

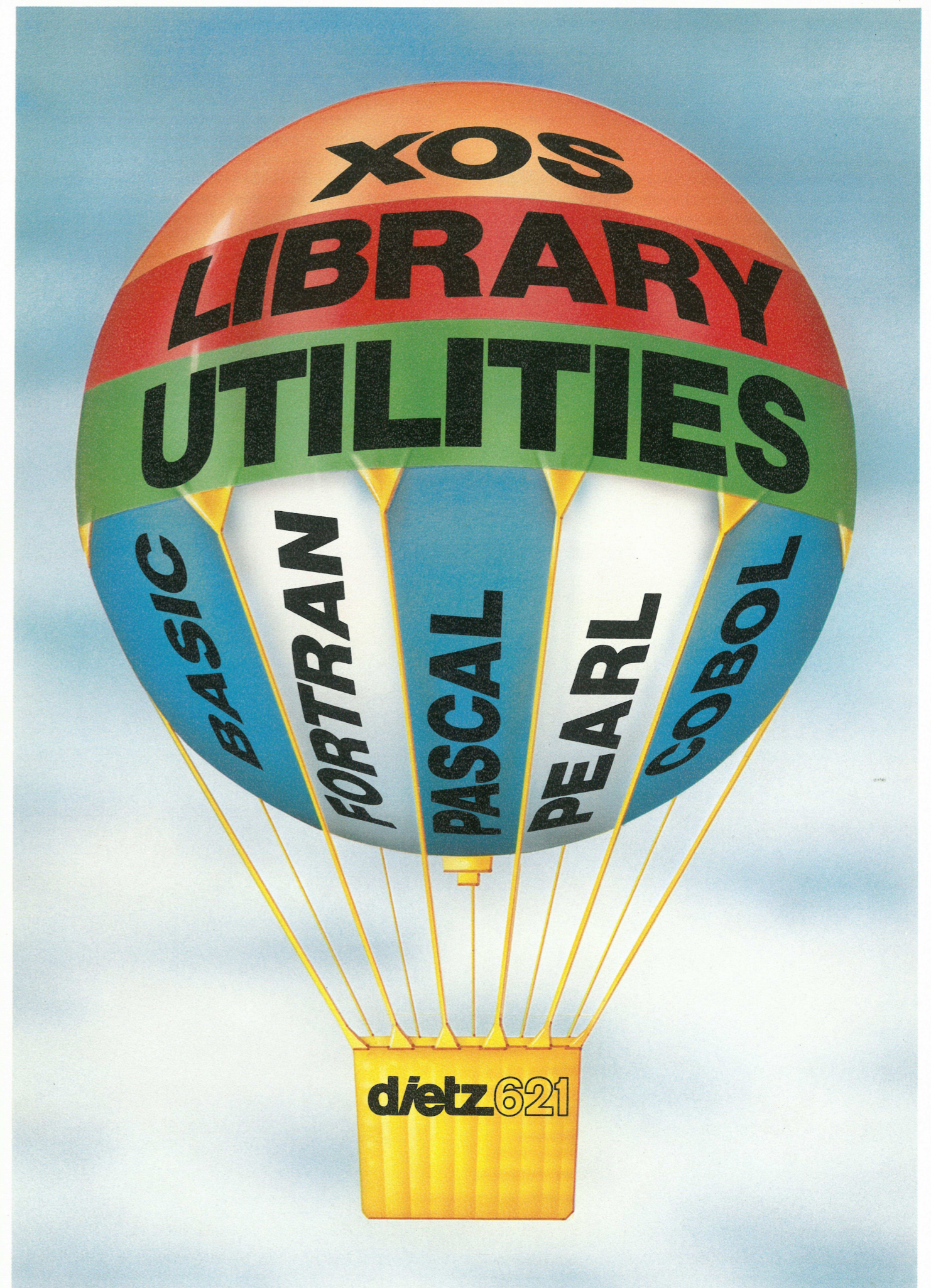
# d/etz 621



**Das Allround-Computersystem**

Fortschrittlich in der Konzeption, auf letztem technologischem Stand, von leistungsfähiger, modularer Software getrieben, universell einsetzbar: Das sind die Eigenschaften des Allround-Computersystems DIETZ 621.

Die vorliegende Broschüre gibt einen Überblick über die Architektur, die Hardware- und die Software-Moduln dieses auf Minicomputern basierenden Systems.



System-Architektur

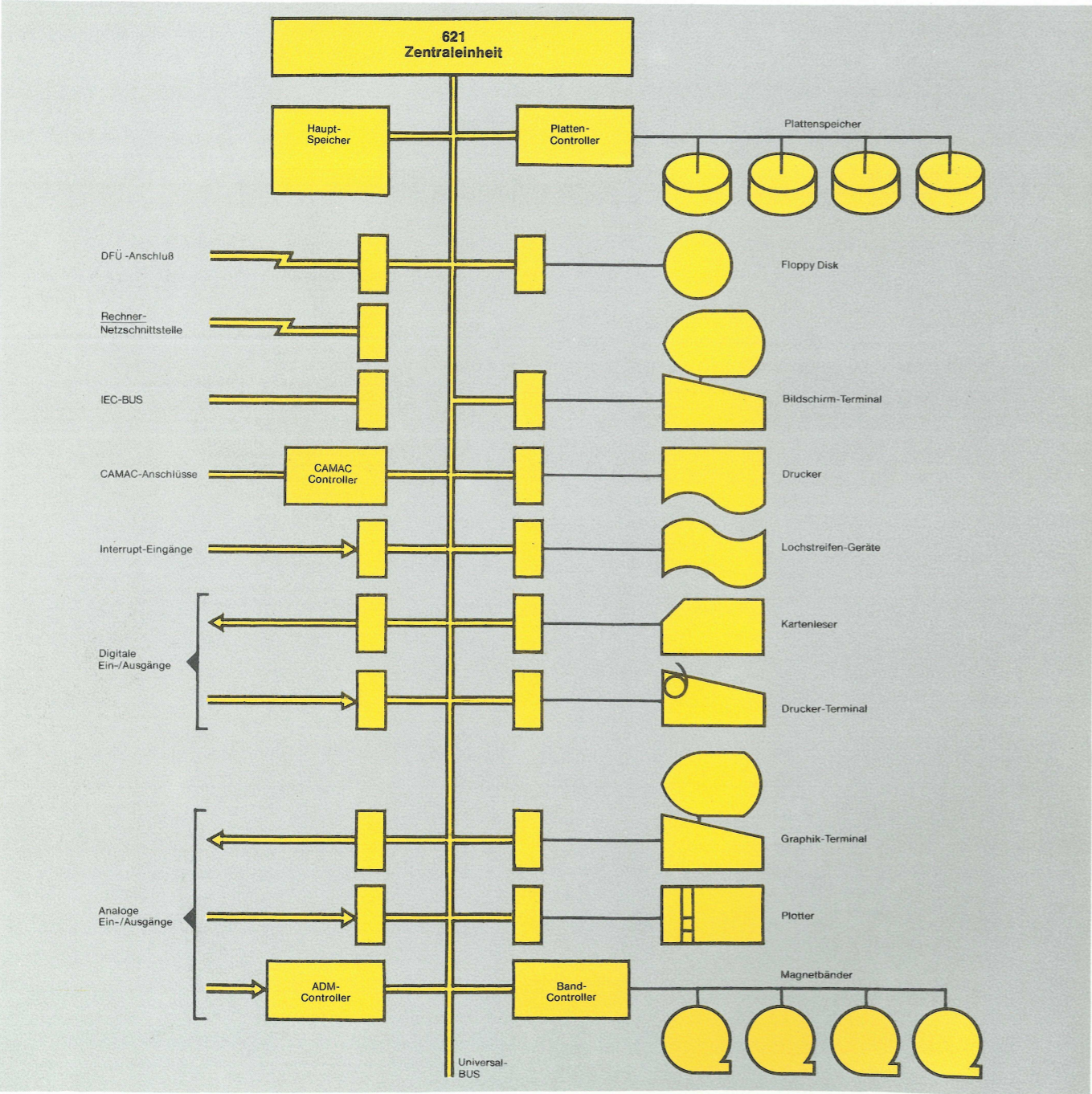
Ein Computer-System läßt sich in zwei Funktionsgruppen gliedern: Zentraleinheit und zugehörige Peripherie. Beide sind beim System DIETZ 621 durch den Universal-Bus verbunden.

Der Universal-Bus ist ein schneller, asynchroner Datenkanal. Schnell, weil er bis zu 6 Millionen Bytes pro Sekunde übertragen kann. Asynchron, um sich Funktionselementen mit unterschiedlicher Zugriffszeit anzupassen.

Am Bus, der das Rückgrat des Systems bildet, liegen – neben der Zentraleinheit – der Hauptspeicher, die

Interfaces für Peripheriegeräte und Prozeßanschlüsse sowie Controller für Geräte mit hoher Datenrate wie Platten- und Bandspeicher. Der Datentransfer von und zu der langsamen Peripherie findet unter Kontrolle der Zentraleinheit statt. Dagegen haben die Controller – ebenso wie die Zentraleinheit – direkten Zugriff zum Hauptspeicher und tauschen Daten unmittelbar mit ihm aus.

Die vom Universal-Bus geprägte Architektur des DIETZ 621 zeichnet sich vor allem dadurch aus, daß sie nahezu beliebige Konfigurationen zuläßt und jedes System modular erweiterbar ist.



Zentraleinheiten

Computer-Systeme müssen in der Lage sein, ein breites Leistungsspektrum abzudecken. Deshalb bietet das System DIETZ 621 mehrere Typen von Zentraleinheiten unterschiedlicher Leistungskraft.

Die 3 Zentraleinheiten der 621 X-Reihe sind Mini-computer mit ähnlichem Aufbau und weitgehend gleichen Komponenten, die sich jedoch hinsichtlich Verarbeitungsleistung, Speicher-Ausbau und -Verwaltung sowie Erweiterungs-Optionen unterscheiden. Sie stehen im Mittelpunkt von Systemen.

Der MICRO-621 ist ein Mikrocomputer, der als Zentraleinheit kleiner Systeme oder als Subeinheit größerer Systemkomplexe Verwendung findet.

Alle Zentraleinheiten bilden eine echte Familie. Sie haben dieselbe Struktur, sind hinsichtlich Befehlsvorrat und Anschlüssen kompatibel und weisen alle die unverwechselbaren Merkmale des DIETZ 621 auf:

Multibyte-Struktur (variable Datenwort- und Instruktionslänge), Multiregister-Konzept (128 Universalregister je Programmebene) und Multiprogramm-Hardware (bis zu 16 Programmebenen mit je einem Registersatz zur unverzögerten Programmunterbrechung).

		Mikrocomputer		Minicomputer	
		MICRO-621	621 X1	621 X2	621 X3
Prozessor	Datenwort-Länge	1 ... 256 byte	1 ... 256 byte	1 ... 256 byte	1 ... 256 byte
	Instruktions-Länge	1 ... 5 byte	1 ... 5 byte	1 ... 5 byte	1 ... 5 byte
	Instruktions-Anzahl	223	223	223	223
	Instruktions-Dauer	3 ... 10 µs	1.9 ... 6.0 µs	0.8 ... 6.0 µs	0.8 ... 6.0 µs
	Adressierungsarten	8	8	8	12
	Adreßraum/Ebene	64 Kbyte	64 Kbyte	64 Kbyte	192 Kbyte
	Programmebenen	max. 16	16	16	16
Register	Technologie	MOS	MOS	bipolar	bipolar
	Zugriffszeit	300 ns	300 ns	70 ns	70 ns
Hauptspeicher	Anzahl je Ebene	128	128	128	128
Hauptspeicher	Technologie	MOS	MOS	MOS	MOS
	Zykluszeit	550 ns	550 ns	550 ns	550 ns
	Kapazität (Standard)	16 ... 56 Kbyte	16 ... 48 Kbyte	32 ... 256 Kbyte	max. 1 Mbyte
Bus	Verwaltung			Partitioning	Mapping
	Datenbreite	8 bit	16 bit	16 bit	16 bit
	Adreßbreite	16 bit	16 bit	20 bit	20 bit
	Transferrate max.		6 Mbyte/s	6 Mbyte/s	6 Mbyte/s
Arithmetik-Prozessor	Echtzeit-Uhr, quartzgesteuert	Option	Standard	Option	Option
	Systemüberwachung	Option	Standard	Standard	Standard
	Bootstrap-ROM	Option	Standard	Standard	Standard
	Integriertes V24-Interface		Standard	Standard	Standard
	Integrierter Plattencontroller		Standard	Standard	Standard
	Weitere Peripherie-Boards		Standard	Standard	Standard
	Einkarten-Interface	1	2	2	2
	Universal-Bus-Anschluß	8 oder 20	1	1	1
	DMA zum Hauptspeicher	Option	Standard	Standard	Standard
	Routine-Bedienungsfeld	Option	Standard	Standard	Standard
	Komfort-Bedienungskonsolle	Option	Standard	Option	Option

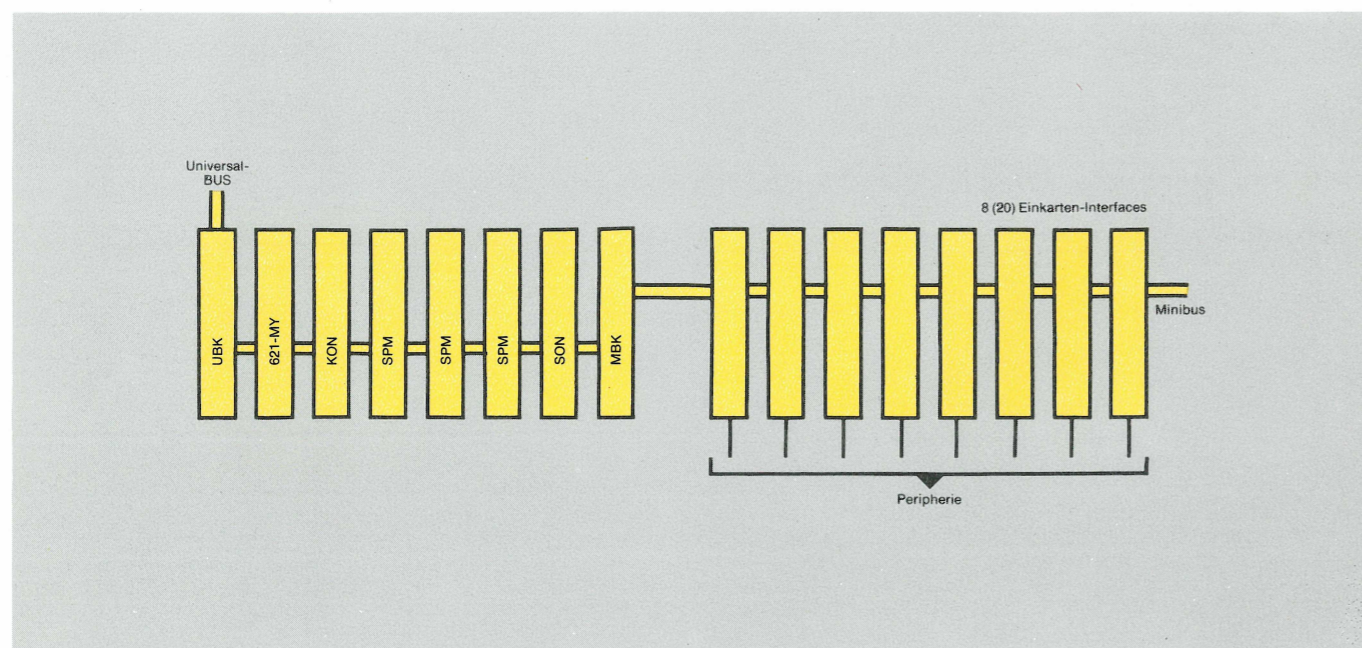
## MICRO-621

Der Mikrocomputer MICRO-621 spielt eine zweifache Rolle im Rahmen der 621-Familie: Einerseits als selbständige Zentraleinheit kleiner Konfigurationen, zum anderen als intelligente Subeinheit größerer 621-Systeme. Für beide Aufgaben ist er aufgrund seiner Technologie und seines physischen Aufbaus gut gerüstet.

