

Product Specification

DRAFT

APR 1 3 1979

TABLE OF CONTENTS

Introducing GDS-II.....	1
Database	3
System Operation	6
New Products	15
Background Functions	19
Multiground RDOS	24
Hardware	29
Future Products.....	38

INTRODUCING GDS-II

In the intensely competitive semiconductor industry, rapid implementation of new technology means success. This demands a production graphic design system that works swiftly, correctly, and without interruption. GDS-II, CALMA's Graphic Design System, meets these criteria.

Standard GDS-II features and facilities include:

- Complete on-line design capabilities
- A VLSI database capable of resolving features to one part in 4 billion.
- Multiground, real time disk operating system allowing up to 15 simultaneous background operations, while running up to 6 work stations in the foreground.
- Dual screen work stations featuring alphanumeric as well as graphic displays
- Databases with information content extended beyond graphics
- User definable extensions to databases
- Consistent, concise command language, adaptable to the needs of novice and expert operators
- Powerful edit commands for fast data manipulation
- Foolproof, symbolized command menu
- Command language extended by GPL-II Programming Language
- DG/L, ALGOL, FORTRAN, and assembly language user programming
- Wide range of plotter support, including high-speed matrix plotter software
- Optical and e-beam pattern generators with all-angle formatting.
- High-performance ECLIPSE computer with memory mapping and floating-point processor

- Three customized graphics stations for a variety of IC applications.
- High-speed, dual-density magnetic tape transport and 300 megabyte storage module drive

General Description

GDS-II provides the ideal interface between designer and computer. It augments the designer's talents by conforming to his methods of thinking and working. Drawing from experience with GDS, (the IC industry leader for the past seven years), GDS-II is an on-line design system that significantly enhances the design cycle from layout through pattern generation or electron-beam output.

By providing On-Line Design--actually designing the circuit on the computer system--GDS-II eliminates the tedious, time-consuming process of drawing by hand. The system features graphics terminals for designers, software aids to enhance design implementation, background processing for analysis programs and output formatting, and application capabilities to tailor the system to the specific needs of the user. GDS-II is a total system applicable to all areas of graphic IC development.

GDS-II is simple to operate. All commands can be symbolized or expressed in plain English. Every design station features an alphanumeric, as well as a graphics, CRT. The alphanumeric display keeps the designer in constant communication with the system, prompting every step. To reduce mistakes and confusion, on-line Help files are available to explain all commands.

Standard Features

GDS-II features the precision necessary for VLSI design and production. With Data General's RDOS-based multiground operation, the GDS-II system handles design graphics, program development, rules checks, plotting, and pattern generation output concurrently. The expandable database allows additional user defined parameters and provides easy access to data and commands. The GDS-II Database Management System features unique mechanisms for data compression and data

classification, both of which minimize disk requirements. Multiple working libraries facilitate efficient file management. Powerful edit commands allow fast modification of data. Versatile design rule checks minimize errors. GDS-II is the only system with all-angle optical and electron-beam pattern generation output.

Program development is available in FORTRAN, ALGOL, DG/L, Basic, Assembly, and GPL-II (a high-level graphics applications language operating within GDS-II).

Hardware Configuration

The basic GDS-II system has a Data General Eclipse CPU with 300 megabyte disk and dual density magnetic tape drive, keyboard printer, digitize or design station, and on-line plotter. The maximum system can support up to four disks (which gives a range of up to 1200 megabyte disk space), adds a modem for CPU/CPU link, and may have up to six design stations, three on-line plotters, an on-line Versatec, and a line printer. All stations can operate simultaneously. (See the Hardware Section of this manual for more information.)

GDS-II Precision

VLSI (Very Large Scale Integration), which is a reality in today's advanced integrated circuit technology, requires significantly improved precision and the ability to handle much larger databases.

CALMA's previous software, GDS, made use of a 16 bit database. This database provided a software matrix consisting of 32,000 data points in each axis. Dimensions could be resolved to $\frac{1}{2}$ of a micron with circuit sizes approaching 300 mils. This was adequate for most LSI requirements.

