

In dieser Form müssen numerische Werte stets dann vorliegen, wenn man sie addieren, subtrahieren, multiplizieren oder dividieren will. Bei Multiplikation und Division ist darauf zu achten, daß das Komma links von der höchsten Binärstelle (2^{-1}) steht.

Dezimalzahlen, die vor allem beim Datenaustausch mit der Peripherie des Rechners auftreten, sind 6-stellig im Code 8-4-2-1 (BCD-Code) in einem 24-bit-Wort enthalten:

| | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|-----------|---|---|---|-----------|---|----|----|-----------|----|----|----|-----------|----|----|----|-----------|----|----|----|----|
| - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| 8 | 4 | 2 | 1 | 8 | 4 | 2 | 1 | 8 | 4 | 2 | 1 | 8 | 4 | 2 | 1 | 8 | 4 | 2 | 1 | 8 | 4 | 2 | 1 | |
| 10^{-1} | | | | 10^{-2} | | | | 10^{-3} | | | | 10^{-4} | | | | 10^{-5} | | | | 10^{-6} | | | | |

(10^5) (10^4) (10^3) (10^2) (10^1) (10^0)

Zur numerischen Verarbeitung müssen Dezimalzahlen in Binärzahlen umgeformt werden; dafür stehen Umwandlungsbefehle zur Verfügung, und zwar entweder für Linkskomma- (Bruch-) Darstellung oder mit Rechtskomma (als ganze Zahl).

Eine dritte Möglichkeit sind 4 alphanumerische Zeichen zu je 6 bit, die in einem Wort Platz finden:

| | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| - | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
| B | A | 8 | 4 | 2 | 1 | B | A | 8 | 4 | 2 | 1 | B | A | 8 | 4 | 2 | 1 | B | A | 8 | 4 | 2 | 1 | |

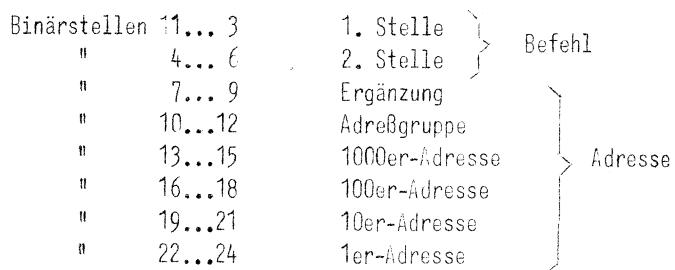
1. 2. 3. 4. Zeichen

Die Darstellung folgt dem 6-bit-Code; die Stellen B-A-8-4-2-1 entsprechen den Spuren 7-6-4-3-2-1 auf dem 8-Kanal-Lochstreifen. Mit diesem Code lassen sich insgesamt 63 verschiedene Charaktere (10 Ziffern 0...9, 26 Buchstaben A...Z und 27 Symbole) sowie das Zeichen "Zwischenraum" darstellen.

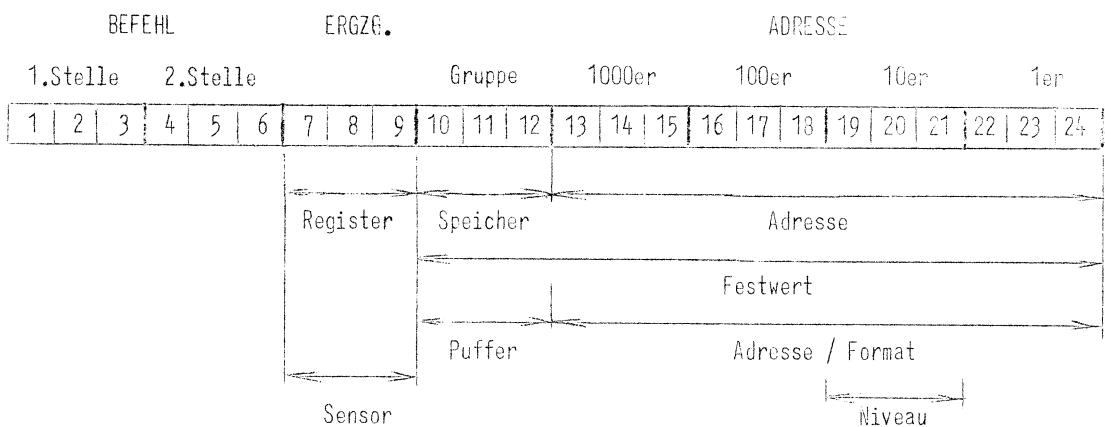
4.2) Darstellung von Instruktionen

Eine Programminstruktion nimmt stets ein 24-bit-Wort in Anspruch. Das Wort wird dabei in 8 Stellen zu je 3 bit unterteilt aufgefaßt; eine in "Maschinencode" geschriebene Instruktion besteht daher aus einer 8-stelligen Oktalzahl (jede Stelle kann die Ziffern 0...7 enthalten).

Die Instruktion ist wie folgt gegliedert:

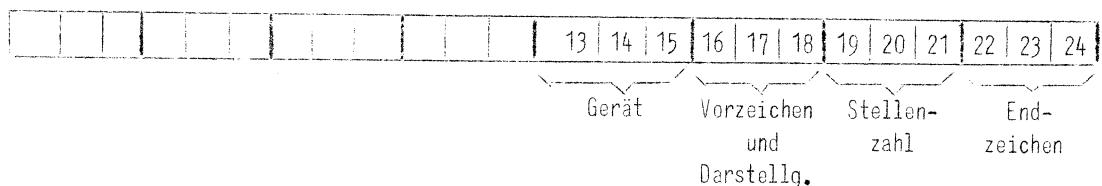


Je nach Befehl kann den einzelnen Stellen einer Instruktion folgende Bedeutung zukommen:



4.3) Adressen

Kernspeicher- und externe Adressen umfassen 15 bit und werden im Maschinencode als 5-stellige Oktalzahlen geschrieben. Die erste Oktalstelle bezeichnet die Adreßgruppe (die Kernspeichereinheit mit 4096 Daten bzw. den Puffer), die 4 weiteren die eigentliche Adresse. Bei Puffern vom R-Typ haben diese Stellen die Bedeutung von Auswahl- und Formatbefehlen:



Die maximal 32768 Kernspeicherplätze sind in oktaler Schreibweise mit 00000...77777 nummeriert. Die ersten 64 Adressen (00000 ... 00777) sind für 8 x 8 feste Register reserviert (s. 3.2).