Der MICRO-621 ist als offener 19"-Rahmen aufgebaut, der in der Grundausrüstung den Mikroprozessor 621 MY, einen Speicher-Modul (SPM) mit 4 KB RAM, den Minibus-Koppler (MBK) sowie – je nach Modell – 8 oder 20 vorverdrahtete Steckplätze für Einkarten-Interfaces enthält.

In System-Anwendungen finden sich folgende Optionen: MICRO-621 Kontrollmoduls mit Echtzeit-Uhr und Systemüberwachung, Erweiterung auf 16 Programmebenen, Speicher-Erweiterung um max. 52 KB in Form von 4-KB-RAMs und/oder programmierbaren ROMs. Ein Universalbus-Koppler sowie ein Sondermodul sind weitere Optionen.

In den vorverdrahteten Plätzen können alle Einkarten-Interfaces des 621-Programms verwendet werden; sie sind über den Minibus mit dem Mikroprozessor verbunden und werden unter dessen Kontrolle bedient. Über die Interfaces werden Anschlüsse zu Peripheriegeräten und Prozeßanschlüssen hergestellt.



Die Eigenschaften des MICRO-621 ermöglichen es, autonome Kleinstsysteme aufzubauen, in deren Mittelpunkt dieser Mikrocomputer steht. Dank der Kompatibilität zu den Minicomputern der DIETZ 621-Familie können Programme auf diesen Systemen erstellt, getestet und problemlos auf den MICRO-621 übertragen werden.

Eine besondere Rolle spielt der MICRO-621 als Subeinheit größerer 621-Systeme. Über den Universal-Bus mit einer oder mehreren Zentraleinheiten 621 X gekoppelt, bilden Mikrocomputer wichtige Bestandteile von Mehrprozessor-Systemen.

## 621 X1

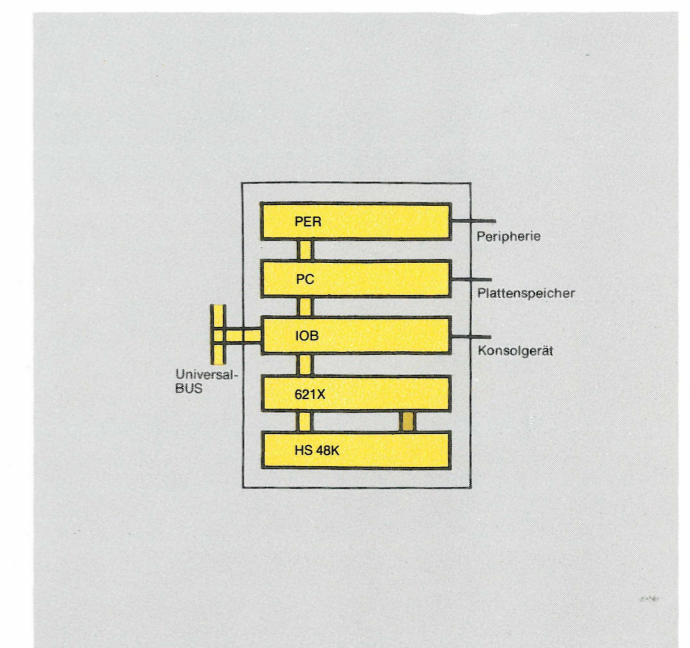
Mit der Zentraleinheit 621 X1 steht ein Minicomputer für Systeme kleinen bis mittleren Umfangs zur Verfügung. Er zeichnet sich durch kompakten, raumsparenden Aufbau und ein Minimum an Komponenten aus, die in einem 19"-Gehäuse von 4 Höheneinheiten untergebracht und über den internen Bus verbunden sind.

Kernstück ist der in modernster Schaltungstechnik aufgebaute 621 X-Prozessor. Er wird ergänzt durch einen Hauptspeicher von max. 48 KB sowie einen Register-Speicher in MOS-Technologie, einen integrierten Plattencontroller und das I/O-Board, das – neben anderen Funktionen – die Verbindung zum Konsolgerät sowie zum Universal-Bus herstellt; über den Bus sind weitere periphere Baugruppen anschließbar.



Im Gehäuse befindet sich ein weiterer Steckplatz für ein Peripherie-Board sowie die Stromversorgung; die Frontplatte ist mit einem Routine-Bedienungsfeld ausgestattet.

Zur Ausrüstung des 621 X1 gehören ferner: Die quartz-gesteuerte Echtzeit-Uhr, Systemüberwachungsfunktion (Speicher-Paritätsbit, Netzausfall, Bus-Fehler), Auto-Restart bei Netzwiederkkehr, ein Bootstrap-ROM und ein Steckplatz für Einkarten-Interface auf dem I/O-Board.



Die Zentraleinheit DIETZ 621 X1 erlaubt es, kostengünstige Computer-Systeme aufzubauen, die – im Rahmen der verfügbaren Hauptspeicherkapazität – für eine Vielzahl von Aufgaben geeignet sind. Trotz seiner niedrigen Kosten und seines kompakten Aufbaus besitzt der 621 X1 alle Eigenschaften eines vollwertigen Minicomputers.

## 621 X2

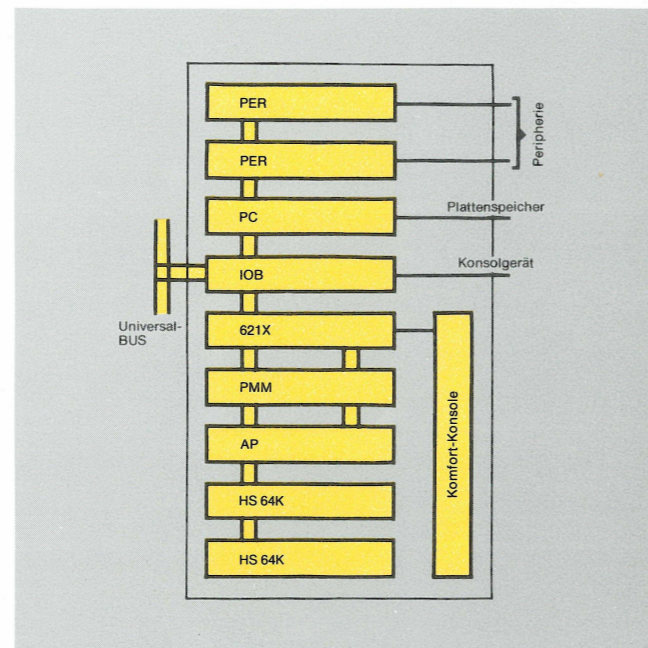
Der Minicomputer 621 X2 ist eine sehr leistungsfähige Zentraleinheit der DIETZ 621-Familie und dadurch für größere Systeme prädestiniert. Er verwendet die Grundkomponenten des 621 X1 und ist ähnlich wie dieser aufgebaut. Jedoch gibt es im 621 X2 eine Reihe bedeutsamer Erweiterungen.

Die Baugruppe PMM enthält einen schnellen Register-Speicher (Pool) und einen Verwaltungs-Modul (Memory Manager) für den Hauptspeicher. Dadurch wird eine wesentlich höhere Verarbeitungsgeschwindigkeit erreicht und der Adreßraum erheblich erweitert.



Zur Beschleunigung numerischer Operationen kann ein Arithmetik-Prozessor AP vorgesehen werden; er führt die vier Grundrechenarten aus und steht in mehreren Versionen zur Verfügung, die sich durch die Anzahl der verwendeten Datentypen unterscheiden.

Die Baugruppen AP und PMM sind untereinander und mit dem 621 X-Prozessor über einen besonderen, vom internen BUS unabhängigen Datenkanal verbunden; auf beiden können gleichzeitig Daten transferiert werden.



Im Gehäuse des 621 X2 finden bis zu 128 KB Hauptspeicher Platz; extern kann der Hauptspeicher auf 256 KB und darüber hinaus erweitert werden. Der Hauptspeicher ist in MOS-Technologie ausgeführt. Werden Speicher-Baugruppen gleicher Kapazität verwendet, wird durch Interleaving-Technik ein Maximum an Geschwindigkeit erzielt.

Der 621 X2 besitzt – neben dem Routine-Bedienungsfeld – eine Komfort-Konsole, die mit Funktions- und Datentasten sowie mit binären und hexadezimalen Anzeigen ausgerüstet ist.

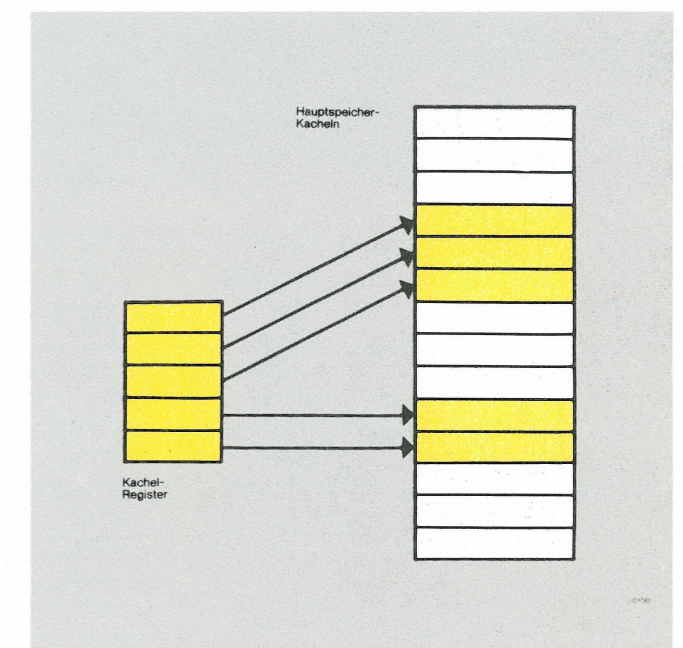
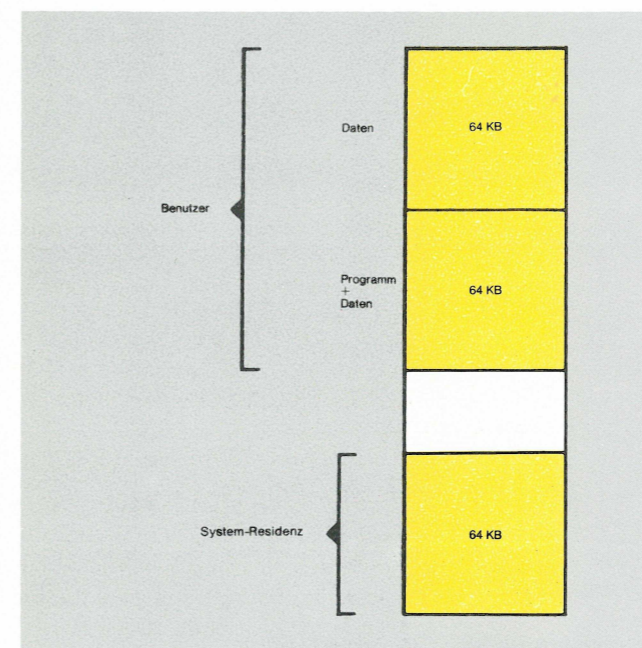
## 621 X3

Mit der Zentraleinheit 621 X3, der leistungsstärksten der 621-Baureihe, steht ein Minicomputer zur Verfügung, der aufgrund seiner besonderen Eigenschaften vor allem den Aufbau größerer Echtzeit- und Mehrbenutzer-Systeme erlaubt.

Besonders hervorzuheben ist der auf 192 Kbyte vergrößerte Speicherraum, der jedes einzelne Benutzerprogramm adressieren kann. Dazu gehört ein Programm- und Datenbereich von max. 64 Kbyte, ein reiner Datenbereich derselben Größe und die

— gemeinsame — Systemresidenz von ebenfalls max. 64 Kbyte, in dem u.a. das Betriebssystem sowie weitere systemnahe Software-Moduln Platz finden.

Bis zu 16 verschiedene, logisch zusammenhängende Benutzer-Bereiche von jeweils max. 128 Kbyte können innerhalb des gesamten verfügbaren Adreßraums von 1024 Kbyte residente Programme und Daten aufnehmen, — eine Voraussetzung für große Echtzeit- und Mehrbenutzer-Systeme.



Der Hauptspeicher wird nach dem Memory-Mapping-Prinzip verwaltet. Er ist in Kacheln eingeteilt, die von 32 Kachel-Registern je Programmebene angesprochen werden.

Die Vorteile dieses Prinzips liegen in optimaler Speicherausnutzung infolge dynamischer Zuordnung, in praktisch unverzögerter Zuschaltung der physischen Adresse sowie in Hardware-unterstütztem Speicherschutz.

Die residenten Systemprogramme sind gegenüber den Benutzern dadurch geschützt, daß der 621X3 zwei Betriebsarten (Benutzer- und System-Modus) besitzt.

Hinsichtlich Ausbaumöglichkeiten und Optionen entspricht diese Zentraleinheit voll dem Modell 621X2.