GDS-II features the precision necessary for VLSI design and production. A 32-bit database provides a software matrix with over 4 billion data points in each axis--a working area over 130,000 times greater than that of GDS. In terms of measurement, a shape can be resolved into increments of $\frac{1}{10,000}$ of a micron. At that resolution, the circuit size could still approach $\frac{1}{2}$ meter in length. The database has the precision necessary to place, rotate, and position any shape without round-off. It can accommodate every IC technology and guarantee perfect resolution over the entire circuit, no matter how large.

DATABASE

Database Management System

To meet the growing needs of IC design, CALMA's GDS-II Database Management System (DBMS) performs all database management functions, (including creation, modification, retrieval, and explosion), for the entire GDS-II system. The DBMS is composed of a standard set of procedures that can be used by both interactive and background processes.

Extensibility

Growing IC design requirements have demonstrated the need to shift from databases that concentrate strictly on graphics toward full-design databases. GDS-II system objectives (such as the capability to verify that final artwork conforms to design specifications) require convenient means to supplement standard graphic information. Experience has also revealed the tendency for each application area (and even installation) to possess its own special data requirements. Therefore, one of the primary design goals of the GDS-II Database Management System is to provide the means for creation and maintenance of simple, flexible database structures. This extensibility facilitates both initial system development as well as user customization.

Besides normal graphic information, CALMA's DBMS allows the user to assign additional data to the standard graphic entities. This information may be additional text for shape description, or special nodes or descriptors for access from other programs. Additional data assignment is possible because each shape within the graphics file has its own individual file, which can be expanded to contain any information pertinent to that shape. Individual shape files can also be linked to other shapes and their accompanying files (i.e., for continuity checking, every device tied to a particular signal can be linked to that signal or node.)

Database Construction

Because it is often desirable to build a circuit from a standard set of geometric figures, GDS-II databases are organized with this feature in mind. GDS-II database hierarchy consists of "Libraries" which are composed of "structures" which in turn are composed of "elements".

Elements

The basic contents of the GDS-II database are its elements. An instance of an element corresponds to an instance of a geometric entity, such as a closed polygon or a reference to a collection of elements. Elements are defined in terms of their properties". An element, then, is defined as an arbitrary collection of properties.

Within the database, each element is associated with one of a number of separate classes, which are called "coordinate property modes". These modes include:

- Path--An open polygon with width (digitized by a center line)
- Boundary--A closed polygon
- SREF--Structure reference
- AREF--Array reference
- Text--Complete texting capability

Various prenotations are associated with different coordinate property modes. For example, paths and boundaries both have layer and datatype (classifies data and allows it to be selected). Other prenotations include width, angle, magnification, reflection, and array shape (specifies the number of rows, columns, and spacing of a shape to be arrayed, and character string data (undisplayed text consisting of up to 64 characters).

Any element can be assigned to any of 64 available layers. Layers can be easily differentiated by using line coding, which allows shapes to be displayed as dotted, dashed, broken, or solid lines. Various fill codes are also available to highlight individual shapes. In addition, there are 64 different datatypes available per layer. These datatypes are associated with a table of 0 through 63 and can be used to classify data, (i.e., for specific identification of certain shapes, or layers, and access information used by other programs). Datatypes can be assigned to any shape(s) on any layer(s). An example of the use of datatypes would be classifying particular transistor types within an integrated circuit. For example, metal layer for PG output might be on layer six. By adding a

datatype, VDD can be classified as layer six (metal), datatype (1). Datatype 1 may then be selected for viewing and the VDD connections will be easily seen. The same classifications can be done for clock lines, ground, and special nodes or devices.

The DBMS automatically classifies elements according to their properties. This classification insures that the exact nature of an element retrieved from the database is available as soon as the program interface begins.

Any definable property may be added to an element, and database extension depends primarily on the users' ability to create and assign new properties to the basic elements.

Structures

The DBMS allows for any group of elements to be organized into structures. A structure, then, is a collection of elements with a user-assigned name. The DBMS records the time of creation and time of last modification for each structure.

Structures provide a convenient way to build hierarchies of geometries into increasingly complex assemblies. For example, a structure is placed within another structure by creating an element that names the referenced structure and specifies its origin(s). Structures can call other structures, or be arrayed. GDS-II allows up to 10 levels of structure nesting. (During placement, structures may be magnified, demagnified, and rotated at any angle.)