4.4) Indirekte Adressierung

Während bei logischen, numerischen und Ein/Ausgabebefehlen nur direkte Adressierung möglich ist (der Operand befindet sich in der angegebenen Adresse), ist bei Transportoperationen (ins bzw. aus dem Arbeitsregister W) neben der direkten auch indirekte Adressierung möglich; der Operand wird aus einem Platz geholt bzw. in einen Platz gebracht, dessen Adresse in dem Platz steht, der in der Instruktion angegeben ist.

4.5) Adreßmodifikation

Bei allen Transport-, numerischen, logischen und Ein/Ausgabebefehlen kann die Adresse über eines der 4 Indexregister modifiziert werden, indem dessen Inhalt zur Adresse addiert wird. Die Nummer des Registers wird in der Ergänzung angegeben.

Bei indirekter Adressierung wird die indirekte Adresse modifiziert, nicht die angegebene.

4.6) Festwertoperation

Bei numerischen, logischen und Ladebefehlen kann statt der Adresse ein 15-bit-Festwert programmiert werden ($2^{10} \dots 2^{24}$), der als Operand aufgefaßt wird. Die Festwertoperation ist auf alle 8 festen Register anwendbar (außer bei Multiplikation und Division, wo sie sich stets auf das Arbeitsregister W bezieht). Das Register wird in der Ergänzung angegeben.

4.7) Datentausch innerhalb der festen Register

Besondere Operationen erlauben die Übertragung des W-Register-Inhalts in jedes der 7 anderen festen Register und zurück.

4.8) Verzweigungen

Programmverzweigungen bewirken, daß der Programmablauf nicht mit der nächsten Instruktion fortgesetzt wird, sondern an einer beliebigen anderen Stelle; im MINCAL 4 wird hierzu das Register N auf einen neuen Wert gesetzt (Verzweigungsadresse). Bei der absoluten Verzweigung wird dieser Wert in den Adreßstellen angegeben; die relative Verzweigung dagegen findet die neue Adresse in einem Platz, dessen Adresse programmiert wird.

Verzweigungen finden entweder unbedingt statt oder nur dann, wenn der programmierte Bedingungseingang (Sensor) erregt ist. Eine Reihe von Verzweigungsbefehlen ist inhaltsbedingt; der Programmsprung findet nur statt, wenn das in der Ergänzungsstelle angegebene Register Nullinhalt, positiven oder negativen Inhalt hat.

4.9) Unterprogramme

In Unterprogramme kann absolut oder relativ verzweigt werden; das Aufrufen eines Unterprogramms ist unbedingt oder von Bedingungseingängen abhängig programmierbar.

Beim Unterprogrammsprung wird die um 1 erhöhte Instruktions-Nummer des Befehls in den adressierten Platz übertragen (Rückkehradresse), und das Programm läuft mit dem nächstfolgenden Platz weiter. Am Ende des Unterprogramms ist ein relativer Sprung vorzusehen, der zum Hauptprogramm zurückführt.

5. Die Programmbefehle des MINCAL 4

Die 60 Befehle des MINCAL 4 sind in 8 Gruppen unterteilt, entsprechend der Ziffer in der ersten Oktalstelle der Instruktion:

| Gruppe # | |
|----------|--|
| 0 | Nulloperation, Vorzeichen, Schiften |
| 1 | Transportoperationen |
| 2 | Einfache numerische und logische Operationen |
| 3 | Umfangreiche numerische Operationen |
| 4 | Verzweigung |
| 5 | Unterprogramm, Prioritätswechsel, Halt, Ende |
| 6 | Ein/Ausgabe über Puffer |
| 7 | Ein/Ausgabe über Parallelkanal |

Im folgenden sind die einzelnen Programmbefehle mit oktaler und symbolischer Bezeichnung, deren Erklärung in deutsch und englisch (letztere liegt der symbolischen Bezeichnung meist zugrunde) und Angaben über Verlauf und Dauer der Operation angegeben:

| | | |
|----|-----|---|
| 00 | NOP | <u>Keine Operation</u> No Operation Diese Instruktion wird übersprungen. Dauer: 40 µs |
| 01 | MSP | <u>Vorzeichen auf Plus</u> Make Sign Plus Das Vorzeichen des W-Registers wird auf Plus gesetzt. Dauer: 60 µs |
| 02 | MSM | <u>Vorzeichen auf Minus</u> Make Sign Minus Das Vorzeichen des W-Registers wird auf Minus gesetzt. Dauer: 60 µs |
| 03 | MCS | <u>Vorzeichen wechseln</u> Make Change of Sign Das Vorzeichen des W-Registers wird in sein Gegenteil verkehrt. Dauer: 60 µs |
| 04 | SR | <u>Schiften rechts</u> Shift Right Der Inhalt des W-Registers wird um die angegebene Zahl von Binärstellen nach rechts verschoben. Der rechts überlaufende Teil steht linksbündig im X-Register. Dauer: 100 µs |
| 05 | SL | <u>Schiften links</u> Shift Left Der Inhalt des W-Registers wird um die angegebene Zahl von Binärstellen nach links verschoben. Der links überlaufende Teil steht rechtsbündig im X-Register. Dauer: 100 µs |
| 06 | SRR | <u>Schiften rechts mit Runden</u> Shift Right and Round Der Inhalt des W-Registers wird um die angegebene Zahl von Binärstellen nach rechts verschoben. Zum Absolutbetrag des Inhalts wird vorher eine 1 in der höchsten, rechts herauslaufenden Stelle addiert. Dauer: 100...120 µs |