## Peripherie-Geräte

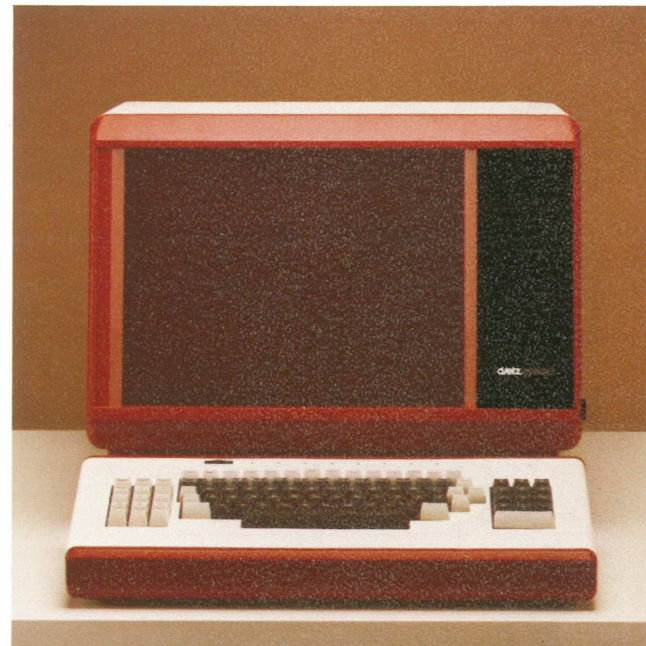
Das Peripherie-Spektrum des Systems DIETZ 621 ist außerordentlich breit. Es umfaßt eine Vielzahl von Geräten, die der Kommunikation zwischen Benutzer und System dienen, Daten auf maschinell lesbare Medien aufzeichnen oder solche in das System eingeben.

Hierzu gehören leise, druckende Terminals ebenso wie bedienungsfreundliche Bildschirm-Terminals mit großem, blendfreiem 16"-Schirm. Sie stellen den Dialog mit dem System her, entweder als Masterkonsole oder als normales Benutzer-Gerät.



Eine reiche Auswahl von seriell und zeilenweise arbeitenden Druckern unterschiedlicher Leistung sorgt für die Ausgabe von Programmen und Daten auf Papier. Alle haben die volle Schreibbreite von 132 Zeichen je Zeile, sind gepuffert und besitzen Einrichtungen zur Vorschub-Steuerung.

Zur Standard-Peripherie gehören außerdem Kartenleser, Lochstreifenleser und -stanzer, OCR-Handler, Floppy Disks und Magnetband-Laufwerke.



Die Geräte-Peripherie ist über Interfaces an den Universal-Bus angeschlossen und durch spezielle Software-Treiber im Rahmen des Betriebssystems unterstützt. Sie wird zusammen mit der übrigen System-Hardware von DIETZ gewartet.

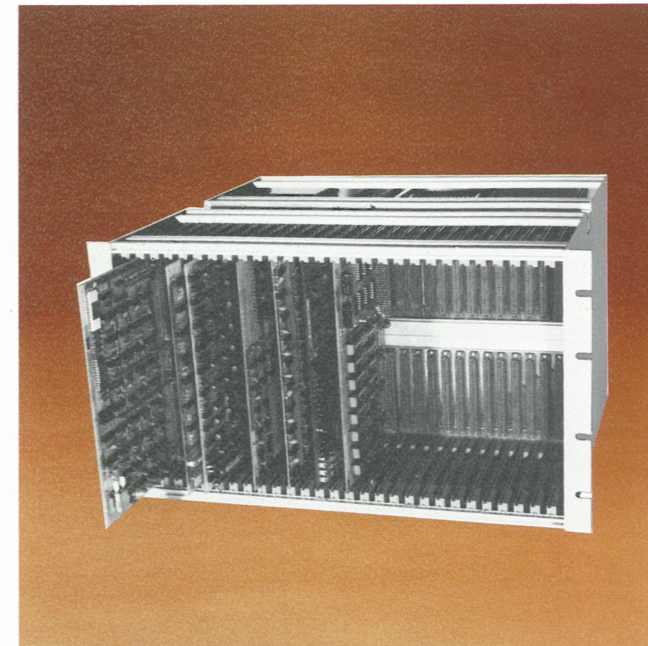
Spezielle, ebenfalls Software-unterstützte Schnittstellen sehen den Anschluß weiterer Geräte vor. Dazu gehören u. a. graphische Bildschirm-Terminals, Digitizer und Plotter.

## Prozeß-Peripherie

Der Einsatz von Minicomputern für die Prozeß-Datenverarbeitung setzt voraus, daß Prozeßsignale empfangen und abgegeben werden können. Das Computer-System DIETZ 621 verfügt über eine reiche Auswahl an Prozeß-Peripherie, von der entsprechend den Bedürfnissen des Anwendungsfalls Gebrauch gemacht wird.

Hierzu gehören unter anderem:

- Statische digitale Eingänge
- Speichernde digitale Ausgänge
- Dynamische Interrupt-Eingänge
- Zählengänge und zeitgesteuerte Ausgänge
- Einkanal-Analogeingänge



- Analog-Meßsysteme mit schnellen Stufenverschlüßlern oder integrierendem Meßverfahren
- Analog-Ausgänge.

Die meisten dieser Moduln sind mit unterschiedlichen Spannungspegeln sowie auf Wunsch mit galvanischer Trennung verfügbar. Alle werden vom Betriebssystem voll unterstützt und sind in den Prozeß-Programmierungssprachen des Systems DIETZ 621 ansprechbar.



Statt dessen – oder zusätzlich – kann der Anwender auch auf das weitverbreitete CAMAC-System zurückgreifen, um die Verbindung zwischen Computer und Prozeß herzustellen; es steht ihm dann ein breites Spektrum marktgängiger Ein-/Ausgabe-Moduln zur Verfügung. DIETZ liefert auf Wunsch CAMAC-Crates mit einem speziellen, an den 621-BUS angeschlossenen Controller, der programm- oder selbstgesteuerten Datentransfer zuläßt und die CAMAC-Moduln, deren Subeinheiten und Funktionen sowie die Interrupts bedient. Die auf BASEX beruhende problembezogene Programmiersprache CAMBAS unterstützt den Anwender auf komfortable Weise in der Handhabung der CAMAC-Peripherie.

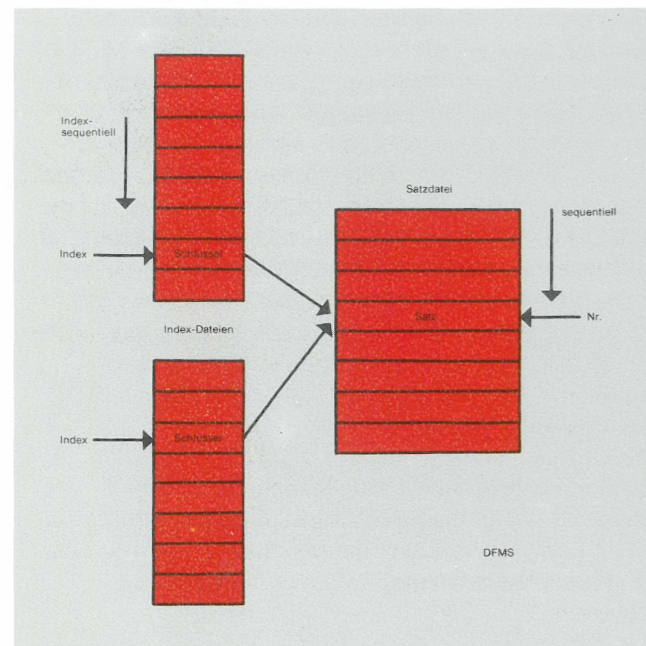
Schließlich besteht die Möglichkeit, externe Meß- und Steuereinheiten über den international genormten IEC-Bus anzuschließen. Ein entsprechendes, von der Systemsoftware unterstütztes IEC-Interface ist Bestandteil der DIETZ 621-Peripherie.

## Plattenspeicher

Computer-Systeme sind heute ohne Plattenpeicher nicht mehr denkbar. Als Hintergrundspeicher großer Kapazität unterstützen sie Programmierung und Test, halten Teile von Programmen bereit und große Datenmengen im schnellen Zugriff. Deshalb sind Plattenpeicher ein integraler Bestandteil von DIETZ 621-Systemen. Plattenlaufwerk und zugehöriger Controller bilden ein Subsystem, das an den Bus des Systems angeschlossen ist und Daten – meist im direkten Zugriff – mit dem Hauptspeicher austauscht.

Das System DIETZ 621 umfaßt vier Typen von Plattenspeichern mit unterschiedlicher Kapazität und Zugriffsgeschwindigkeit. Bis zu vier Laufwerke sind im allgemeinen anschließbar. Dadurch sind Plattenkapazitäten von 512 KB bis 240 MioB zu erzielen.

Typ	3402	3512	3521	3522
Medium	Klein-Plattenspeicher (Floppy Disk)	Feste Platte und wechselbare Plattenkassette	Wechselbarer Plattenstapel	Wechselbarer Plattenstapel
Anzahl Laufwerke	2	1 . . . 4	1 . . . 4	1 . . . 4
Kapazität je Laufwerke (netto)	0.25 Mbyte	2 x 4.8 Mbyte	30 Mbyte	60 Mbyte
Mittlere Positionierzeit		38 ms	30 ms	30 ms
Mittlere Latenzzeit		13 ms	8 ms	8 ms
Mittlere Zugriffszeit	243 ms	51 ms	38 ms	38 ms
Max. Transferrate	25 Kbyte/s	312 Kbyte/s	1.2 Mbyte/s	1.2 Mbyte/s



Meist ist ein System nur mit einem Plattentyp ausgestattet; jedoch ist es auch möglich, Platten unterschiedlichen Typs an ein System anzuschließen.

Alle 4 Plattentypen werden von speziellen Software-Treibern sowie von Datei-Zugriffs- und Verwaltungssystemen unterschiedlichen Komforts unterstützt. Auf Ebene des Benutzers werden sie völlig identisch behandelt, so daß der Anwender – abgesehen von Kapazität und Zugriffsgeschwindigkeit – auf den Plattentyp keine Rücksicht zu nehmen braucht.

Plattenspeicher bieten wesentlich erhöhte Gesamt-Speicherkapazität. Trotz der – verglichen mit dem Hauptspeicher – längeren Zugriffszeit tragen sie erheblich zur Leistungsfähigkeit des Computersystems bei und geben dem Benutzer die Möglichkeit, bequem und sicher mit dem System zu arbeiten.

Die Plattenspeicher des Systems DIETZ 621 enthalten umfangreiche Sicherungs-Funktionen wie Schreibschutz, CRC-Prüfung, Sektor- und Spur-Kennungen sowie weitere Funktionen.

## Basis-Systeme

Basis-Systeme sind Standard-Konfigurationen, die – von einem Minimalumfang ausgehend – systematisch erweitert werden können.

Sie bestehen aus einer oder zwei Zentraleinheiten sowie einem Plattenspeicher-Subsystem, die in einem vollständig mit Netzfilter und -verteiler, Erdungsschienen, Kabeln usw. ausgestatteten Systemschrank untergebracht sind. Außerdem gehört ein Drucker- oder Bildschirm-Terminal als Konsolgerät dazu.

Aus den Kombinationen von Zentraleinheit- und Plattentypen ergeben sich die in der Tabelle aufgeführten, als DIETZ-Systeme 621 bezeichneten Basis-Konfigurationen.

		Ein-Prozessor-Systeme								Zwei-Prozessor-Systeme			
DIETZ-System 621		B0	B1	E1	B2	E2	F2	E3	F3	E22	F22	E33	F33
	Typ	M	X1	X1	X2	X2	X2	X3	X3	X2	X2	X3	X3
Zentral- einheit	Anzahl	1	1	1	1	1	1	1	1	2	2	2	2
	Hauptspeicher min. Hauptspeicher max. (KB)	32 56	32 48	32 48	64 256	64 256	64 256	64 1024	64 1024	128 512	128 512	128 2048	128 2048
Platten- speicher	Typ*	3402	3402	3512	3402	3512	352x	3512	352x	3512	352x	3512	352x
	Laufwerke max.	2	2	4	2	4	4	4	4	4	4	4	4
	Kapazität min. Kapazität max. (MB)	0.5	0.5	9.6 38.4	0.5	9.6 38.4	30 240	9.6 38.4	30 240	9.6 38.4	30 240	9.6 38.4	30 240

\*  $352x = 3521$  oder  $3522$

Zwei Prozessor-Systeme enthalten zwei Zentraleinheiten gleichen Typs, die über einen schnellen Datenkanal (Inter-Bus) verbunden sind und auf ein gemeinsames Plattenspeicher-Subsystem zugreifen. Sie bieten erhöhten Systemdurchsatz bei relativer Unabhängigkeit und Sicherheit der Teilsysteme.

Die Basis-Systeme bilden eine Modellreihe, die hinsichtlich ihrer Leistungsfähigkeit ein breites Spektrum möglicher Aufgaben abdeckt.

Dazu gehört der Einsatz als Prozeßrechner ebenso wie technisch-wissenschaftliche Anwendungen mit einem oder mehreren, voneinander unabhängigen Benutzern. Kommerziell orientierte Aufgaben können mit ihnen gelöst werden; aber auch für Zwecke der Datenfernverarbeitung werden sie eingesetzt.

Die Wahl des geeigneten Modells hängt von verschiedenen Faktoren ab; die wichtigsten Kriterien sind die Verarbeitungsleistung sowie die benötigte Kapazität des Haupt- und des Plattenspeichers.