For example, when building a database that represents a circuit, a user might choose to define the geometries of transistors, capacitors, and resistors as basic structures. These would be referenced within structures to form logic gates, and gates further referenced to form logical functions.

Libraries

Collections of structures are organized by GDS-II's DBMS into libraries. These libraries contain all data associated with a given structure or group of structures. This information includes graphic data on shapes, their associated text and descriptors, and all references to other data.

The structure within a given library may reference structures within a second, designated library. Structures within the second library may reference

those within a third. Each library is maintained as a separate RDOS file; therefore, the user is not logically constrained by how many libraries may be resident within a GDS-II system at one time.

The only limit to the number of libraries that may reside concurrently within a GDS-II system is the limit enforced by the physical size of disk storage.

Any library can be opened as either a "Working" or "Reference" Library. Working Libraries contain the majority of data in production. (Typically each circuit is contained as a separate, Working Library.) Users may access these Libraries to edit data or make copies for editing.

Reference Libraries contain frequently used data (such as standard structures). A read-only classification protects any Reference Library from alteration. Data may be copied for use or change without affecting the original database. Up to two Reference Libraries may be accessed at any one time.

If needed, the DBMS allows the user to combine (merge) libraries and view and edit any part of the interface between them. Libraries can be "write-protected" to prevent modification of the original data. Write-protected libraries, however, can still be copied.

Database Specifications

- All database management functions are performed by the GDS II Database Management System (DBMS).
- The DBMS allows the user to assign additional data to graphic information.
- Database elements can be classified into five modes:
 - Paths
 - Boundaries
 - SREF
 - AREF
 - Text
- Up to 64 layers and 64 datatypes are available for further element classification.

- A Structure is a collection of elements with a user-assigned name.
- Up to 10 levels of structure nesting are available.
- Libraries can be opened as either: Working or Reference
- More than one Reference Library can be designated, and two may be accessed at any one time.

SYSTEM OPERATION

The CALMA Design Terminal

The majority of all data input, design, and edit functions are implemented at a GDS-II design terminal. A typical terminal comes equipped with tablet and cursor (for command input and data manipulation), menu, alphanumeric keyboard, and dual displays. CALMA offers a variety of different terminals for different functions. (See the Hardware section for specific details.)

Command Language

The basic means for the designer to communicate with the system is through the GDS-II command language. Using this command language, the designer can create, present, modify, copy, select, store, output, and delete all graphic data.

Despite its powerful capabilities, GDS-II is definitely not a complicated language filled with obscure buzz words and irregular syntax. In fact, all GDS-II commands are in self-explanatory, plain English, (e.g., MOVE, ROTATE, DELETE), which makes the language easy to understand and learn. The basics can be mastered in a matter of hours. Moreover, GDS-II is logically structured to build a foundation for all operations. Its syntax is composed of a consistent set of rules which cover all command and parameter formats.

Since the system is controlled by button menus (Squares touched by the user to implement a command to the system), there is no real limitation on command name length. (The user has the option to

never be typing.) Any command can be called simply by touching the assigned button with the design terminal cursor. An entry any place within the square executes the command. The movement of the cursor is represented on the graphics screen by a flashing cross.

Command Menu

A "menu" is any group of selectable system command buttons; it is the major method of implementing commands to the system. Pictured below is one of the menus CALMA provides.

Each button on this menu stands for a user-defined command or macro (i.e., string of commands). Most of these have been symbolized for easy recognition.

All the commands on the menu are divided into five categories:

- data creation
- data modification
- numeric selection
- data display and output
- view control

CALMA menus can be any size (from 1 to 512 buttons) and any shape. In addition, there are up to four levels available, each containing its own set of commands.

A number of commands may be strung together as a macro and then assigned to a single button. All commands can be iterated at the touch of a button.