| | | |
|----|-----|--|
| 07 | SLN | <u>Schiften links zwecks Normierung</u> Shift Left and Normalize Der Inhalt des W-Registers wird so weit nach links verschoben, bis in der höchsten Stelle eine 1 steht. Die Größe der Stellenverschiebung steht rechtsbündig im X-Register. Dauer: 80 µs |
| 10 | LDC | <u>Laden Festwert</u> Load Constant Das angegebene Register wird gelöscht und mit dem angegebenen Festwert geladen. Dauer: 60 µs |
| 11 | LDR | <u>Laden aus Register</u> Load from Register Das W-Register wird gelöscht und mit dem Inhalt des angegebenen Registers geladen. Dauer: 80 µs |
| 12 | LD | <u>Laden</u> Load Das W-Register wird gelöscht und mit dem Inhalt des adressierten Speicherplatzes geladen. Dauer: 80...100 µs |
| 13 | LDY | <u>Laden indirekt</u> Load Indirectly Das W-Register wird gelöscht und mit dem Inhalt eines Platzes geladen, dessen Adresse im angegebenen Speicherplatz steht. Dauer: 120...140 µs |
| 15 | TRR | <u>Transfer in Register</u> Transfer to Register Der Inhalt des W-Registers wird in das angegebene Register übertragen, das vorher gelöscht wird. Dauer: 80 µs |
| 16 | TR | <u>Transfer</u> Transfer Der Inhalt des W-Registers wird in den adressierten Speicherplatz übertragen, der vorher gelöscht wird. Dauer: 80...100 µs |

| | | |
|----|-----|---|
| 17 | TRY | <u>Transfer indirekt</u> Transfer Indirectly |
| | | Der Inhalt des W-Registers wird in einen Platz übertragen, dessen Adresse im angegebenen Speicherplatz steht und der vorher gelöscht wird. |
| | | Dauer: 120...140 µs |
| 20 | ADC | <u>Addieren Festwert</u> Add Constant |
| | | Der angegebene Festwert wird zum Inhalt des angegebenen Registers addiert; die Summe steht im angegebenen Register, ein eventueller Überlauf als rechtsbündige 1 im X-Register. |
| | | Dauer: 60...80 µs |
| 21 | AD | <u>Addieren</u> Add |
| | | Der Inhalt des adressierten Speicherplatzes wird zum Inhalt des W-Registers addiert; die Summe steht im W-Register, ein eventueller Überlauf als rechtsbündige 1 im X-Register. |
| | | Dauer: 80...120 µs |
| 22 | SBC | <u>Subtrahieren Festwert</u> Subtract Constant |
| | | Der angegebene Festwert wird vom Inhalt des angegebenen Registers subtrahiert; die Differenz steht im angegebenen Register, ein eventueller Überlauf als rechtsbündige 1 im X-Register. |
| | | Dauer: 60...80 µs |
| 23 | SB | <u>Subtrahieren</u> Subtract |
| | | Der Inhalt des adressierten Speicherplatzes wird vom Inhalt des W-Registers subtrahiert; die Differenz steht im W-Register, ein eventueller Überlauf als rechtsbündige 1 im X-Register. |
| | | Dauer: 80...120 µs |
| 24 | FOC | <u>Boole'sches Oder mit Festwert</u> Boolean Or Function with Constant |
| | | Alle Stellen des angegebenen Registers werden mit den Stellen des angegebenen Festwertes in Oder-Verknüpfung gebracht; das Ergebnis steht im angegebenen Register. |
| | | Dauer: 60 µs |

| | | |
|----|-----|---|
| 25 | F0 | <u>Boole'sches Oder</u> Boolean Or Function Alle Stellen des W-Registers werden mit den Stellen des adressierten Speicherplatzes in Oder-Verknüpfung gebracht; das Ergebnis steht im W-Register. Dauer: 80...100 μ s |
| 26 | FAC | <u>Boole'sches Und mit Festwert</u> Boolean And Function with Constant Alle Stellen des angegebenen Registers werden mit den Stellen des angegebenen Festwerts in Und-Verknüpfung gebracht; das Ergebnis steht im angegebenen Register. Dauer: 60 μ s |
| 27 | FA | <u>Boole'sches Und</u> Boolean And Function Alle Stellen des W-Registers werden mit den Stellen des adressierten Speicherplatzes in Und-Verknüpfung gebracht; das Ergebnis steht im W-Register. Dauer: 80...100 μ s |
| 30 | MPC | <u>Multiplizieren mit Festwert</u> Multiply by Constant Der Inhalt des W-Registers wird mit dem angegebenen Festwert multipliziert; die bedeutenden Stellen des Produkts stehen im W-Register, die unbedeutenden im X-Register. Dauer: 1040 μ s |
| 31 | MP | <u>Multiplizieren</u> Multiply Der Inhalt des W-Registers wird mit dem Inhalt des adressierten Speicherplatzes multipliziert; die bedeutenden Stellen des Produkts stehen im W-Register, die unbedeutenden im X-Register. Dauer: 1060...1080 μ s |
| 32 | DVC | <u>Dividieren durch Festwert</u> Divide by Constant Der Inhalt des W-Registers wird durch den angegebenen Festwert geteilt; der Quotient steht im W-Register, der Rest im X-Register. Bei Quotienten ≥ 1 bleibt der ursprüngliche Inhalt des W-Registers erhalten, und das Vorzeichen des X-Registers wird Minus gesetzt. Dauer: 1560 μ s |

| | | |
|----|-----|---|
| 33 | DV | <u>Dividieren</u> Divide |
| | | Der Inhalt des W-Registers wird durch den Inhalt des adressierten Speicherplatzes geteilt; der Quotient steht im W-Register, der Rest im X-Register. Bei Quotienten ≥ 1 bleibt der ursprüngliche Inhalt des W-Registers erhalten, und das Vorzeichen des X-Registers wird auf Minus gesetzt. |
| | | Dauer: 1580...1600 μ s |
| 34 | VBL | <u>Binär-Umformung mit Linkskomma</u> Convert to Binary Left-point |
| | | Der dezimale Inhalt des W-Registers wird in eine Binärzahl umgeformt, und zwar als Bruch. |
| | | Dauer: 680...1160 μ s |
| 35 | VBR | <u>Binär-Umformung mit Rechtskomma</u> Convert to Binary Right-point |
| | | Der dezimale Inhalt des W-Registers wird in eine Binärzahl umgeformt, und zwar als ganze Zahl. |
| | | Dauer: 680...1160 μ s |
| 36 | VDL | <u>Dezimal-Umformung mit Linkskomma</u> Convert to Decimal Left-point |
| | | Der binäre Inhalt des W-Registers wird in eine Dezimalzahl umgeformt, und zwar als Bruch. |
| | | Dauer: 1160...1640 μ s |
| 37 | VDR | <u>Dezimal-Umformung mit Rechtskomma</u> Convert to Decimal Right-point |
| | | Der binäre Inhalt des W-Registers wird in eine Dezimalzahl umgeformt, und zwar als ganze Zahl. |
| | | Dauer: 1160...1640 μ s |
| 40 | BA | <u>Verzweigen absolut</u> Branch Absolute |
| | | Das Programm verzweigt auf den angegebenen Platz. Ist eine der Ziffern 1...7 als Ergänzung programmiert, so wird nur dann verzweigt, wenn der betreffende Bedingungseingang erregt ist. |
| | | Dauer: 40...60 μ s |