## Konfigurationen

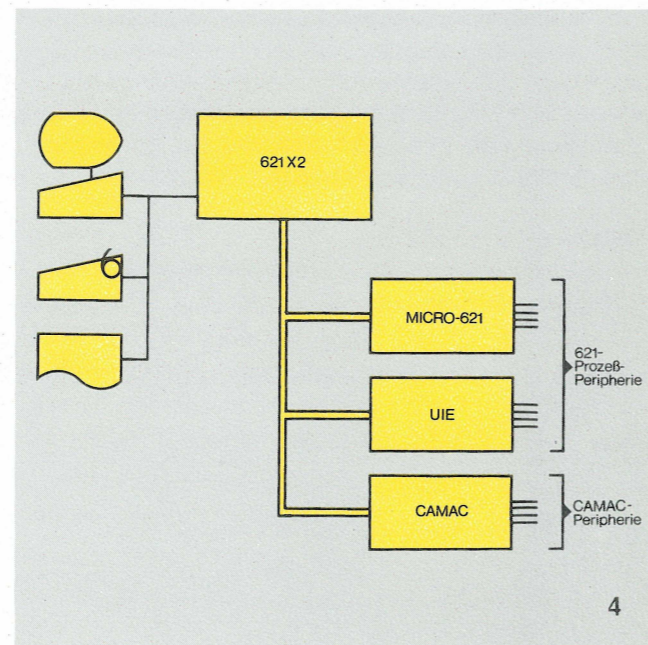
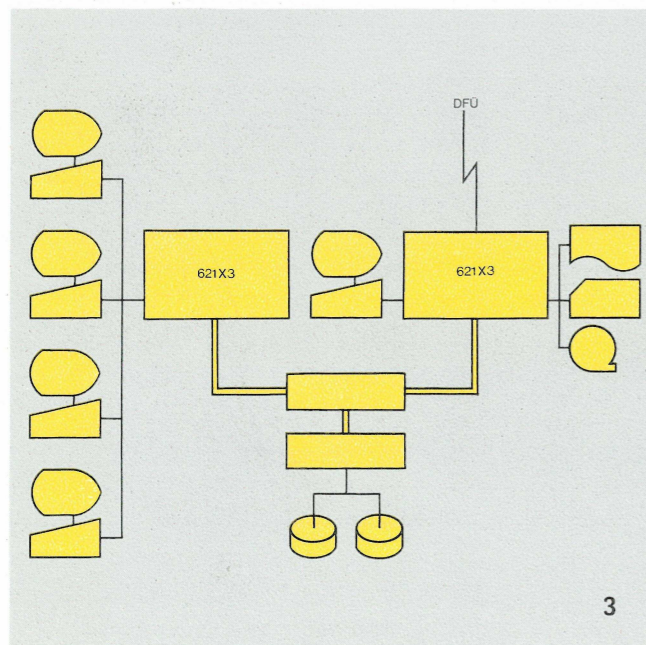
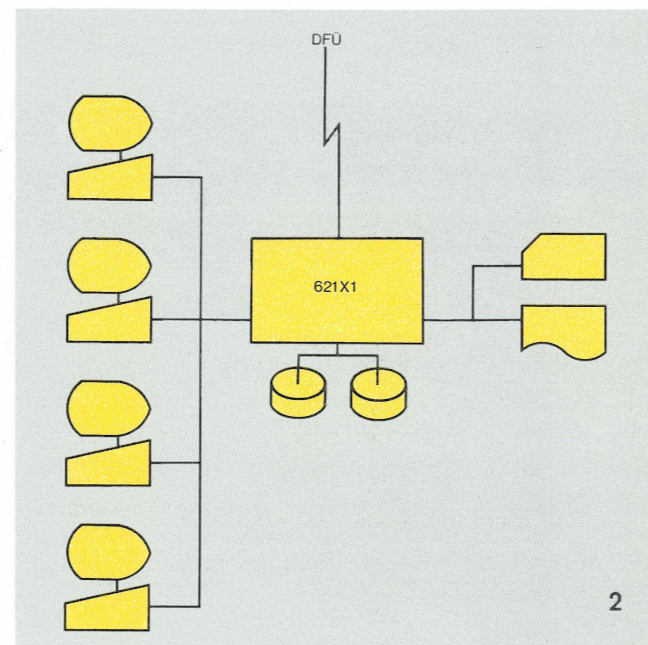
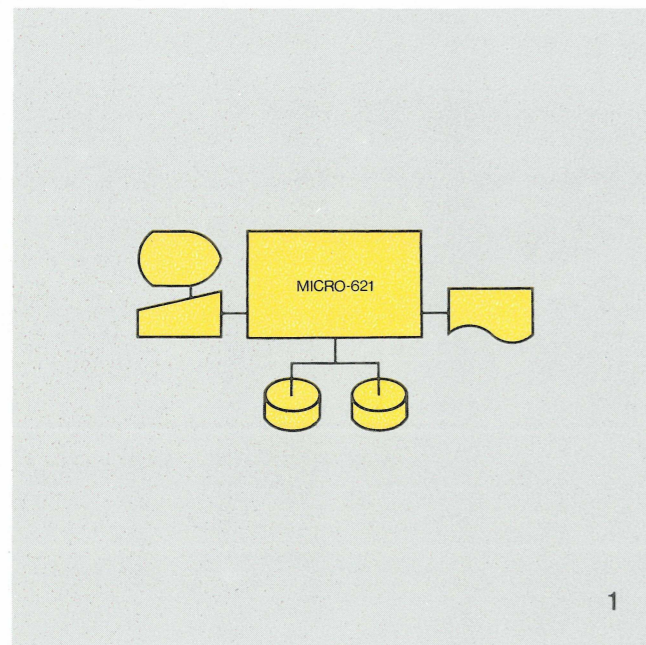
Die Modularität und die Menge der verfügbaren Komponenten des Computer-Systems DIETZ 621 läßt eine nahezu unbegrenzte Vielfalt möglicher Konfigurationen zu. Einige Beispiele mögen dies verdeutlichen:

Bild 1 zeigt ein plattenunterstütztes Einbenutzer-System, wie es häufig für Aufgaben z. B. der technisch-wissenschaftlichen Datenverarbeitung benutzt wird.

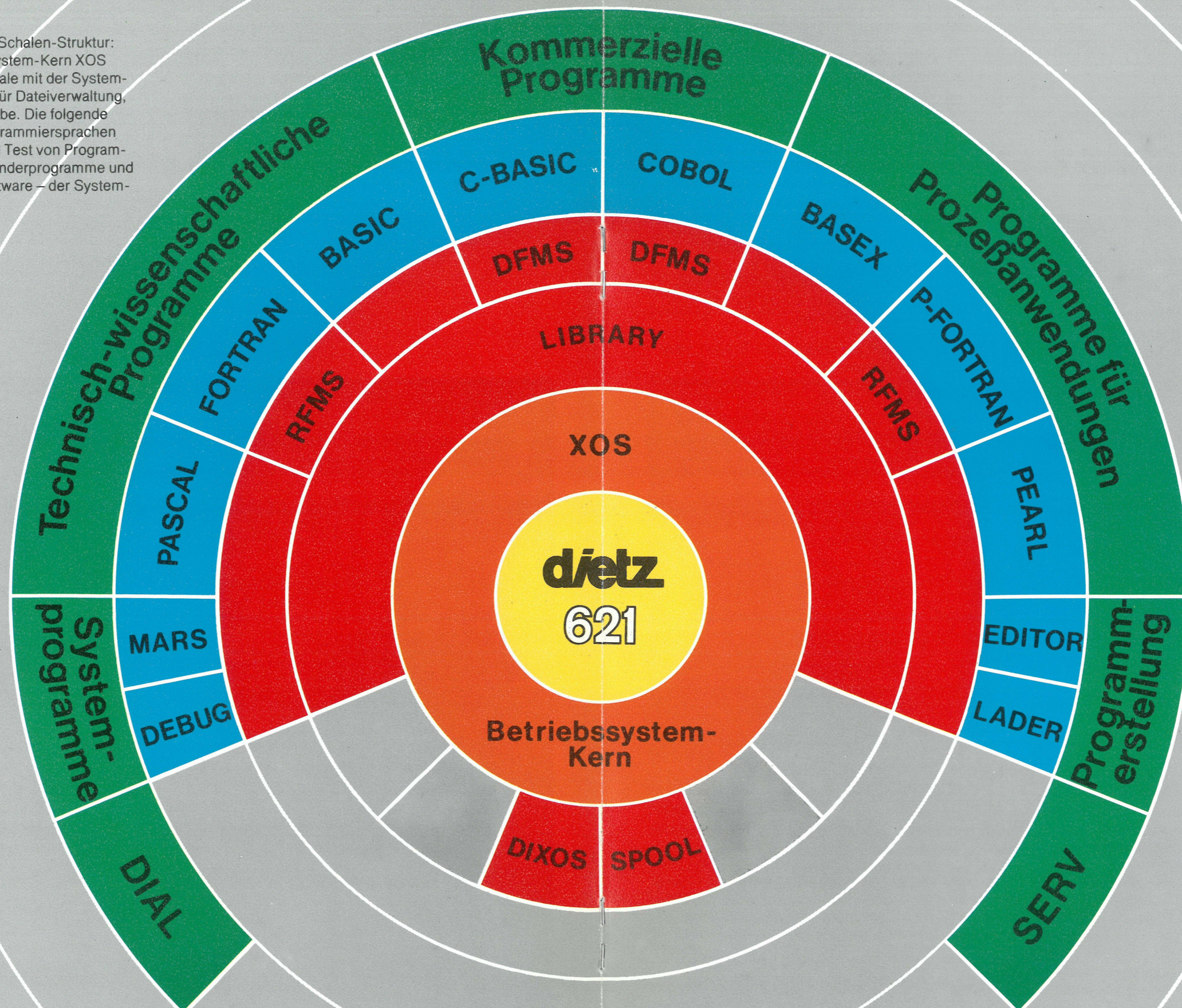
In Bild 2 ist ein Mehrbenutzer-System dargestellt, das im Timesharing-Betrieb arbeitet und gleichzeitig als RJE-Station einer übergeordneten EDV-Anlage dient.

Dieselbe Funktion erfüllt die Konfiguration von Bild 3; sie ist jedoch als Zwei-Prozessor-System aufgebaut, enthält umfangreichere Peripherie und ist demzufolge wesentlich leistungsfähiger.

Bild 4 gibt ein typisches Prozeßrechner-System wieder; die Prozeßsignale werden teilweise im Zentralrechner behandelt, zum Teil aber auch von einem Mikrocomputer vorverarbeitet. Die Prozeßperipherie besteht aus DIETZ 621- und CAMAC-Modulen.



Die System-Software besitzt eine Schalen-Struktur:  
Die Hardware wird vom Betriebssystem-Kern XOS  
umgeben. Daran schließt eine Schale mit der System-  
Bibliothek und speziellen Moduln für Dateiverwaltung,  
Rechner-Kopplung und Ein/Ausgabe. Die folgende  
Schale umfaßt Übersetzer für Programmiersprachen  
sowie Hilfsmittel für Erstellung und Test von Program-  
men. Ganz außen liegen die Anwenderprogramme und  
— als Bestandteile der System-Software — der System-  
Dialog und die Dienstprogramme.



Betriebssystem XOS

XOS ist die Bezeichnung für das universelle Betriebssystem der Computer-Familie DIETZ 621. Es handelt sich um ein nach dem neuesten Stand der Technik konzipiertes Software-System, welches auf die besondere Hardware-Struktur des DIETZ 621 zugeschnitten ist und die Basis aller Software-Moduln darstellt.

Besonders hervorzuheben am Betriebssystem XOS sind:

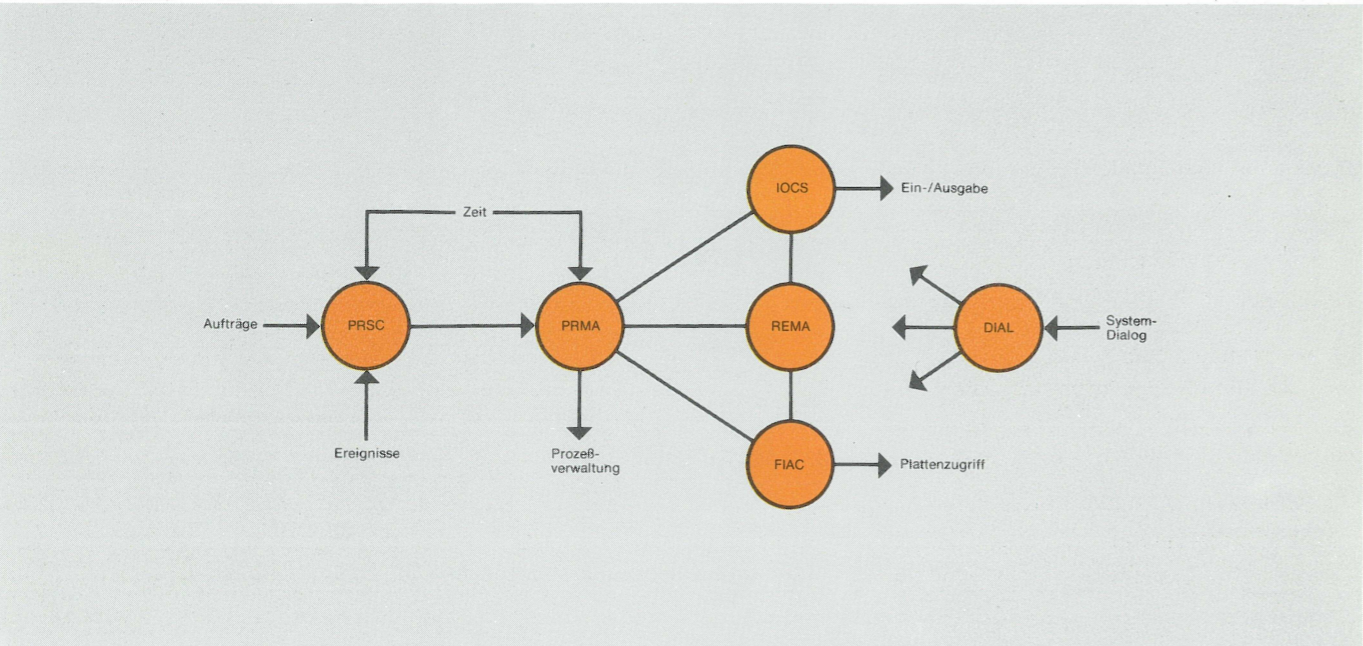
- Modularer Aufbau
- Geringer Speicherbedarf
- Laufzeitoptimale Implementierung
- Erweiterbarkeit um zusätzliche Moduln
- Kompatibilität zu allen Sprachübersetzern, Programmhilfen, Dienstprogrammen usw.
- Universelle Einsetzbarkeit
- Vielseitige Betriebsarten.

Der Kern des Betriebssystems XOS besteht aus folgenden sechs Moduln:

Der Modul PRMA verwaltet die Prozesse, d. h. die Benutzerprogramme, indem er ihnen den Prozessor zuteilt oder entzieht. Die Zuteilungs-Strategie ist wählbar, so daß z. B. Mono- und Multiprogramming-, Timesharing- und Multitasking-Betrieb möglich ist.

Der – optionale – Modul PRSC erlaubt die Einplanung von Prozessen, die Zeit- oder Ereignis-gesteuert ablaufen. Damit kann insbesondere das Echtzeit-Verhalten des Systems sichergestellt werden. PRSC reagiert auf externe Interrupts sowie die Echtzeit-Uhr und wirkt auf den Modul PRMA ein.