COMMAND MENU

MENU	DATA CREATION			DATA MODIFICATION							NUMERIC SELECTION				DATA DISPLAY				VIEWPORT CONTROL		
	DATA TYPE	WIDTH	LAYER	UN-DO	EDIT AREA	EDIT AREA	DELETE	EXPAND	ERASE	C	BACK SPACE	SPACE	÷	GRID	TEXT FILL	SNAP FILL	GROUP FILL	SREF WIDE	↖	↑	↗
CLOSE LIBRARY				90°	180°	270°	MIRROR	SPECIFY MACRO	EXECUTE MACRO	7	8	9	X	RESET DISPLAY	•				←	STOP	→
OPEN LIBRARY FOR EDIT				SELECT	UN-SELECT	CLOSE	NOTCH	CHOP	JOIN	4	5	6	-		B	D	P	OFF	↖	↓	↘
OPEN STRUCT. FOR EDIT	AREF	PATH		ADD	GROUP	COPY	OVER	RULER		1	2	3	+	ALL	NONE	REFER	EDIT	↖	FIT	↗	
DISPLAY SET UP	SREF	BOUNDARY		SELECT ITEM	STRETCH SIDE	RIGID MOVE	MOVE POINT	EDIT POLYGON		•	0	⇒	?	DISPLAY 1	DISPLAY 2	DISPLAY 3	DISPLAY 4	REDRAW	AUTO REDRAW		

Creation and modification of menu buttons is easily achieved using the GDS-II Text Editor (TED). Menu definitions may be saved and restored. Even though menus are easily produced if desired, CALMA provides a standard menu for the user.

Effective use of command menus is of major importance to on-line design. GDS-II can now actually display these menus directly on the screen. Users save valuable time because they never have to look away from their work to locate a particular command.

With this advanced on-screen capability, command buttons can be both exhibited along the top and sides of the viewing screen as well as being physically placed on the tablet under the user's hand. Any number of command buttons can be shown on the screen. However, users have the option to place frequently-used command buttons on the screen, leaving the other less-used commands on the tablet.

A software menu can be used at any station with user specified origin and size. Several menus, set with different configurations for different jobs, can be stored and recalled as needed. Different stations can use different menus.

Commands and macros can also be entered using a special function keyboard. Any macro that can be assigned to a menu button can be assigned to a function keyboard button.

With CALMA's Multicolor Raster Display (See New Products), menu graphics switch instantly as menu levels are switched, without redrawing the screen.

Command Line Echoing

All command input to the system is performed interactively.

The degree of appropriate interaction is directly proportional to the complexity of the task at hand. Minimum interaction consists of echoing all commands and parameters entered into the system. If processing at a design terminal is waiting for command input, all character prompts will be echoed immediately on the alphanumeric screen. If a menu button is activated or a point digitized, several characters can be echoed for each keystroke. Until a terminator character is entered, the contents of the input line can be corrected. In other words, the last character, the last coordinate, or even the whole line

can be deleted. Once the terminator character is entered, execution of the command begins. This immediate feedback is very important to an operator who is learning to use the system.

When a large circuit has been created, the experienced operator can use commands that require a large amount of data to be processed. In this case, it is not necessary to wait for the previous command to finish execution before entering the next command. Even though the previous commands have not been echoed, the operator can go ahead with command input. The echoed command line that appears on the alphanumeric screen will inform the operator of the system position in the command stream. He will be aware of exactly what task the system is processing at any moment. Error messages are synchronized with the command input which causes the error. When an error is detected, any input that is pending (and has not been executed) is discarded. Command input echoing is therefore valuable to operators of all levels of experience.

Data Creation/Collection

Traditionally, already completed IC layouts could be entered into the system through one of CALMA's several digitizing terminals. Now, data may also be entered from either the tablet or the digitizer in design mode. Certain commonly used structures stored in the database can be called up and used in a new diagram. References to structures can be placed at any orientation and magnification.

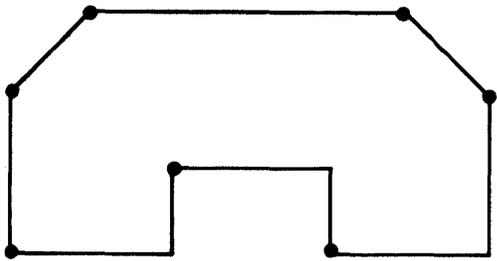
Several hardware and software features are available to facilitate rapid data input. These include lights for accurate positioning half way between grid lines as well as grid display on the graphics CRT.

User Units

IC designers work in various units of measurement ranging from angstroms to inches. GDS-II coordinate value displays can be set up to read out in pre-defined, user-specified units.