| | | |
|----|-----|--|
| 41 | BR | <u>Verzweigen relativ</u> Branch Relative |
| | | Das Programm verzweigt auf einen Platz, dessen Adresse im angegebenen Speicherplatz steht. Ist eine der Ziffern 1...7 als Ergänzung programmiert, so wird nur dann verzweigt, wenn der betreffende Bedingungseingang erregt ist. |
| | | Dauer: 40...80 μ s |
| 42 | BAZ | <u>Verzweigen absolut wenn Null</u> Branch Absolute if Zero |
| | | Das Programm verzweigt auf den angegebenen Platz, wenn der Inhalt des angegebenen Registers Null ist. |
| | | Dauer: 60...80 μ s |
| 43 | BRZ | <u>Verzweigen relativ wenn Null</u> Branch Relative if Zero |
| | | Das Programm verzweigt auf einen Platz, dessen Adresse im angegebenen Speicherplatz steht, wenn der Inhalt des angegebenen Registers Null ist. |
| | | Dauer: 60...100 μ s |
| 44 | BAP | <u>Verzweigen absolut wenn Plus</u> Branch Absolute if Plus |
| | | Das Programm verzweigt auf den angegebenen Platz, wenn der Inhalt des angegebenen Registers positiv ist. |
| | | Dauer: 60...80 μ s |
| 45 | BRP | <u>Verzweigen relativ wenn Plus</u> Branch Relative if Plus |
| | | Das Programm verzweigt auf einen Platz, dessen Adresse im angegebenen Speicherplatz steht, wenn der Inhalt des angegebenen Registers positiv ist. |
| | | Dauer: 60...100 μ s |
| 46 | BAM | <u>Verzweigen absolut wenn Minus</u> Branch Absolute if Minus |
| | | Das Programm verzweigt auf den angegebenen Platz, wenn der Inhalt des angegebenen Registers negativ ist. |
| | | Dauer: 60...80 μ s |
| 47 | BRM | <u>Verzweigen relativ wenn Minus</u> Branch Relative if Minus |
| | | Das Programm verzweigt auf einen Platz, dessen Adresse im angegebenen Speicherplatz steht, wenn der Inhalt des angegebenen Registers negativ ist. |
| | | Dauer: 60...100 μ s |

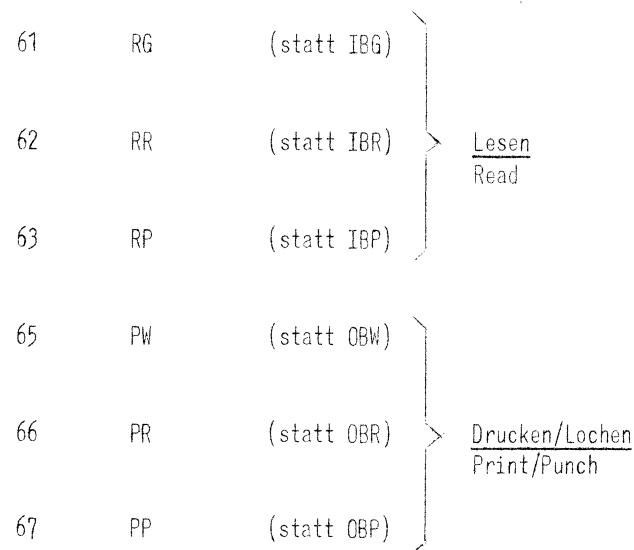
| | | |
|----|-----|---|
| 50 | CA | <u>Aufrufen Unterprogramm absolut</u> Call Subroutine Absolute |
| | | Die um 1 erhöhte Instruktionsadresse wird im angegebenen Platz gespeichert, und das Programm verzweigt zu dem ihm folgenden Platz. Ist eine der Ziffern 1...7 als Ergänzung programmiert, so läuft diese Operation nur dann ab, wenn der entsprechende Bedingungseingang erregt ist. |
| | | Dauer: 40...100 µs |
| 51 | CR | <u>Aufrufen Unterprogramm relativ</u> Call Subroutine Relative |
| | | Die um 1 erhöhte Instruktionsadresse wird in einem Platz gespeichert, dessen Adresse im angegebenen Speicherplatz steht, und das Programm verzweigt zu dem ihm folgenden Platz. Ist eine der Ziffern 1...7 als Ergänzung programmiert, so läuft diese Operation nur dann ab, wenn der entsprechende Bedingungseingang erregt ist. |
| | | Dauer: 40...120 µs |
| 53 | NCP | <u>Normaler Prioritätswechsel</u> Normal Change of Priority |
| | | Das angegebene Programmniveau wird gestartet. |
| | | Dauer: 60 µs |
| 54 | HLT | <u>Halt</u> Halt |
| | | Das Programm hält an; ein neuer Start lässt es von der gleichen Stelle an weiterlaufen. |
| | | Dauer: 60 µs |
| 55 | HCP | <u>Halt mit Prioritätswechsel</u> Halt and Change Priority |
| | | Das angegebene Programmniveau wird gestartet; das Programm hält an; ein neuer Start lässt es von der gleichen Stelle an weiterlaufen. |
| | | Dauer: 80 µs |
| 56 | END | <u>Ende</u> . End |
| | | Das Programm hält an; bei einem neuen Start beginnt es von vorn. |
| | | Dauer: 60 µs |