Zur Ein-/Ausgabe dient der Modul IOCS. Er führt, stark unterstützt von der Ebenen-Struktur des DIETZ 621,



alle Ein- und Ausgabevorgänge von und zu den Peripheriegeräten durch. IOCS umfaßt eine Reihe von Standard-Treibern, die bei Bedarf um weitere ergänzt werden; die gesamte Peripherie des Systems DIETZ 621 wird so von IOCS unterstützt.

Der ebenfalls zum Betriebssystem-Kern gehörende Modul FIAC steuert den Zugriff zum Plattenspeicher und verwaltet dessen Dateien.

Die Verwaltung der Betriebsmittel des Systems übernimmt der Modul REMA; mit Hilfe von Semaphoren werden Ressourcen angefordert und zugeteilt, die mehreren Prozessen zur Verfügung stehen.

Der Systemdialog wird mit Hilfe des Moduls DIAL durchgeführt. Er erlaubt – in Form einer Kommando-Sprache – dem Benutzer das Aufsetzen des Systems, Eingriffe, Abfragen, Zuteilung und Eintragung von Betriebsmitteln, den Aufruf von Dienstprogrammen und viele andere für den Betrieb des Systems wichtige Funktionen.

XOS-Moduln

Das Betriebssystem XOS kann bei Bedarf um eine Reihe problemspezifischer Moduln erweitert werden.

Dazu zählt unter anderem ein SPOOL-Modul, der die Ein- und Ausgabe von Daten in Verbindung mit dem IOCS steuert. Er puffert die Daten auf Spool-Dateien zwischen und sorgt für eine Entkopplung von Verarbeitungs- und Ein-/Ausgabe-Vorgängen. Dadurch werden unnötige Wartezeiten vermieden und die Betriebsmittel des Systems besser genutzt.

Typische weitere Moduln erlauben Hintergrund-Verarbeitung, die Verwaltung von Tasks, automatischen Anlauf bei Netzwiederkehr und andere Funktionen.

System Library

Die System-Bibliothek gehört zum Betriebssystem XOS im weiteren Sinne. Sie umfaßt eine Vielzahl von Moduln, von denen insbesondere die Formalsprachen zwecks Erweiterung der hardwaremäßigen Funktionen Gebrauch machen.

Dazu gehören arithmetische Operatoren für verschiedene Datentypen und -formate, Konversions-Routinen, mathematische Funktionen und andere Programmteile, die entweder als speicherresidente Bestandteile des Gesamt-Betriebssystems geführt oder von den Übersetzern in die lauffähigen Programme eingebunden werden.

Zur System-Bibliothek im erweiterten Sinne gehören auch die Dateiverwaltungs-Moduln RFMS, und DFMS.

Utilities

Einen festen Bestandteil des Betriebssystem XOS bilden die Dienstprogramme. Sie werden im System-dialog unter dem Sammelnamen SERV aufgerufen.

Hierzu gehören Programme für das Benennen, Initialisieren, Formatieren, Kopieren und Reorganisieren von Plattenspeichern, für das Listen von Dateiverzeichnissen, das Kopieren von Dateien bzw. deren Transfer von und zu anderen Peripherals und für die Ausgabe von Hauptspeicher- und Platteninhalten.

Weitere Dienstprogramme erlauben den Einbau vorhandener und die Definition neuer Betriebssystem-Moduln, die Abbildung der Hardware-Konfiguration im Betriebssystem und die Generierung des Systems im ganzen.

Das Dienstprogramm BATCH gestattet es, Systemkommandos und Parameter, die üblicherweise im Dialog eingegeben werden, von einer Datei zu lesen und in Form einer Stapelverarbeitung auszuführen.

XOS-Versionen

Das Betriebssystem XOS kann in mehreren Versionen generiert werden, die durch den Modul PRMA sowie durch Existenz und Gestalt des Moduls PRSC voneinander abweichen. Dadurch ergibt sich ein unterschiedliches Verhalten bezüglich der Prozeß-Verwaltung und -Einplanung. Welche Version gewählt wird, hängt von der Aufgabenstellung ab.

Bei der Version MPOS werden konkurrierende Prozesse mit hierarchischer Priorität verwaltet, wie es der Ebenen-Struktur des DIETZ 621 entspricht. Mit RTOS steht eine Erweiterung zur Verfügung, welche die Einplanung von Zeit- und Ereignis-gesteuerten Prozessen vorsieht.

XOS-Version	Betriebsart	Prozeß-Verwaltung	Prozeß-Einplanung
MPOS	Multiprogramming	hierarchisch	
RTOS	Realtime		vorhanden
TSOS	Timesharing	gleichrangig	
TROS	Times. + Realtime		vorhanden
MTOS	Multitasking	wählbare Priorität	vorhanden

Unter TSOS werden alle Benutzerprogramme gleichrangig nach einem optimierten Timeslice-Verfahren behandelt – sofern nicht explizit Prioritäten vergeben sind. Es stellt eine typische Betriebssystem-Version für mehrere, voneinander unabhängige Benutzer dar. Durch Einbau des Einplanungs-Moduls PRSC wird es zu TROS erweitert und gestattet dann auch die Behandlung von Echtzeit-Aufgaben.

Die Version MTOS ist ein echtes Multitasking-Betriebssystem. Hierbei werden Tasks nach vom Benutzer vergebenen Prioritäten verarbeitet und ihnen zugleich dynamisch Speicherplatz zugewiesen; darüber hinaus erlaubt MTOS die Einplanung von Echtzeit-Prozessen.

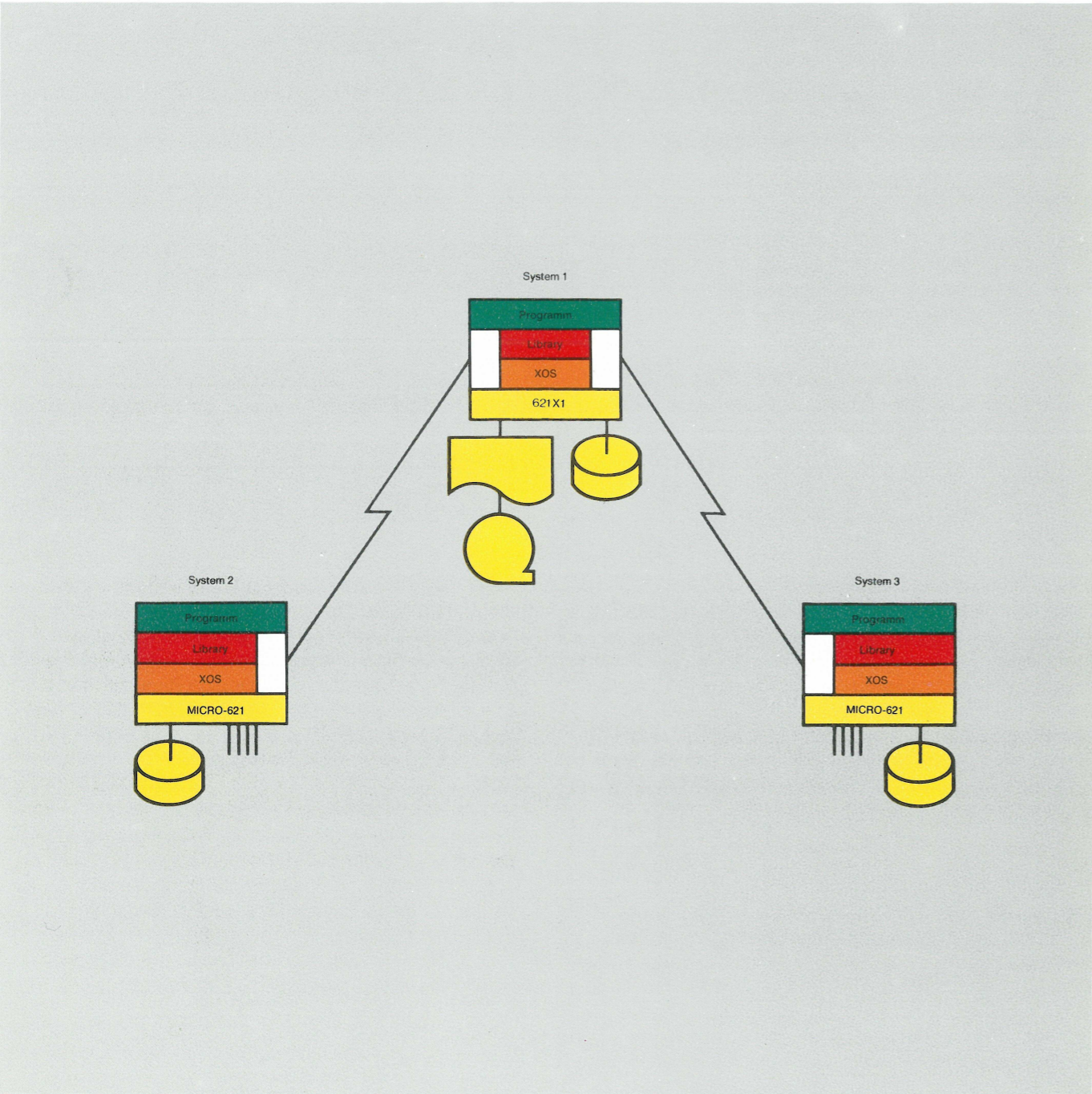
DIXOS

Mit DIXOS hat der Anwender die Möglichkeit, Systeme vom Typ DIETZ 621 untereinander zu verbinden und dadurch Rechner-Netze aufzubauen. DIXOS als „verteiltes Betriebssystem“ ist eine modulare Erweiterung von XOS, die hinsichtlich der Schnittstellen zu anderen Systemsoftware-Moduln und zu den Anwenderprogrammen voll kompatibel mit XOS ist.

Ziel von DIXOS ist die Kopplung von Systemen derart, daß jedes System im Netz zu den Betriebsmitteln der anderen so zugreifen kann, als ob sie ihm selbst gehörten. Das betrifft vor allem die Benutzung von Druckern, Magnetband-Laufwerken und anderen Ein-/Ausgabe-Geräten, aber auch den Zugriff zu Dateien auf Plattenspeichern eines anderen Systems.

DIXOS die Aktivierung von Programmen auf dem gekoppelten System, die Programm-Synchronisation und den Austausch von Daten zwischen Programmen, die auf den beiden Systemen laufen.

Der besondere Nutzen von DIXOS liegt in der Möglichkeit, besonders aufwendige Betriebsmittel nur an einer zentralen Stelle des Netzes vorzuhalten. Die Programme in den übrigen Systemen greifen mit normalen Mitteln auf diese Ressourcen zu, das heißt ohne Modifikationen der Anwenderprogramme. Die Umstellung der Datenwege, den prozeduralen Verkehr auf den Verbindungsleitungen zwischen den Systemen und die Durchführung der entfernten Transaktionen übernimmt DIXOS völlig selbständig.



Dateiverwaltung

Fortschrittliche Computer-Systeme enthalten als integralen Bestandteil einen oder mehrere Plattenspeicher, auf denen Programme und Daten zugriffsbereit sind. Verwaltung der Plattendateien, schneller Zugriff zu ihnen und optimale Strukturierung der Dateien setzen entsprechende Systemsoftware voraus.

Das System DIETZ 621 verfügt über ein mehrstufiges Dateiverwaltungssystem, das in den einzelnen Stufen unterschiedlichen Benutzer-Komfort bietet und problembezogen einsetzbar ist.

Der Modul FIAC ist als Bestandteil des Betriebssystems XOS die Basis aller übrigen Moduln und somit in jedem System enthalten. Er gestattet den Zugriff zu

Sektor-orientierten Dateien, in denen sich Programme oder Daten befinden. Wichtige Funktionen des Moduls FIAC sind ferner: Genereller Schreibschutz sowie Benutzer-bezogener Schutz von Dateien; Eröffnen, Verändern und Löschen von Dateien im Systemdialog oder durch das Programm; Dienstprogramme für das Kopieren von Daten und zur Reorganisation des Plattenspeichers sowie zur Ausgabe des Datei-verzeichnisses.

Mit dem Modul RFMS können Daten-Dateien aufgebaut werden, die in Sätzen beliebiger Länge strukturiert sind. Zu den Sätzen wird über deren Nummer

Modul	Datei-Typen Daten	Index	Datei-Struktur	Datei-Zugriff	Satzlängen physisch	logisch	Satz-Struktur	Feld-Zugriff	Feld-länge	Gast-Sprache
FIAC	x		Sektoren	Sekt.-Nr.						alle
RFMS	x		Sätze	Satz-Nr.	fest	variabel	Felder	sequentiell	fest	FORTRAN
DFMS	x	x	Sätze	Index-bezogen	fest	variabel	Felder	direkt sequentiell	fest	C-BASIC COBOL

zugriffen; die Felder innerhalb der Sätze werden sequentiell behandelt.

Einen wesentlichen Schritt weiter geht der Modul DFMS mit der Möglichkeit, Index-Dateien zu verwenden. Der Zugriff zu Datensätzen erfolgt – direkt oder sequentiell – über Schlüssel in einer oder mehreren Index-Dateien; daneben kann zu Datensätzen über deren Nummer oder in ihrer Folge zugegriffen werden. Komfortable Methoden sind auf die Schlüssel anwendbar; dazu gehören das Sortieren von Schlüssel, der Aufbau invertierter Dateien und die Reorganisation beliebiger Dateien.

## Programmiersprachen

Der Benutzer eines Computer-Systems DIETZ 621 hat eine Vielzahl von problemorientierten Programmiersprachen zur Auswahl. Er wird diejenige wählen, die Art und Umfang seiner Aufgabe möglichst angemessen ist. Gleichgültig, ob es um technisch-wissenschaftliche, kommerzielle oder Prozeßdatenverarbeitung geht oder um die Erstellung systemnaher Programme: Für jeden Problemkreis gibt es eine geeignete Sprache.

Die Programmiersysteme für die verschiedenen Sprachen enthalten gemeinsame und spezifische Komponenten. Dazu gehört

- ein Programm-EDITOR, der die Eingabe und Änderung von Quellprogrammen in Dateien auf Plattenspeichern unterstützt:

- je ein Übersetzer für die verschiedenen Programmiersprachen (siehe Tabelle), der das Quellprogramm auf formale Fehler prüft und eine bindefähige Form des Programms erzeugt;
- für die Sprachen BASIC, BASEX und PASCAL auch ein interaktives System, jeweils mit Editierung, Übersetzung und interpretativer Ausführung der Programme;
- ein Binder-Lader-System für BASIC, BASEX, C-BASIC und MARS;
- ein Binder und ein Lader als getrennte Moduln für COBOL, FORTRAN, P-FORTRAN, PEARL und PASCAL.