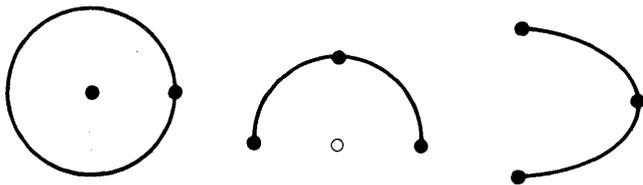
A user building hybrids, for example, could work in mils, with a resolution of 1/1000 of a mil. The displays would read in mils to 3 decimal places. The actual work area is over 4 billion data units. The precision is defined by the number of data units per user unit.

- Octagonal - Extends along the horizontal or vertical line segment until it intersects with a 45° line from the digitized point. The vertex becomes a coordinate.



- Hardlocks - Supports orthogonal digitizing by hardware locks on the restrained cursor digitizing table.

With GDS-II the user can create circles and arcs, neither of which requires more than three coordinates.



Editing

Once an item has been entered, the ability to delete or modify this data becomes important. Item editing is a two-step process. First, the elements that require editing are selected (i.e., retrieved by the GET command). Then, the appropriate item transformation is specified.

Selection

Selection can be as easy as the specification of one or more paths, boundaries, structures, arrays, or text. If this data is in the form of a single item, the designer simply has to specify the number of the item that needs to be edited, and the selection process is complete. In some cases, however, the operator may choose to manipulate a group of data which cannot be selected with a single request. For example, suppose a user wanted to edit only the intersecting portions of three distinct elements in the database. For cases like this, GDS-II provides the capability to identify a "group" of elements interactively. Data can

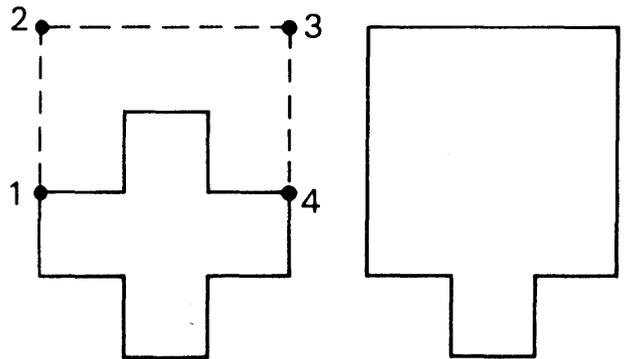
be selected implicitly, using a window or other criteria, or explicitly, by pointing out individual data. Data may be "unidentified" in the same way. For example, it is possible to select all closed polygons on a particular layer within a certain window, and then exclude specific polygons from the identified group.

All commands which accept a window to implicitly select a group of points are also sensitive to other selection criteria. In GDS-II data can be qualified for further editing according to any defined property in addition to its geometric position. For example, elements can be made selectable by layer, datatype, textype, or kind.

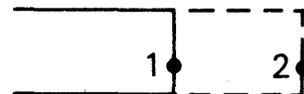
Transformation

Once an item or group of items has been selected for editing, transformation of that data can take place. The following are certain transformation options available to the designer:

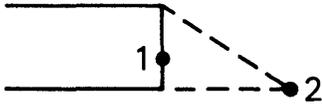
- *Modify Polygon*, the most versatile edit command, allows the user to open a polygon beginning at any location; then, to close the polygon in the same or a new way, simply by entering the desired change. The system will automatically merge to change with the original data.



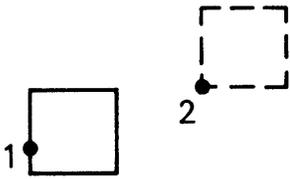
- The *Stretch* command stretches a user-selected side of a polygon to a new location.



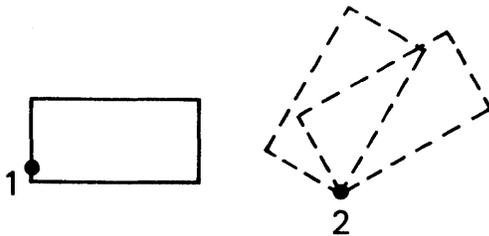
- With *Move Point*, a user-selected point is moved to a new location.