- 57 ECP Ende mit Prioritätswechsel
 End and Change Priority
Das angegebene Programm niveau wird gestartet, das Programm hält an; bei einem neuen Start beginnt es von vorn.
Dauer: 80 µs
- 60 GB Übernehmen aus Puffer
 Get from Buffer
Der Inhalt des angegebenen Puffers wird ins W-Register übernommen, das vorher gelöscht wurde.
Dauer: 80 µs
- 61 IBG Eingabe mit Übernehmen aus Puffer
 Input by Buffer and Get
Die Adresse wird in den angegebenen Puffer übertragen; der Rechner hält an, bis die externe Information im Puffer ist; dann wird sie ins W-Register übernommen, das vorher gelöscht wurde.
Dauer: 100 µs + Wartezeit
- 62 IBR Eingabe über Puffer ohne Anhalten
 Input by Buffer and Run
Die Adresse wird in den angegebenen Puffer übertragen, das Programm läuft weiter.
Dauer: 60 µs
- 63 IPB Eingabe über Puffer ohne Anhalten mit Prioritätsberechtigung
 Input by Buffer and Enable Priority
Die Adresse wird in den angegebenen Puffer übertragen, das Programm läuft weiter. Wenn die externe Information im Puffer ist, wird ein Prioritätsstart ausgelöst.
Dauer: 60 µs
- 64 OBW Ausgabe über Puffer mit Anhalten
 Output by Buffer and Wait
Die Adresse und der Inhalt des W-Registers werden in den angegebenen Puffer übertragen; der Rechner hält an, bis die Information ausgegeben ist.
Dauer: 80 µs + Wartezeit
- 66 OBR Ausgabe über Puffer ohne Anhalten
 Output by Buffer and Run
Die Adresse und der Inhalt des W-Registers werden in den angegebenen Puffer übertragen; das Programm läuft weiter.
Dauer: 80 µs

| | | |
|----|-----|--|
| 67 | OBP | <u>Ausgabe über Puffer ohne Anhalten mit Prioritätsberechtigung</u> Output by Buffer and Enable Priority Die Adresse und der Inhalt des W-Registers werden in den angegebenen Puffer übertragen; das Programm läuft weiter; wenn die Information ausgegeben ist, wird ein Prioritätsstart ausgelöst. Dauer: 80 µs |
| 70 | GX | <u>Übernehmen extern</u> Get External Der adressierte, externe Geber wird parallel abgefragt und sein Inhalt ins W-Register übertragen, das vorher gelöscht wurde. Dauer: 80 µs |
| 71 | IXG | <u>Eingabe und Übernehmen extern</u> Input External and Get Der externe Geber wird angewählt; der Rechner hält an, bis der Geber die Bereitstellung der Information meldet; sie wird parallel abgefragt und ins W-Register übertragen, das vorher gelöscht wurde. Dauer: 100 µs + Wartezeit |
| 72 | IXR | <u>Eingabe extern ohne Anhalten</u> Input External and Run Die Adresse wird parallel in einen externen Speicher übertragen, der den externen Geber anwählt; das Programm läuft weiter. Dauer: 60 µs |
| 73 | IXP | <u>Eingabe extern ohne Anhalten mit Prioritätsberechtigung</u> Input External and Enable Priority Die Adresse wird parallel in einen externen Speicher übertragen, der den externen Geber anwählt; das Programm läuft weiter. Wenn die externe Information bereitgestellt ist, wird ein Prioritätsstart ausgelöst. Dauer: 60 µs |
| 75 | OXW | <u>Ausgabe extern mit Anhalten</u> Output External and Wait Der Inhalt des W-Registers wird parallel in den adressierten externen Empfänger übertragen; der Rechner hält an, bis die Übernahme der Information gemeldet ist. Dauer: 80 µs + Wartezeit |

| | | |
|----|-----|--|
| 76 | OXR | <u>Ausgabe extern ohne Anhalten</u> Output External and Run Der Inhalt des W-Registers wird parallel in den adressierten, externen Geber übertragen, das Programm läuft weiter. Dauer: 80 µs |
| 77 | OXP | <u>Ausgabe extern ohne Anhalten mit Prioritätsberechtigung</u> Output External and Enable Priority Der Inhalt des W-Registers wird, zugleich mit der Adresse, parallel in einen externen Geber übertragen, der in diesem Falle ein Speicher ist und einen Arbeitszyklus startet. Am Ende des Zyklus' wird ein Prioritätsstart ausgelöst. Dauer: 80 µs |

Ein/Ausgabe-Befehle in Verbindung mit Puffern vom R-Typ (für Registratoren- und Eingabegeräte mit Formatangaben) können symbolisch wie folgt geschrieben werden:



Anstelle oktaler Adressen wird das Format dann ebenfalls symbolisch angegeben. Die Formatangaben enthalten Hinweise für:

das Vorzeichen (ohne, mit)

die Darstellung (oktal, dezimal, dezimal mit Nullunterdrückung, alphanumerisch)

die Stellenzahl (0 bis 8)

das Endzeichen (ohne, Zwischenraum, Tabulator, Zeilenwechsel, Rücklauf, Punkt, Schwarz-, Rot-Umschaltung).

6. Die Ausführungsformen des MINCAL 4

6.1) Grundausführung

Der Digitalrechner MINCAL 4 besteht in seiner einfachsten Form aus folgenden Baugruppen:

Rechenkörper MINCAL 4
19"-Einschub Größe K (Höhe 10 Einheiten = 445 mm)

Stromversorgung
19"-Einschub Größe C (Höhe 3 Einheiten = 133 mm)

Kernspeichereinheit
19"-Einschub Größe B (Höhe 2 Einheiten = 90 mm)

Der Rechenkörper umfaßt sämtliche Funktionsgruppen des Rechners (Taktsteuerung, Operations-, Verknüpfungs-, Bedingungs- und Prioritätslogik, A-, B- und C-Werk usw.). Sie sind auf steckbaren, gedruckten Schaltungen angeordnet, die nach Aufklappen der Frontplatte zugänglich und nach vorn herausziehbar sind. Die Frontplatte enthält alle Bedienungs- und Anzeigeelemente, die für Programmierung, Testen eines Programms und Überprüfung der Rechnerfunktionen wichtig sind. Im Rechenkörper sind verdrahtete Plätze für 8 Puffer vorgesehen. Zur Grundausstattung gehört ein Puffer (mit der Nummer #) einschließlich Steuerlogik und Anpassungsschaltungen für eine Schreibmaschine.

Die Stromversorgung enthält transistor-geregelte Netzteile für die digitalen Schaltungen des Rechners und für den Kernspeicher, ferner Netzteile für Anzeigen, für eine Schreibmaschine usw. Die Stromversorgung ist so ausgelegt, daß alle weiteren Kernspeichereinheiten und Puffer sowie die digitalen Schaltungen für weitere Periphergeräte von ihr versorgt werden können.