Sprache	Verwendung	Übersetzer	XOS-Version	Multi-tasking	Multi-user	Multi-language
BASIC	Technisch-wissenschaftliche DV, Ausbildung	Interpreter	TSOS		X	(2)
		Compiler				X
BASEX	Prozeß-Datenverarbeitung	Interpreter	RTOS/TROS	(1)	X	(2)
		Compiler				X
C-BASIC	Kommerzielle Datenverarbeitung	Int. Compiler	TSOS		X	(2)
COBOL	Kommerzielle Datenverarbeitung	Compiler	TSOS		X	X
FORTRAN	Technisch-wissenschaftliche DV	Compiler	TSOS		X	X
P-FORTRAN	Prozeß-Datenverarbeitung	Compiler	MTOS	X		
PEARL	Prozeß-Datenverarbeitung	Compiler	MTOS	X		
PASCAL	Wissenschaftliche DV, Ausbildung, Systemprogrammierung	Compiler	TSOS		X	(2)
		Dialog-Compiler				X
MARS	Systemprogrammierung	Assembler	alle		X	X

(1) im Rahmen RTOS/TROS

(2) gleichzeitig nur möglich: BASIC/BASEX-Interpreter  
oder C-BASIC oder Dialog-PASCAL

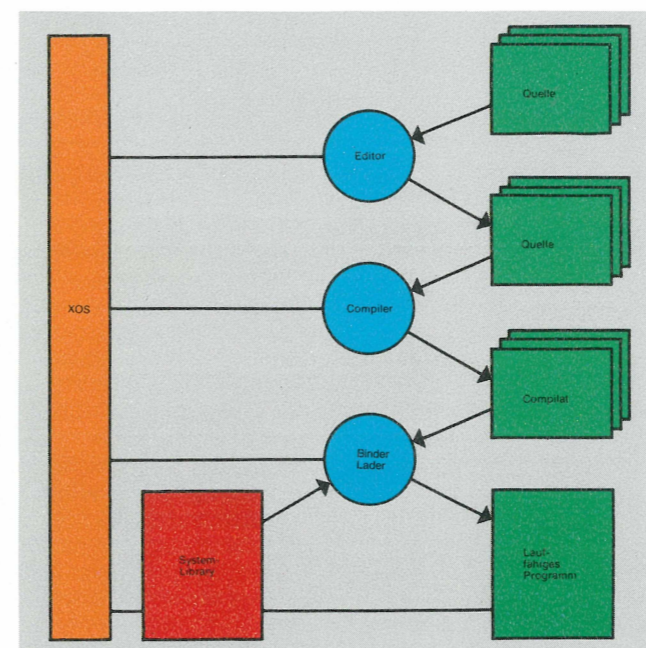
Alle Übersetzer basieren auf den Grundfunktionen des Betriebssystems XOS und erzeugen Programme, die die System-Bibliothek benutzen. Die Interpreter-Systeme lassen Mehrbenutzer-Betrieb zu. Dies stellt eine Erweiterung des Betriebssystems dar und unterstützt jeweils mehr als eine Benutzer-Partition. Die Compiler können nebeneinander in den Benutzer-Partitionen wie andere Programme laufen. Durch diese herausragenden Eigenschaften des Software-Systems können größere DIETZ 621-Systeme gleichzeitig mehreren Benutzern mit unterschiedlichen Aufgaben und Wünschen dienen.

Die Programme für Prozeßsteuerungszwecke können die Funktionen des XOS für Multitasking nutzen.

## Übersetzungssystem XOS

Für die Erzeugung ablauffähiger Programme, die aus formalen Programmiersprachen hervorgehen, steht ein einheitliches Übersetzungssystem zur Verfügung, das auf dem Betriebssystem XOS ruht. Es läuft in der jeweiligen Teilnehmer-Partition und läßt daher Mehrbenutzer-Betrieb zu. In der Regel wird es interaktiv verwendet, es kann jedoch auch im Stapel-Betrieb arbeiten.

Der Modul EDITOR gestattet das Erstellen und Verändern von Quellprogrammen beliebiger Sprachen; auf sehr komfortable Weise werden damit Programmteile erzeugt, die dann vom jeweiligen Compiler übersetzt und unter ihrem Namen vom System gespeichert werden.



Der Binder fügt die Programmteile zusammen und erzeugt ein ablauffähiges Programm. Außerdem bindet er die Operatoren, Funktionen und Routinen aus der System-Bibliothek ein, die vom Programm benötigt werden; daneben besteht auch die Möglichkeit residente und reentrante Teile der System-Bibliothek zu benutzen, die mehreren Teilnehmern zur Verfügung stehen. Der Lader sorgt für die gewünschte Allokierung der Programme im Hauptspeicher.

Das Übersetzungssystem XOS unterstützt nicht nur die Erzeugung von Programmen aus höheren Sprachen sondern auch von Assembler-Programmen.

## MARS

Das System DIETZ 621 verfügt über eine Vielzahl problemorientierter Programmiersprachen, von denen der Benutzer in aller Regel Gebrauch machen wird. Dennoch kann es gelegentlich notwendig sein, systemnahe Programm-Modulen sowie speicher- und laufzeitoptimale Routinen zu erstellen.

Hierfür dient der Assembler-MARS. Er übersetzt symbolische Programme, welche der Struktur des Computers und seinem Befehlsvorrat angepaßt sind, in Maschinencode. Er gestattet das Einbinden von

```

SEITE 1 SUBROUTINE DATE (FORTRAN)
USER 0 0 0 0 7:33

0
1 TESTPROGRAMM FUER FORTRAN ( SUBROUTINE DATE )
2 AUFRUF IN FORTRAN CALL DATE (11:20:22)
3 I>=INTEGERVARIABLE JAHR
4
5 0000 COM ." SUBROUTINE DATE (FORTRAN)"
6
7 0000 XOS ,1
8 0000 KC ,DATE
9 0000 BEGIN ;
10 0000 D ,0000 REGISTERDEFINITIONEN:
11 11 0000A TAG ; 0 ,0A AGR VON TAG
12 12 0000C MONAT ; 0 ,0C AGR VON MONAT
13 13 0000E JAHR ; 0 ,0E AGR VON JAHR
14 14 0000Z ZR ; 0 ,0Z ZWISCHENREGISTER FUER VARIABLENDNR.
15 15 0007A PARAB1 ; 0 ,7A AGR DES PARAMETERLOCKES
16 16 0000B PARABIN ; 0 ,0B AGR DES AUSGEWICHTLICHEN PARABLOCK
17 17 0003E RUECK ; 0 ,3E RUECKSPRUNG-REGISTER
18
19 PROGRAMMHAUPT
20
21 0000 LDAO,0 ,CTIM 1E 0C 0E 00
22 0004 ADCO,0 ,1 1E 00 00 00 AGR DES TAGESDATUMS
23 0008 STRO,0 ,TAG 1E 0A 0A
24 000B ADCO,0 ,1 1E 00 01 00 AGR DES MONATS
25 000F STRO,0 ,MONAT 1E 0A 0E
26 0012 ADCO,0 ,1 1E 00 01 00 AGR DES JAHRES
27 0016 STRO,0 ,JAHR 1E 0A 0E
28 0019 LDX ,0 ,PARAB1 02 7A
29 001B BNC ,0 ,3 ,ERR210 SG R3 23 FEHLER, WENN NICHT 3 PARAMETER
30 001E LDRD,PARABIN,PARAB1 1E 05 40 7A
31 0022 ADCO,PARABIN,1 1E 01 40 01 00 AGR DES TYP'S DES 1. PARAMETER
32 0027 LDX ,0 ,PARABIN 02 40

```

## BASIC

Die am Dartmouth-College in USA entwickelte Programmiersprache BASIC zeichnet sich durch ihren einfachen Aufbau und ihre leichte Erlernbarkeit aus. Sie wird für nicht zu umfangreiche technisch-wissenschaftliche Aufgaben sowie für Ausbildungszwecke verwendet. Besonders hervorzuheben ist die Möglichkeit, Programme interaktiv zu erstellen, zu verändern und zu testen.

Für den DIETZ 621 wurde BASIC um wesentliche Funktionen erweitert: Identifikatoren mit bis zu vier Zeichen, Datentyp „String“ mit entsprechenden Operatoren, Dateiverwaltung mit sektor-orientiertem Dateizugriff, Bedienung der gesamten Standard-Peripherie, Programm-Segmentierung.

```
5 CHAR A$(1024)
10 FOR I=1 TO 1024
20 LET A$(I)=CHR(0)
30 LET I=INT(100*0)
40 IF R>90 THEN 20
60 LET B=CHGR
75 LET A$(I,1)=B$(1,1)
80 NEXT I
90 OPEN(0,"ZEICH",AR0)
100 PPOS(AR0,0,0,0)
110 CLOSE(AR0)
10000 CHAR B$(1024)
10010 CHAR C$(1024)
10020 INPUT "DEV",X
10040 CHAR C$(1024)
10050 OPEN(0,"ZEICH",AR0)
10060 PPOS(AR0,0,0,0)
10070 LET A=0
10080 CHAR N$(0)
10090 LET N=XFFFFFFFFFFFF0000X
10100 LET ANZ=1024/0
10110 LET B=0
10120 GOSUB 10290
10130 LET C=0
10140 GOSUB 10290
10150 LET B=0
10160 GOSUB 10290
10170 LET A=1
10180 LET B=0
10190 PRINT DEV(X);B;A$(A,0);" "
10200 PRINT DEV(X);C$(A,0);" "
10210 PRINT DEV(X);B$(A,0);" "
10220 PRINT DEV(X);B$(A,0)
10230 LET A=A+0
10240 IF A=ANZ+0+1 THEN 10260
10250 GOTO 10100
10260 IF C=0 THEN 10310
10270 PRINT "UNTERSCHIEDLICHE STRINGS"
10280 END
10290 SOR(C,0,ANZ,N)
10300 RETURN
10310 PRINT
```

BASIC ist in Form eines Interpreters und eines Compilers implementiert; beide lassen – unter dem Betriebssystem TSOS – Mehrbenutzer-Betrieb zu. Während der Interpreter ein eigenes Editionssystem besitzt und damit den vollen Dialog-Komfort bietet, erzeugt der Compiler Objektprogramme, die mit höchstmöglicher Geschwindigkeit ablaufen.

## BASEX

BASEX ist eine vom Physikalischen Institut der Universität Freiburg entworfene Sprache für Echtzeit-Programmierung. Sie eignet sich für Anwendungen der Experiment- und Prozeßsteuerung kleinen bis mittleren Umfangs. BASEX ist interaktiv programmierbar, leicht verständlich und liefert ablauffähige Programme ohne große Umwege.

Der Sprachumfang von BASEX baut auf dem von DIETZ 621-BASIC auf. Hinzu kommen Sprachelemente für den Echtzeit-Betrieb, insbesondere für die Behandlung von Zeit- und Interrupt-Aufträgen, für die Synchronisation quasi-simultan ablaufender Prozesse mit unterschiedlicher Priorität und für die Behandlung der Prozeß-Peripherie.

```
LIST
1 REM PROGRAMM ZUR MESSUNG DER REAKTIONSZEIT BEI ANTWORTAUSWAHL
2 REM STIMULUS: EINE VON 3 LAMPEN: REAKTION DURCH INTERRUPTTASTE
100 REM EINGABE DER PARAMETER UEBER DIE TASTATUR
101 REM MITTLERE WARTEZEIT ZWISCHEN 2 REAKTIONEN -MWZ-
102 REM FEHLER WARTEZEIT NACH STIMULIERUNG -FWZ-
110 INPUT "MITTLERE WARTEZEIT ZWISCHEN ZWEI REAKTIONEN(>=2000HSEC)",MWZ
111 IF MWZ<2000 THEN 110
120 INPUT "FEHLERWARTEZEIT",FWZ
121 IF FWZ<1000 THEN 120
200 REM INITIALISIERUNG DES EXPERIMENTS
210 REM VERWENDETE VARIABLENWERTE AUF -0- STZEN
211 REM (EXP=ZAHLE RICHTIGER REAKTIONEN; FEN=ZAHLE FALSCHER REAKTIONEN;
212 REM SUM=AUFSUMMIERTE REAKTIONSZEIT)
215 LET EXP=0;FEN=0;SUM=0
220 REM ANZEIGE FELD LOESCHEN
225 LET OUTW(1)=0;OUTW(2)=0;OUTD(0)=0;OUTD(0)=0
230 REM START/STOP VORBEREITUNG DES EXPERIMENTES
235 PRINT "BEI TASTE 0 START; BEI TASTE 3 STOP DES EXPERIMENTS"
236 LET FLA=0;FLE=0
237 ON INT 0: LET FLA=1
238 ON INT 3: LET FLE=1
239 ENAB 0,3
240 WAIT FLA=1
300 REM DURCHFUEHRUNG DES EXPERIMENTES
310 REM ERRECHNUNG DER WARTEZEIT-MWZ- UND START DES WARTE-
311 REM ZEITINTERVALLS
315 LET MWZ=MWZ+(RAND(0)-.5)*1000
316 LET FLAG=0
317 AFTER MWZ: LET FLAG=1
320 REM ZUFALLSBESTIMMUNG DES STIMULATORS
321 LET LAMP=INT(3+RAND(0)*1)
322 GOTO LAMP OF 324,326,328
324 LET IRI=4;IFA1=5;IFA2=6
325 GOTO 330
326 LET IRI=5;IFA1=6;IFA2=4
327 GOTO 330
328 LET IRI=6;IFA1=5;IFA2=4
330 REM ERTEILUNG DES REAKTIONSAUFTRAGES
335 ON INT IRI: GOTO 410
336 ON INT IFA1: GOTO 420
337 ON INT IFA2: GOTO 420
340 REM AUSGABE DER STIMULATION AUF DAS LAMPENFELD NACH DER WARTEZEIT
345 WAIT FLAG
```

Für BASEX ist sowohl ein Interpreter als auch ein Compiler verfügbar. Beide können in Verbindung mit zwei Betriebssystem-Versionen arbeiten: Unter RTOS im Einbenutzer/Multiprogramm-Betrieb oder unter TROS im Mehrbenutzer-Betrieb.