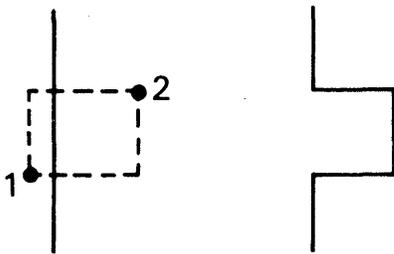
- *Move* moves the item or identified group to a new location specified by the user. It maintains positional and geometrical relationships.



- With *Rotate* the user can rotate the selected shape or structure to any degree specified.

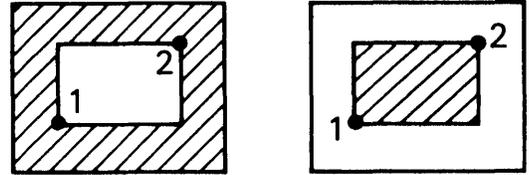


- *Notch* adds a notch of the size and direction defined by the user at any location on the line segment(s).

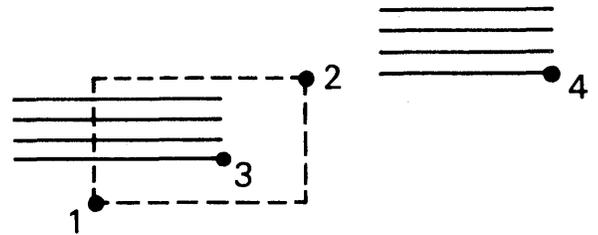


- *Edit Area* allows the user to edit the defined area, by removing all data within the area or outside the area. Any boundaries intersected by the defined area can be

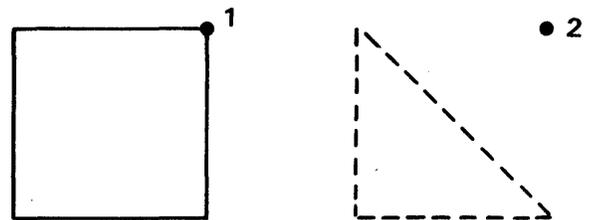
automatically closed. The "area" can be a line as well as rectangle. This allows data to be conveniently split apart and shifted.



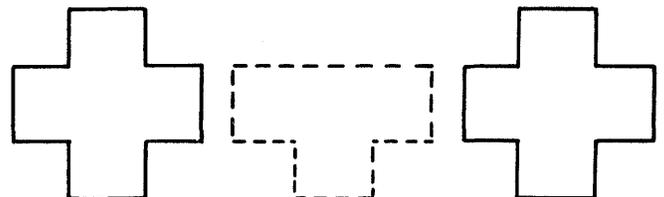
- With *Window Move*, the user can window defined elements using two points. With a third point he specifies the new location for all data within the window. A fourth point can be added to pinpoint the exact location in relation to other previously entered data.



- *Delete Vertex* allows the user to delete any single coordinate pair of the current item by identifying a point either on or near that vertex.



- *Un-Get* allows the user to return an item to its previous state before coordinate modification. This means the designer may go back if a mistake occurs.



Library Status

- Library Status - Displays information regarding the currently open library set.

```

STATION: 1      X: +502.      Y: +1826.      TIME: 05:57:48
COMMAND INPUT      Item #0
?
? OPENREF1
Reference Library I Name: ICS
? LSTAT
Library      Filename      Used Pages  Free Pages  Used %
WORKING      2148.DB      262
REF 1        GDSII:ICS.DB  109

Q GRD LVL PRI PROGRAM      TOT USE  NO  IN  OUT  DIR  SWAP
      MEN  MEN  LFT  COINS  COINS  DCB  DCB  TYPE
WA BG  2  2  GDSII:IN  75  75  50  #T11  #T10  GDSII
W  B62  1  2  JOURNAL  3  3  17  #T11  #T10  BACKGROUND
W  B63  1  2  BINACTION  24  0  17  #T11  #T10  0
W  B64  1  4  CL1      24  9  17  UK.J1  UK.JL  GDSII  15
W  B65  1  2  BINACTION  24  0  17  #T11  #T10  0
W  B66  1  2  BINACTION  24  0  17  UTIM  UTOU  000  0
?
  
```

- List Libraries - Lists the structures within the currently open library set.