Die Kernspeichereinheit besitzt einen Ferritspeicher mit entweder 2048 ("2K") oder 4096 ("4K") Worten Kapazität, jedes Wort zu 26 bit. Die 2K-Ausführung kann nachträglich auf ein Speichervolumen von 4K erweitert werden. Die Einheit ist komplett mit Anwahlschaltungen, Impulstreibern, Inhibit- und Lese-verstärkern ausgerüstet.

6.2) Schreibmaschine

Zur Programmierung des Rechners ist eine Ein/Ausgabe-Schreibmaschine unerlässlich. Der MINCAL 4 benutzt den Typ IBM 735 BCD (Kugelkopfmodell mit leicht modifizierter Verdrahtung). Die Schreibgeschwindigkeit beträgt ca. 15 Z/s; in einer Zeile finden maximal 130 Zeichen Platz. 63 verschiedene Charaktere können eingetastet und gedruckt werden; die Funktionen Zwischenraum, Tabulator, Zeilenwechsel, Schreibkopfrücklauf (mit Zeilenwechsel) und Farbbandumschaltung sind vom Programm ansteuerbar.

Die Schreibmaschine wird direkt an den Rechenkörper angeschlossen, der die Anpassungsschaltung in Verbindung mit dem ersten Puffer und dessen Steuerlogik enthält.

Außer für die Programmierung ist die Schreibmaschine als wichtiges Daten-Ein- und vor allem -Ausgabegerät zu verwenden; da sie mit dem Rechner über einen Puffer verbunden ist, arbeitet sie unabhängig vom Programmverlauf, d.h. Drucken und Rechnen können gleichzeitig erfolgen.

6.3) Lochstreifeneinheit

Der MINCAL 4 wird über 8-Kanal-Lochstreifen programmiert. Als Standardgerät ist eine Lochstreifeneinheit vorgesehen, die einen 8-Kanal-Locher und einen 8-Kanal-Leser enthält, jeweils mit Auf- und Abwickel- bzw. Vorratsspule. Die Lese- bzw. Lochgeschwindigkeit beträgt 75 Z/s. Da Programme alphanumerisch mit Vorzeichen gelocht bzw. gelesen werden, sind je Wort 5 Zeichen erforderlich; Lochen bzw. Lesen eines Programms von 4K Länge dauert viereinhalb Minuten.

Die Lochstreifeneinheit ist als 19"-Einschub (Höhe 10 Einheiten = 445 mm) ausgeführt. Sie enthält die Anpassungsschaltungen für Locher und Leser sowie ein Netzteil für deren Stromversorgung und wird direkt mit dem Rechenkörper, d.h. mit der Steuerlogik des ersten Puffers, verbunden.

Die Lochstreifen werden im üblichen BCD-Code gelocht und gelesen; der Locher kann auf den (geringfügig abweichenden) IBM-8-Kanal-Code umgeschaltet werden.

Auch die Lochstreifeneinheit kann - außer zum Programmieren - für den laufenden Betrieb als Eingabe- und Ausgabemedium benutzt werden.

Auf Wunsch können statt der Lochstreifeneinheit andere, noch schnellere Streifenlocher bzw. -leser angeschlossen werden. Im allgemeinen ist dann eine spezielle, jedoch nicht sehr umfangreiche Anpassungsschaltung erforderlich.

6.4) Erweiterung der Speicherkapazität

Der MINCAL 4 kann auf maximal 8 Kernspeichereinheiten erweitert werden; die 7 weiteren Einheiten gleichen der in der Grundausrüstung vorhandenen und besitzen jeweils eine Kapazität von 4096 (4K) Worten; auf Wunsch kann auch hier eine (nachträglich erweiterbare) 2K-Ausführung vorgesehen werden.

Die maximale Kapazität beträgt somit 32768 Worte, die sämtlich direkt adressierbar sind.

6.5) Puffer vom R-Typ

Im Rechenkörper des MINCAL 4 können maximal 8 vom Programm einzeln anwählbare Puffer verschiedener Art untergebracht werden.

Der R-Typ-Puffer ist - in Verbindung mit einer Steuerlogik - für Registriergeräte und serielle Dateneingaben gedacht und durch folgende Eigenschaften gekennzeichnet:

Eingabe aus 8 verschiedenen Geräten im Start-Stop-Betrieb,

Ausgabe in 8 verschiedene Geräte bzw. 8 Gerätekombinationen im Start-Stop-Betrieb,

Ein/Ausgabe stellen-seriell, bit-parallel über jeweils 8 Datenleitungen (BCD-Code), sowie (bei der Ausgabe) über 7 Funktionsleitungen,

Format - ohne und mit Vorzeichen, Darstellung (oktal, dezimal, alphanumerisch) und Anzahl der Stellen (0 bis 8), ohne und mit verschiedenen Endzeichen - automatisch gesteuert.

Über einen Puffer kann zur gleichen Zeit nur entweder eine Ausgabe oder eine Eingabe laufen; bei kombinierter Ausgabe erhalten alle betroffenen Geräte das gleiche Format. Die Eingabe- bzw. Ausgabegeschwindigkeit der angeschlossenen Geräte ist ohne Bedeutung, da mit Start und Rückmeldung gearbeitet wird; im kombinierten Betrieb bestimmt das langsamste Gerät den Takt.

Geräteauswahl- und Formatangaben sind als "Adresse" im betreffenden Befehl enthalten, die im Puffer gespeichert und von der Steuerlogik interpretiert wird. Die Zuordnung von Auswahlziffern zu Gerätekombinationen wird nach Bedarf verdrahtet.

Zu jedem Puffer vom R-Typ gehört, wie erwähnt, eine Steuerlogik, die außerhalb des Rechenkörpers untergebracht wird (nur für den ersten Puffer ist sie bereits dort enthalten). Außerdem ist für jedes Periphergerät eine Anpassungsschaltung vorzuschen (für die erste Schreibmaschine ist sie im Rechenkörper, für Locher und Leser in der Lochstreifeneinheit enthalten), sowie gegebenenfalls eine spezielle Stromversorgung.

Das MINCAL 4-Peripheriesystem umfaßt Anpassungsschaltungen für weitere Schreibmaschinen, Streifenlocher und -leser, für serielle Kartenlocher und -leser, für Fernschreiber usw.; spezielle Anpassungen werden auf Wunsch vorgesehen, ebenso eventuelle Netzteile usw.