## C-BASIC

Mit C-BASIC steht dem System DIETZ 621 eine äußerst komfortable, Dialog-orientierte Programmiersprache für kommerziell-administrative Anwendungen zur Verfügung. Sie bietet wie BASIC alle Vorzüge der leichten Erlernbarkeit, umfaßt darüber hinaus jedoch viele für den kommerziellen Einsatz wichtige Funktionen:

Zwei Datentypen „Zahl“ neben dem Datentyp „String“, spezielle Textstring-Operationen, Bildschirm-orientierte Sonderfunktionen und Sprachelemente für die Aufbereitung von Drucker-Ausgaben. Compound-States und Unterprogramm-Aufrufe mit Parametern erleichtern die Strukturierung der Programme.

```
40 REM FOR ARBE 57 000
20 LET EIN=5;OP=0
30 CHAR OP#13;OPTI#3
40 LET OP#1
50 REM
60 ON ERR=65 THEN RESUME 80
70 CREATE 1,"VERNON" WITH 50,240
80 OPEN 1,"VERNON" AS B
90 FOR I=0 TO 49
100 MODIFY (B,1)
110 FOR J=0 TO 7
120 WRITE (B)OP#1
130 FOR K=0 TO 8: WRITE (B)OP#1: NEXT K
140 NEXT J
150 NEXT I
270 PRINT
280 ON ERR=404 THEN RESUME 690
290 REM LET I=0
300 REM UPDATE(A,M)2:BMON:
310 REM PRINT "310":BMON:
320 REM IF BMON=1 THEN 680
330 REM RE=OP#13;OPTI#;ANID;154:AVR;
332 PRINT DEV(0)
335 INPUT "OPTI ? "OPTI#;"ANID ? "ANID;"AVER ? "AVER
337 PRINT DEV(3)
340 PRINT "340 "OPTI#;ANID;OP#
350 LET C=ANID-1
370 LET L=0
390 IF L=0 THEN PRINT "390 "L: GOTO 460
400 PRINT "400 "L;OP#;OPTI#;ANID
410 UPDATE (B,C)
413 READ (B)OP#
415 PRINT "415 "OP#
420 FOR I=0 TO 8: READ (B)OP#; PRINT OP#; " "
430 PRINT
440 PRINT "440 "OP#
450 GOTO 470
460 READ (B)L:OP#
470 PRINT "470 "OP#
480 PRINT "480 MARKE 3"
490 IF OP#="" THEN 540
500 IF OP#OPTI# THEN 660
510 PRINT "510 MARKE 2"
```

Besonders hervorzuheben ist das in C-BASIC verwendete Dateiverwaltungs-Modul DFMS, das unter anderem Mehrbenutzer-Zugriff, indexsequentielle Verarbeitung und Mehrfachschlüssel zuläßt. Hierzu gehört auch eine Vielzahl von Funktionen und Dienstprogrammen für das Sortieren und Reorganisieren von Plattendateien.

C-BASIC arbeitet unter TSOS im Mehrbenutzer-Betrieb.

## COBOL

COBOL ist die am weitesten verbreitete Programmiersprache und bietet Anwendern die Möglichkeit der Nutzung einer Vielzahl verfügbarer Programmpakete, insbesondere im Bereich der kommerziellen Datenverarbeitung. Der Vorteil dieser Sprache liegt in der Unterstützung verschiedener Dateioorganisationen, in der Verfügbarkeit mächtiger Instruktionen für Daten-Manipulation und Editierung und in der Verbesserung der Wartbarkeit von Programmen durch selbstdokumentierende Programmtexte. Der Sprachumfang von COBOL für das System DIETZ 621 ist streng am US-Standard von 1974 orientiert.

```
1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. BEISP.
3 ENVIRONMENT DIVISION.
4 CONFIGURATION SECTION.
5 SOURCE-COMPUTER. DIETZ-SYSTEM.
6 OBJECT-COMPUTER. DIETZ-SYSTEM.
7 INPUT-OUTPUT SECTION.
8 FILE-CONTROL.
9 SELECT DATE11 ASSIGN TO DISK: ORGANIZATION INDEXED;
10 ACCESS MODE DYNAMIC; RECORD KEY IS TEXT-IDENT.
11 SELECT DATE12 ASSIGN TO LINE-PRINTER.
12 DATA DIVISION.
13 FILE SECTION.
14 FD DATE11: LABEL RECORDS STANDARD.
15 01 DATE11-SATZ
16 02 TEXT-IDENT PIC X(4).
17 02 TEXTE PIC X(20).
18 FD DATE12: LABEL RECORDS OMITTED.
19 01 DATE12-SATZ PIC X(24).
20 WORKING-STORAGE SECTION.
21 77 MODUS PIC X(3).
22 PROCEDURE DIVISION.
23 PROGRAM-ANFANG. OPEN OUTPUT DATE12, I-O DATE11.
24 SCHLEIFEN-ANFANG.
25 DISPLAY "TEXT-IDENT: ", ACCEPT TEXT-IDENT.
26 DISPLAY "EIN-AUSGABE ? ", ACCEPT MODUS.
27 IF MODUS EQUAL "END" GO TO ENDE.
28 IF MODUS EQUAL "EIN" PERFORM TEXT-EINGABE.
29 IF MODUS EQUAL "AUS" PERFORM TEXT-AUSGABE.
30 IF MODUS EQUAL "SEQ" PERFORM TEXT-LISTE THRU LIST-ENDE.
31 GO TO SCHLEIFEN-ANFANG.
32 TEXT-EINGABE.
33 DISPLAY "TEXT: ", ACCEPT TEXTE.
34 WRITE DATE11-SATZ: INVALID KEY GO TO ENDE.
35 TEXT-AUSGABE.
36 READ DATE11: INVALID KEY GO TO ENDE.
37 WRITE DATE12-SATZ FROM DATE11-SATZ.
38 TEXT-LISTE.
39 START DATE11 KEY NOT LESS TEXT-IDENT: INVALID GO ENDE.
40 LIST-SCHLEIFE.
41 READ DATE11 NEXT: AT END GO TO LIST-ENDE.
42 WRITE DATE12-SATZ FROM DATE11-SATZ.
43 GO TO LIST-SCHLEIFE.
44 LIST-ENDE. EXIT.
```

Die Dateiverwaltung von COBOL geht wie bei C-BASIC über den Modul DFMS. Daher besteht keine Schwierigkeit, Dateien von C-BASIC- und von COBOL-Programmen aus zu bearbeiten.

Die durch den COBOL-Compiler generierten Programme werden durch den Binder in die ausführbare Form gebracht und sind unter allen Betriebssystem-Versionen lauffähig.

## FORTRAN

Der FORTRAN-Compiler des DIETZ 621 ermöglicht es, Programme in dieser weitverbreiteten Sprache für technisch-wissenschaftliche Anwendungen zu verarbeiten. Damit steht das System einer großen Benutzergruppe zur Verfügung, die gewohnt ist, ihre Probleme in FORTRAN zu lösen.

Der Sprachumfang entspricht Full FORTRAN gemäß DIN 66027, enthält jedoch eine Reihe von Erweiterungen. Diese betreffen insbesondere den Zugriff zu Plattendateien unter dem satzorientierten Dateiverwaltungssystem RFMS.

```
C PROGRAMM *** FOR023 ***
C
  DIMENSION IX(80),TITEL(20)
  WRITE(3,5)
  5 FORMAT('///ADRESSENPROGRAMM///')
  1 READ(8,10) TITEL
  10 FORMAT(20A4)
  IF(TITEL(1).EQ.TITEL(2)) STOP
  READ(8,15) I AUS, ITR, IBL
  15 FORMAT(15,2(4X,A1))
  WRITE(3,16) TITEL, I AUS, ITR
  16 FORMAT(1X,20A4/// ' AUSGABEEXEMPLARE', 15// ' TRENNZEICHEN'
  *4X,A1///)
  20 READ(8,25) IX
  25 FORMAT(80A1)
  IF(IX(1).EQ.IBL.AND.IX(2).EQ.IBL) GOTO 1
  K1=0
  30 30 L=1,80
  IF(K1.NE.0) GOTO 26
  IF(IX(L).EQ.ITR) K1=L-1
  GOTO 30
  26 IF(IX(L).EQ.ITR) GOTO 31
  30 CONTINUE
  31 K2=L-1
  K3=K1+2
  K4=K2+2
  30 40 L=1, I AUS
  WRITE(3,45) (IX(I),I=1,K1)
  WRITE(3,50) (IX(I),I=K3,K2)
  WRITE(3,50) (IX(I),I=K4,80)
  40 CONTINUE
  45 FORMAT('///IX,80A1')
  50 FORMAT('IX,80A1')
  GOTO 20
  END
```

FORTRAN-Programme können mit dem EDITOR interaktiv erstellt, übersetzt und mit dem Bindelader zu ungeteilten oder segmentierten Programmen zusammengefügt werden, wobei Teile aus der System-Bibliothek mit eingebunden werden.

Unter TSOS können mehrere Benutzer gleichzeitig und unabhängig voneinander FORTRAN-Programme erstellen, verändern, binden, laden und zur Ausführung bringen.

## P-FORTRAN

P-FORTRAN ist ein Programmsystem, das die Realisierung komplexer Echtzeit-Anwendungen gestattet. Es wurde in Anlehnung an den Vorschlag „Prozeß-FORTRAN“ der VDI/VDE-Gesellschaft für Meß- und Regeltechnik für Prozeßrechner vom Typ DIETZ 621 entwickelt.

Kennzeichnend für P-FORTRAN ist, daß der Anwender mehrere voneinander unabhängige Programme (Tasks) wählbarer Priorität definieren kann, die Zeit- oder Ereignis-gesteuert ablaufen. Spezielle, in das Betriebssystem eingebundene Moduln verwalten die Tasks, teilen ihm dynamisch Speicherraum zu, planen ihren Ablauf ein und synchronisieren sie bei Bedarf miteinander.

```
C HAUPTPROGRAMM HPREA
C
  DIMENSION IW(5)
  DATA A/4HREAD/
  DATA B/4HREAD/
  DATA EN/1H/
  RL=25474
  WRITE (8,10)
  5 WRITE (8,100)
  CALL OUT (8,1)
  1 CALL CON (8,1,1J,62)
  CALL OUT (8,0)
  CALL MSG (39)
  CALL UNCON (8,1,1J)
  CALL MSG (39)
  IT=RND(8)*5+1
  CALL WAIT(1T,2,1J)
  CALL CON (A,8,1J,63)
  CALL OUT (1,0)
  CALL TIME (HH,MM,MS,BS)
  CALL MSG (40)
  CALL MSG (40)
  CALL TIME (HH,MM,MS,BS1)
  CALL UNCON (A,8,1J)
  CALL OUT (8,0)
  IZ=IZ+1
  IW(IZ)=BS1-BS
  CALL OUT (IW(IZ),1)
  WRITE (8,201) IZ,IW(IZ)
  IF (IZ<5) IZ=1
  2 N=(IW(1)+IW(2)+IW(3)+IW(4)+IW(5))/5
  WRITE (8,202) N
  WRITE (8,102) N
  READ (8,101) ENDE
  IF (ENDE.NE.EN) GOTO 4
  3 IZ=0
  GOTO 5
  4 WRITE (8,200)
  100 FORMAT (1H,'ANFANG REAKTIONSTEST')
  102 FORMAT (1H,'ENDE Y/N ?')
  101 FORMAT (A1)
  201 FORMAT (1H,'REAKTIONSZEIT IN ',11,' VERSUCH : ',14,' MSEC')
  202 FORMAT (1H,'DURCHSCHNITTICHE REAKTIONSZEIT **** ',14,
```

Darüber hinaus gestatten spezielle Sprachelemente in P-FORTRAN die Behandlung der Prozeßperipherie.

P-FORTRAN arbeitet unter der Betriebssystem-Version MTOS, die uneingeschränktes Multitasking zuläßt.

## Basis PEARL

Die Programmiersprache PEARL setzt sich zum Ziel, Programme für umfangreiche Echtzeit- und Prozeßanwendungen zu formulieren. Durch die Aufteilung in System- und Problemteil sind PEARL-Programme weitgehend portabel, das heißt unabhängig vom verwendeten Computer-Typ.

PEARL gestattet die Formulierung von Tasks, die vom System in einer vom Benutzer vorgegebenen Weise verwaltet werden, sowie die Behandlung der Prozeßperipherie – typische Eigenschaften einer Prozeßsprache. Darüber hinaus weist PEARL viele Eigenschaften einer modernen Programmiersprache auf: Problembezogene Datentypen und verschiedene Datenstrukturen, Blockstruktur des Programms, Prozedur-Konzept, elementare Kontrollstrukturen.

```
0 MODULE PALL; /* PALL */
1
2 SYSTEM;
3 DP: <-> DEV=0;
4
5 PROBLEM;
6 SPECIFY OP DEVICE INOUT;
7 DECLARE SA FILE EXTERNAL INOUT SEQUENTIAL (*) CHAR(1);
8
9 MAIN: TASK;
10 CREATE SA UPON DP; OPEN SA;
11 ALL 10 SEC
12 ALL 15 SEC UNTIL 1:0:0 ACTIVATE AUSGABE1;
13 ALL 20 SEC DURING 1 HRS ACTIVATE AUSGABE2;
14 END;
15
16 AUSGABE1: TASK Prio 1;
17 PUT FROM 'TASK, AUSGABE1' TO SA THRU A(14);
18 END;
19
20 AUSGABE2: TASK Prio 2;
21 PUT FROM 'TASK, AUSGABE2' TO SA THRU A(14);
22 END;
23
24 AUSGABE3: TASK Prio 3;
25 PUT FROM 'TASK, AUSGABE3' TO SA THRU A(14);
26 END;
27
28 MODEND; /* PALL */
```

Die Implementierung auf dem System DIETZ 621 entspricht der 1977 von KFK/PDV herausgegebenen Sprachbeschreibung „Basis-PEARL“ und umfaßt den Compiler sowie eine Reihe von Bibliotheksfunktionen.

Basis PEARL arbeitet unter der Betriebssystem-Version MTOS mit uneingeschränktem Multi-Tasking.

## PASCAL

Die von Niklaus Wirth entworfene Programmiersprache PASCAL zeichnet sich vor allem dadurch aus, daß sie – bei Beschränkung auf ein Minimum von Sprach-elementen – zur Formulierung wohlstrukturierter Programme anleitet und die Verwendung problembezogener Datentypen und -strukturen ermöglicht. Deshalb wird PASCAL vor allem in der Informatik-Ausbildung, aber auch für Aufgaben der wissenschaftlichen Datenverarbeitung und systemnaher Programmierung eingesetzt.