Errors

- Errors are displayed in bright characters accompanied by an audible beep. An error message describes the type of error.

Help Files

- Help files allow the user to request information about a particular command. These files will display on the alphanumeric screen the same documentation that is in the hardcopy Users' Manual. Help files may be called at the touch of a command button.

For a quick refresher to the experienced operator, each Help file begins with an example of the command. Next come prerequisites to execution, followed by a detailed description of operation, the classification of the command, the restrictions for use, and examples of the command in use. These messages can be terminated at any point after the desired information is displayed.

Every command has a Help file, so user assistance is always available. The user can edit the Help files or create new files using the text editor. This allows users to document their own application notes and procedures.

```

STATION: 1      X: +12.      Y: +22.      TIME: 06:29:38
APPROVAL      2148.101      Item #0
? HELP
Command name: INITLIB
COMMAND: INITLIB      02/08/77
INITLIB(CR)
Enter Library Name : filename(CR)
Maximum file size (0 if read): n (CR)
Data Base Units per User Unit : n(CR)
User Unit Size (H, CH, MH, MCH, A, DCH, MDL) : arg(CR)
or
INITLIB "filename";fileize;MU;arg(CR)
Prerequisites:
None Required
Function:
INITLIB creates a new library or initialized an existing
GDS II library file.
  
```

Complex Program Development

The GDS-II command language with its consistent set of syntax rules, conceptually simple primitives, and accompanying command menu make system operation easy to master. During the "get-acquainted" period with the GDS-II System, and perhaps for some time after, the user's experience with the command language will be limited to defining new menu buttons to meet special requirements. This ease of operation, however, does not mean that the GDS-II System is limited to unsophisticated data manipulation. In fact, the system supports a wide range of capabilities in the development of user programs. Development of custom programs allows the user to tailor the system to fulfill unique data collection and data processing requirements.

GPL-II

The highest-level programming language provided by the GDS-II system is the GPL-II Programming Language. GPL II is a general purpose language designed for interactive use. It is patterned after APL, a widely available commercial language. Both GPL and APL allow for definition of a complete set of array operators. With the GPL-II System, complex tasks (such as translation or rotation of a matrix

representing a polygon) can be written as a single operation. Additional benefits include interactive execution and a full set of debugging aids.

Despite the high-powered capabilities of GPL-II, this language is easy to learn and use. In fact, all system commands are actually GPL-II commands; therefore, any set of commands can be transformed into a program with no changes. Additional GPL-II commands are available for more complex programs, but for the beginner, a working knowledge of the command menu is sufficient for development of simple user programs.

Typically, the computational power of GPL-II is used to provide conditional execution of commands based on feedback from the system. On the other hand, the computational power of GPL-II can be used, for example, to evaluate arithmetic expressions in the command language without writing a program at all. A more significant use of the GPL-II language is to implement programs which access values stored in user defined extensions to the database.

GPL-II includes a large number of primitives designed for interactive use. The primitives are defined at a high level for efficient operator input and for efficient execution. One reason for writing GPL-II programs is to extend the number of primitives available. While GPL-II programs can generally be used in the same way that the built-in primitives are used, the programs can be tailored for more specific data processing requirements.

In addition to receiving data passed as parameters, more advanced GPL-II programs and functions have direct access to global variables which define the terminal context. Most operating systems peripherals are fully supported for use by GPL-II programs.

GPL-II data manipulation capabilities are equivalent to those provided in APL/360 developed by IBM. Data may be organized as scalars, vectors, matrices, or higher dimensional arrays. Data may be stored in real, integer, logical, or character form. All storage allocation is performed dynamically. Automatic mode conversions are also supplied as necessary.

Other Programming Languages

The GDS-II System supports several standard Data General programming languages. The most important of these is DG/L which is a derivative of

ALGOL. Users who have the DG/L compiler can write their own application programs. These programs can be run as background jobs.

The advantage in using DG/L is that the source language is compiled directly into the ECLIPSE instruction set rather than being interpreted, as is the case for the GPL-II programming language. Therefore, DG/L is a more suitable vehicle for programs with very extensive computational requirements.

The GDS-II System also supports FORTRAN as defined by Data General as well as the macro assembly language available from Data General.