6.6) Puffer vom D-Typ

Diese Ausführungsform besitzt - in Verbindung mit einer außerhalb des Rechners untergebrachten Steuerlogik - folgende Funktionen:

stellen-serielle, bit-parallele Eingabe von 6-stelligen, dezimalen Informationen,

stellen-serielle, bit-parallele Ausgabe von 6-stelligen, dezimalen Informationen,

Ausgabe von Schaltbefehlen programmierbarer Dauer.

Für Eingabe und Ausgabe ist je ein 4-bit-Kanal (Code 8-4-2-1) sowie eine Vorzeichenleitung vorgesehen; außerdem gibt es einen dezimalen Eingang. Der im Adressteil des Befehls programmierte Geber bzw. Empfänger der Information wird über 4 x 8 Adressleitungen identifiziert, so daß jeweils 4096 verschiedene, 6-stellige Informationsträger adressiert werden können. Die Ein/Ausgabe-Geschwindigkeit kann entsprechend den Verzögerungszeiten eingestellt werden; jede Stelle wird durch einen Stellenkanal (6 Anschlüsse, die nacheinander erregt werden) identifiziert.

Ein Ausgabebefehl kann statt eines seriellen Datentransfers als Schaltbefehl wirken (Unterscheidung durch die 1000er-Adresse; Zuordnung wird verdrahtet). Die Dauer des Schaltbefehls ist programmierbar, über den Dateninhalt des W-Registers, in Einheiten von 100 µs. Ist dessen Inhalt gleich Null, so hat der Schaltbefehl unbegrenzte Dauer.

6.7) Puffer vom A-Typ

Dieser Puffer benötigt keine zusätzliche Steuerlogik. Er nimmt die 4-stellige (oktale) Adresse auf, speichert sie und wählt einen von 4096 Datengebern an, der seinen Inhalt bit-parallel (über 12 oder 24 Leitungen plus Vorzeichen) anbietet. Nach Bereitstellen der Information wird sie durch den Puffer abgefragt und seriell übernommen.

Diese Ausführung wird u.a. in Verbindung mit dem Analog-Digital-Umsetzer des MINCAL 4-Systems verwendet. Die Meßstelle wird durch die Adresse bestimmt; die 1000er-Adressstelle bestimmt dabei gegebenenfalls Meßbereich und Filter-Zeitkonstante in einer verdrahteten Zuordnung.

6.8) Parallel-Ein/Ausgang

Der Datenaustausch zwischen Rechner und Umwelt kann, statt über die eingebauten Puffer, auch über die Parallel-Kanäle erfolgen. Dies empfiehlt sich immer dann, wenn wortparallele Übertragung mit Rechengeschwindigkeit erfolgen kann (Dauer für die Bereitstellung oder Übernahme einer Information 20 µs) oder wenn geringfügige Verzögerungszeiten in Kauf genommen werden können. Die parallele Ein/Ausgabe ist auch dann von Nutzen, wenn die eingebauten Puffer nicht ausreichen; man kann dann weitere, externe Puffer mit parallelem Ein/Ausgang vorsehen.

Das Parallelsystem arbeitet über je einen Kanal mit 24 + 1 Leitungen für die Daten-Ein/Ausgabe sowie einen Adreßkanal mit 12 Leitungen; ferner sind 8 Anschlüsse für die Adreßgruppe sowie einige Steuerleitungen vorgesehen.

6.9) MINCAL 4 - Standardkonsole

Eine Ausführungsform des MINCAL 4, die sowohl für den Betrieb als selbständige Datenverarbeitungsanlage ("off-line") wie als Prozeßrechner ("on-line") geeignet ist, wird als "Standard-Konsole" angeboten.

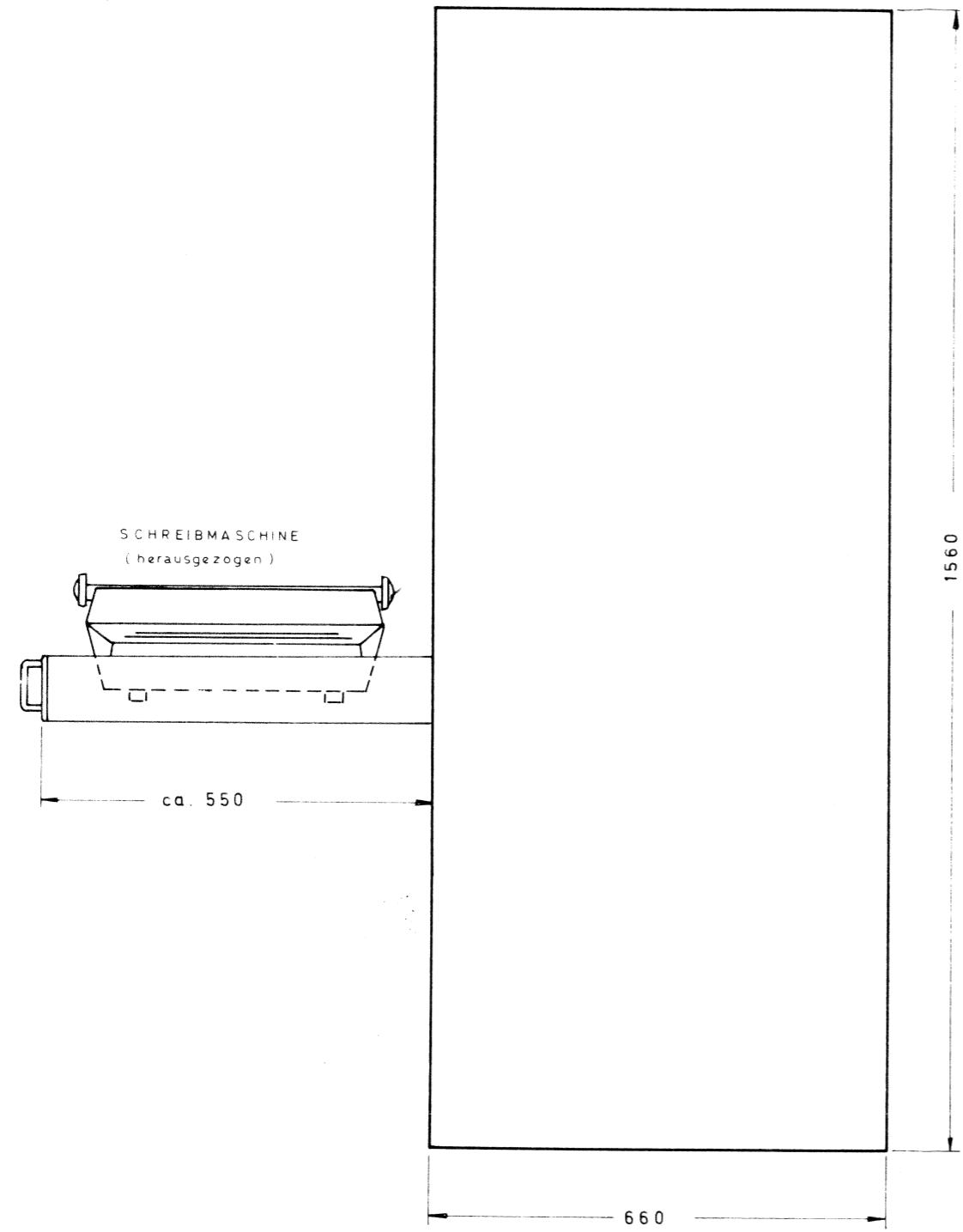
Sie besteht aus einem 19"-Doppel-Schrank mit 1100 mm Breite, 660 mm Tiefe und 1560 mm Höhe. Vorder- und Rückseite des Schranks sind mit je 2 verschließbaren Türen versehen, die bei Bedarf entfernt werden können.