PASCAL ist als Compiler verfügbar, der — in Verbindung mit EDITOR, Binder und Lader — unter TSOS mehreren Benutzern das Erstellen, Übersetzen, Laden und die Ausführung von Programmen ermöglicht.

```
1 PROGRAM TESTEWIG(INPUT,OUTPUT);
2 TYPE
3 BYTE=0..255;
4 BYTE2=0..FFFF;
5 STRING10=ARRAY[1..10] OF CHAR;
6 VAR
7 JAHR:INTEGER;
8 TAG,MONAT:BYTE;
9 WOCHENTAG:ARRAY[0..6] OF STRING10;
10
11 INITPROCEDURE;
12 BEGIN
13 WOCHENTAG[0]='SONNTAG';
14 WOCHENTAG[1]='DINSTAG';
15 WOCHENTAG[2]='MITTWOCH';
16 WOCHENTAG[3]='DONNERSTAG';
17 WOCHENTAG[4]='FREITAG';
18 WOCHENTAG[5]='SAMSTAG';
19 WOCHENTAG[6]='SONNTAG';
20 END;
21 (* INIT *)
22
23 FUNCTION EWIG(T,M:BYTE; J:BYTE2):BYTE;
24 VAR
25 Z,V,S:INTEGER;
26 BEGIN
27 Z:=0;
28 IF (M=1) OR (M=2)
29 THEN V:=31*(M-1)
30 ELSE
31 BEGIN
32 IF J MOD 4 = 0
33 THEN
34 IF (J MOD 100 = 0) AND
35 (J MOD 400 = 0) OR
36 (J MOD 100 <> 0)
37 THEN Z:=1;
38 V:=(306*M-324) DIV 10;
39 END;
40 EWIG:=(Z+365*(J-1)+(J-1) DIV 4 - (J-1) DIV 100 + (J-1) DIV 400
41 +V+T) MOD 7;
42 END;
43 (* EWIG *)
44
```

Darüber hinaus wird Dialog-PASCAL angeboten; es umfaßt einen reentranten Compiler und ein BASIC-verwandtes Editorsystem. Dialog-PASCAL bietet hohen Benutzerkomfort hinsichtlich Sicherheit und Geschwindigkeit; die Implementierung auf dem System DIETZ 621 entspricht weitgehend dem Sprachumfang, wie er im Feoll-Bericht »PASCAL-E« niedergelegt ist.

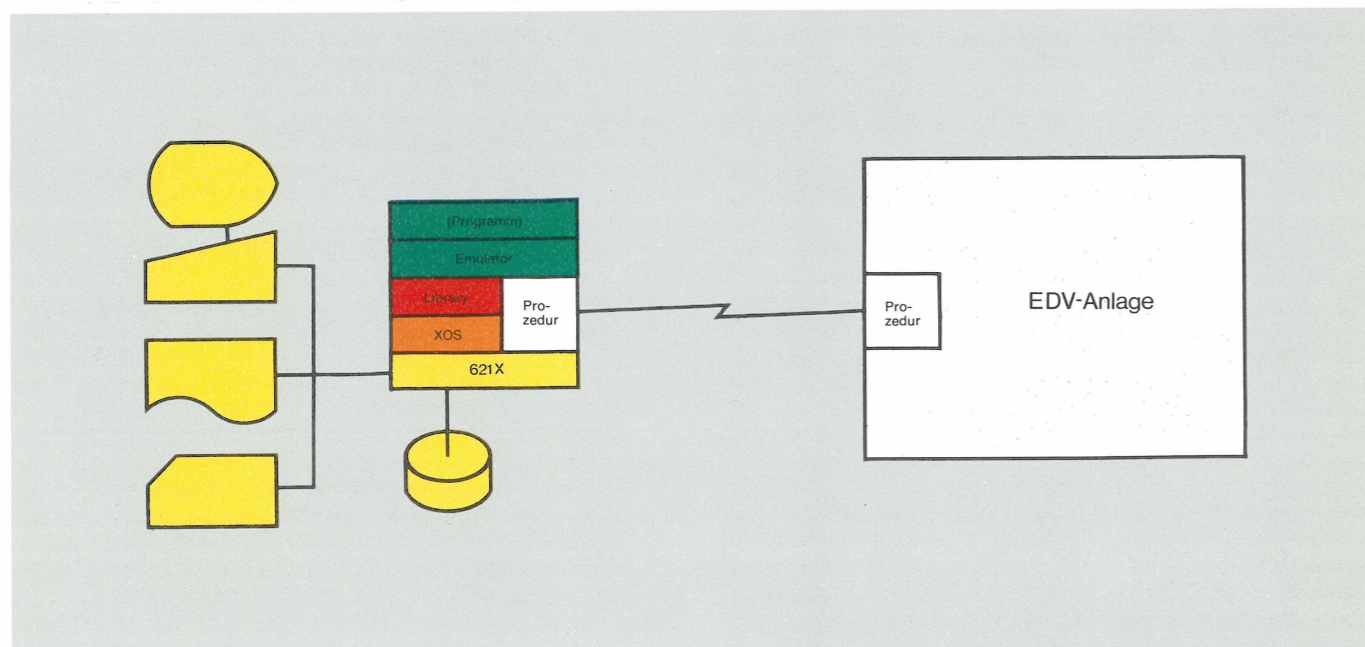
## Datenfernverarbeitung

Es gehört zum Stand der Technik, Rechner-Systeme über größere Entfernungen miteinander zu verbinden, um Daten und Programme zwischen ihnen auszutauschen. Minicomputer wie der DIETZ 621 machen hiervon keine Ausnahme, indem sie entsprechende Hardware- und Software-Werkzeuge bereitstellen.

Zunächst besteht die Möglichkeit, Systeme vom Typ DIETZ 621 miteinander über Wähl- oder Standleitungen mit Übertragungsgeschwindigkeiten bis 9600 bit/s zu koppeln. Hierfür stehen DFÜ-Prozeduren zur Verfügung, die für einen sicheren Transport der Information über die Leitung sorgen. Sie haben einfach handhabbare Schnittstellen zu den Anwenderprogrammen und gestatten blockweise Übertragung

alphanumerischer Informationen in beiden Richtungen.

Darüber hinaus verfügt DIETZ über eine Reihe von DFÜ-Prozeduren und Terminal-Emulationen, die den Anschluß an übergeordnete EDV-Anlagen nahezu aller marktüblichen Fabrikate ermöglicht. Dazu gehören gängige Halb- und Vollduplex-Leitungsprotokolle ebenso wie Emulatoren für Standard-Terminals der EDV-Anlagen, die entweder im Stapel- oder im Dialog-Betrieb arbeiten. Hierbei ist besonders hervorzuheben, daß RJE- und Dialog-Terminals auf der Basis des DIETZ 621 entweder alternativ oder sogar — im Rahmen der verfügbaren Kapazität — simultan lokale Aufgaben der technisch-wissenschaftlichen, kommer-



ziellen oder Prozeß-Datenverarbeitung durchführen und somit wirtschaftlich optimal genutzt werden können.

## Anwendungsorientierte Systeme

DIETZ verfügt mit dem System 621 nicht nur über eine Vielfalt von Werkzeugen zur Lösung von Anwender-Problemen, sondern bietet darüber hinaus viele spezifische Hardware- und Software-Komponenten bis hin zu schlüsselfertigen Systemen, welche die fertige Problemlösung enthalten.

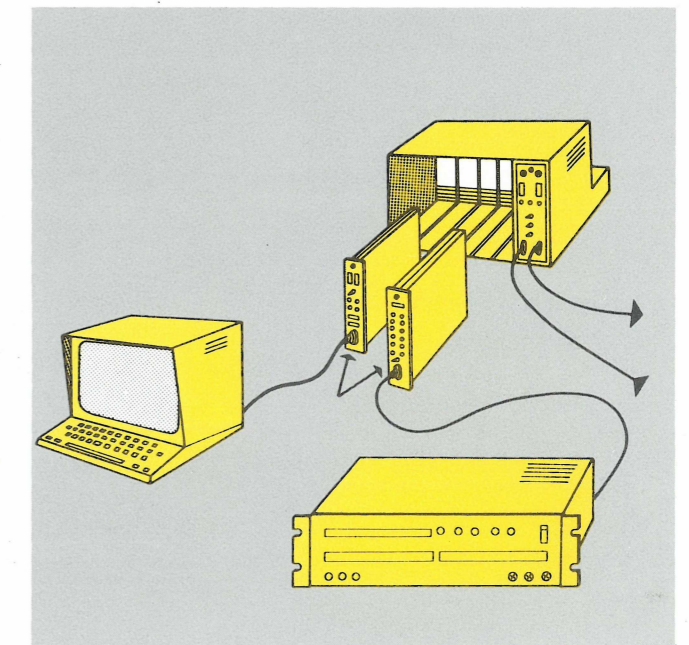
Software-Pakete unterstützen eine Vielzahl graphischer Ein- und Ausgabegeräte wie Plotter, Digitizer und graphische Terminals.



Für Zwecke der Laborautomatisierung und Meßwertverarbeitung sind parametrierbare Erfassungs- und Auswertungs-Programme sowie mathematisch-statistische Pakete verfügbar, die unter Echtzeit-Bedingungen einsetzbar sind und in umfangreichere Anwender-Systeme eingebunden werden können.

Auf besondere Anwendungen zugeschnittene Hardware-Komponenten wie funktionsorientierte Terminals, Ringleitungs-Anschlüsse und dezentrale Stationen unterstützen Aufgaben der Betriebsdaten-Erfassung, Fertigungstechnik und Prozeßkontrolle.

Außerdem liefert DIETZ, ebenfalls auf der Basis des Systems 621, schlüsselfertige Systeme für eine Vielzahl von Anwendungsgebieten im Bereich der Produktions-Steuerung.



Labor- und Prozeßautomatisierung, Informationsverarbeitung, Datenferverarbeitung und Nachrichtenverwaltung.

Eine wichtige Ergänzung des Anwendungs-Spektrums bilden die interaktiven CAD-Systeme des Tochter-Unternehmens Dietz TECHNOVISION GmbH.

## Dienstleistungen

Der einfache Umgang mit modernen Computer-Systemen darf nicht vergessen machen, daß es sich um komplexe technische Systeme handelt. Daher ist ihr sinnvoller und störungsfreier Betrieb nur dann gewährleistet, wenn der Hersteller des Systems dem Anwender ein Mindestmaß an Unterstützung bietet.

## Beratung

Qualifizierte Mitarbeiter in den DIETZ-Geschäftsstellen beraten den Interessenten, welche Konfiguration für seinen Betrieb am günstigsten ist, welche späteren Ausbaustufen sich ergeben, welche Programmiersprache der Anwendung am besten entspricht und welche Anschaffungs- und Folgekosten auf ihn zukommen.



## Systemunterstützung

Auch bei Standard-Komponenten gibt es Fragen zu den Schnittstellen von Hardware und Software, zu Aufstellungs- und Umgebungsbedingungen des Rechners und anderen Problemkreisen. Bei all diesen auftretenden Fragen ist es von großem Nutzen für den Anwender, wenn er einen qualifizierten Systemberater des Herstellers als Ratgeber zur Seite hat.

Bei DIETZ steht hierfür die Kundenbetreuung zur Verfügung; sie besitzt langjährige Erfahrung im Einsatz von Computer-Systemen in allen Bereichen.

## Schulung

Das DIETZ-Schulungszentrum hält regelmäßig Kurse ab, in denen sich Benutzer von DIETZ-Systemen im Umgang mit dem Computer und in allen verfügbaren Programmiersprachen unterweisen lassen können.

## Aufstellung und Einweisung

DIETZ 621-Systeme werden von erfahrenen Fachkräften am Einsatzort aufgestellt und in Betrieb genommen.

Dabei wird das Personal des Benutzers ausführlich in die Bedienung des Systems, die Handhabung der System-Software und in die Behandlung der Geräte-Peripherie eingewiesen.

## Kundendienst und Wartung

Zwar sind Minicomputer wie der DIETZ 621, die sich im harten industriellen Einsatz bewährt haben, sehr robust und wenig stör anfällig. Dennoch erfordern auch sie ein Minimum an Pflege und, im Falle der Störung, deren schnelle Beseitigung.



Die Kundendienst-Techniker in den Geschäftsstellen erreichen jeden Aufstellungsort in kurzer Zeit, um eventuelle Fehler zu lokalisieren und zu beheben. Darüber hinaus kann ein Wartungsvertrag abgeschlossen werden, der dem Anwender die dauernde Bereitschaft und lange Betriebsfähigkeit des Systems durch regelmäßig durchgeführte Wartung garantiert.

## Dokumentation

Zu jedem an den Endbenutzer gelieferten System gehört eine ausführliche Dokumentation über die Systembedienung, die Behandlung von Fehlern sowie insbesondere über die verwendeten Programmiersprachen.

## Das Unternehmen DIETZ

1951 gegründet und seitdem auf dem Gebiet der Automatisierung tätig, begann das Unternehmen vor über zehn Jahren mit der Entwicklung und Herstellung von Prozeßrechnern. Heute zählt DIETZ mit über 2500 installierten Systemen zu den erfolgreichsten europäischen Herstellern von Minicomputern.

DIETZ ist professionell und ausschließlich auf dem Gebiet der Minicomputer tätig. Etwa 450 Mitarbeiter entwickeln, fertigen und vertreiben das System 621, das seit 1971 stetig zu einem universellen Computer-System für Prozeß-Anwendungen, für technisch-wissenschaftliche Zwecke, für die Datenfernverarbeitung und, in abgewandelter Form, auch für die kommerzielle Datenverarbeitung ausgebaut wurde.



Hauptsitz von DIETZ ist das Werk in Mülheim a. d. Ruhr; Mitarbeiter in sieben Niederlassungen helfen Anwendern in der Bundesrepublik mit Beratung und technischem Kundendienst. Auch in den Nachbarländern ist für die Wartung von DIETZ-Systemen gesorgt.

ANMERKUNG:  
Der Nachdruck, auch auszugsweise, ist nur mit Genehmigung der Firma DIETZ-Computer-Systeme gestattet.  
Diese Beschreibung hat lediglich Informationscharakter, Gewährleistungsansprüche daraus sind ausgeschlossen. Die Sachverhalte können jederzeit ohne Ankündigung geändert werden. Diese Beschreibung ist nicht Gegenstand eines Vertrages.

Heinrich Dietz  
Solinger Straße 9  
4330 Mülheim-Ruhr  
Tel.: (0208) 44 34-1  
Telex 856770

**DIETZ**

**Computer  
SYSTEME**

3-7909-02-051