Der rechte Schrankteil enthält, für einen sitzenden Benutzer oder Programmierer in bequemer Höhe, den Rechenkörper mit der Frontplatte; darüber ist die Stromversorgung, darunter der Kernspeicher mit seinen maximal 8 Einheiten angeordnet.

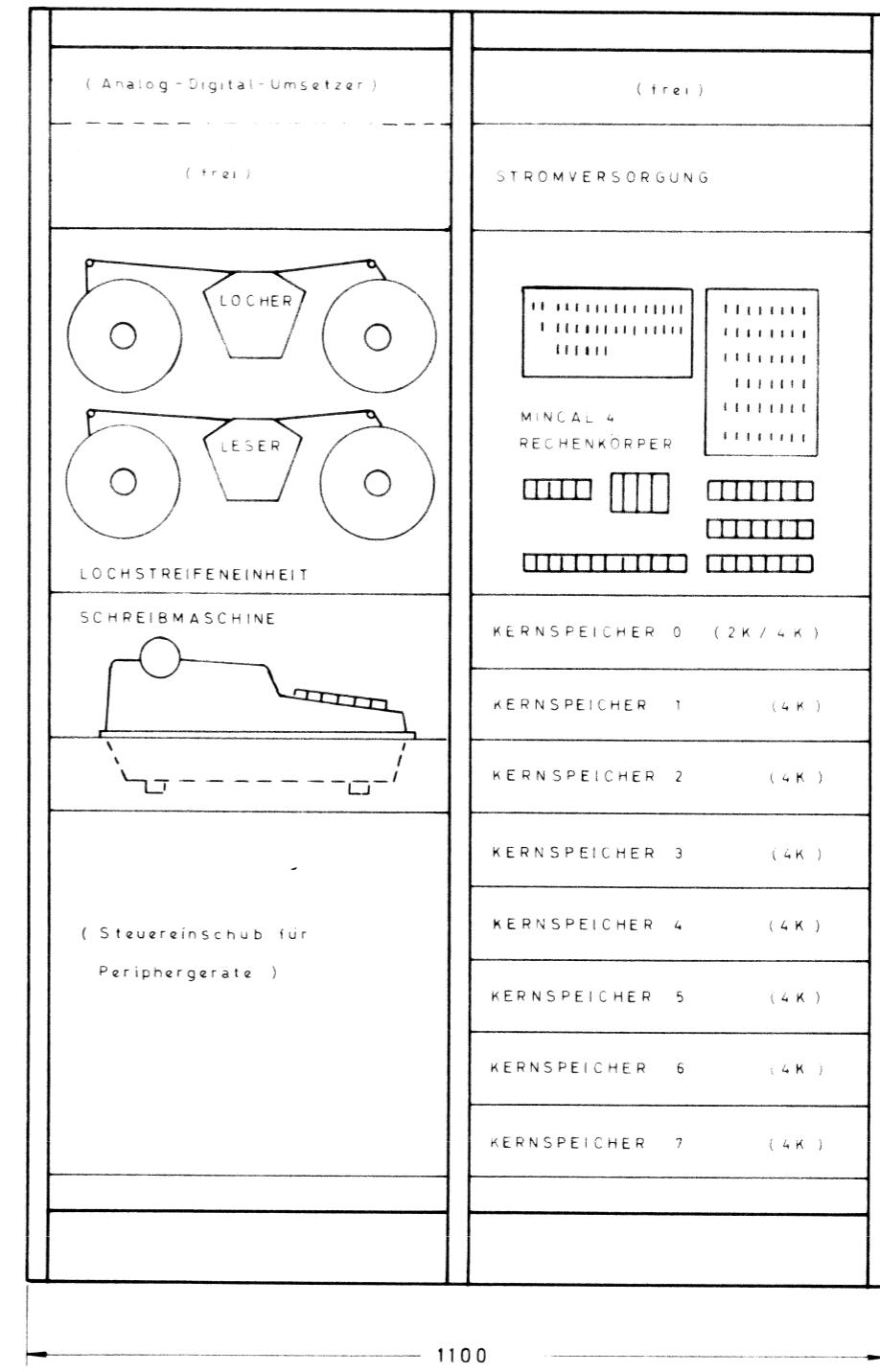
Im linken Teil ist, neben dem Rechenkörper, die Lochstreifeneinheit angeordnet, darunter auf einer ausziehbaren Lade die Schreibmaschine.

Unterhalb der Schreibmaschine kann ein Steuereinschub vorgesehen werden, der Steuerlogik, Anpassungsschaltungen und Netzteile für Periphergeräte und Datenanschlüsse in Verbindung mit den Puffern des MINCAL 4 aufnimmt. Die Konsole enthält außerdem Raum für einen Analog-Digital-Umsetzer sowie 2 weitere Einschübe.

Alle Einschübe sind in Teleskopschienen montiert und können ganz herausgezogen werden.



MINCAL 4 - STANDARDKONSOLE (Seitenansicht)



MINCAL 4 - STANDARDKONSOLE (Vorderansicht)