Except for the GPL-II Programming Language, all programming language development and debugging must be performed as a separate job. It is possible to configure the system so that one or more stations or other terminals are used as non-graphic Command Line Interpreter grounds. In this mode, all of the standard RDOS compilers, loaders, and utilities are available. Even debugging is possible except in those cases when the interrupt off debugger is required to diagnose a problem

System Operation Specifications

- The GDS II command language is in plain English with consistent syntax.
- Command categories include:
 - Data creation
 - Data modification
 - Numeric selection
 - Data display and output
 - View control
- Menus can contain up to 512 buttons and four levels. Any number of buttons can be displayed on-screen.
- All commands are echoed on the alphanumeric screen.
- Items can be:
 - Boundaries
 - Paths
 - Text References
 - Structure References
 - Array References

- Input modes:

- Straight
 - Orthint
 - Edge
 - Octagonal
 - Hardlocks

- Edit functions include:

- Modify polygon
 - Stretch
 - Move Point
 - Move
 - Rotate
 - Notch
 - Edit Area
 - Window Move
 - Delete Vertex
 - Un-Get
 - Intersection Check Mode
 - Un-Do

- The GDS-II Text Editor (TED) enables the user to:

- Edit and create text files
 - Edit GDS-II menus
 - Edit or create GPL-II source programs

- The Alphanumeric prompts include:

- System Status
 - Item Status
 - Operating Modes
 - Library Status
 - List Libraries
 - Errors
 - Help Files

- Program languages:

- GDS-II
 - GPL-II
 - DG/L
 - ALGOL
 - FORTRAN
 - BASIC
 - Assembly
 - Macro Assembler

NEW PRODUCTS

The Multicolor Raster Display (MRD)

CALMA's newest product is a 20-inch refresh graphics display with complete, seven-color capability. This display, called the Multicolor Raster Display (MRD), promises to revolutionize on-line design techniques by enabling the user to exhibit all standard GDS-II graphics in a variety of sharp, clear colors. On-screen menus may also be shown in any color or pattern.

Specific designing enhancements made possible with the MRD include:

Layer Recognition: GDS-II allows definition of up to 64 different layers. Different color combinations will eliminate the difficulty of distinguishing among individual layers.

Identification: During selection, any shape, structure, or area switches instantly from color to white. This provides immediate feedback and verification of items selected.

Area Shading: Along with solid color capabilities, MRD users may completely fill any outlined shape or area with any one of four different shading densities. These shading densities enable easy recognition of areas, as well as exact definition of complex shapes.

Selective Erase: With the MRD, an entire shape or area can be deleted without affecting any other portion of the screen. To move a complex structure, for example, the structure is first identified. Once identified, the structure turns white to verify that the data selected is exactly what the user wants to modify. Then, the user points to the new location where the structure is to be moved. The data disappears and then reappears in full color at the new location.

Ease of Checking: Since colored areas can be shaded, then superimposed over other colored areas, these overlapped areas take on a new color, which makes them stand out more clearly.

The MRD operates in a multi-terminal environment and can be used in combination with conventional storage screens. Because the MRD is a refresh display, special features such as selective erase and

movement may be accomplished without completely repainting the screen. Because selective erase requires much less screen repainting (up to 20 times less), the main graphics CPU is freed from the task of viewing and is available for other jobs.

The MRD is driven from its own graphics processor which controls all display functions including shading. The screen features microprocessor control and state-of-the-art technology.

MRD Specifications

- Fully programmable processor (51 instructions) with 150-nanosecond cycle time
- Fast access MOS RAM memories with read/write, word or bit capabilities
- Automatic DMA access for block transfers (833,000 16-bit words/second)
- Bit map type refresh--expandable from 1 to 16 bits per picture element
- Full complement of user-oriented options
- 512 x 480 resolution
- 40-cycle refresh
- 20-inch, Polarized screen
- 7-color capability

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Power Requirements	103.5 to 126.5 VAC, 47 to 63 Hz, single phase
--------------------	--

Matrix Plotter Package

Another new CALMA GDS-II product is a high-performance software package that drives the Versatec 8272, 8242, 8236, and 8224 matrix plotters as well as the Printronix printer/plotter. By providing, at minimum, a 100% increase in plotter speed, this



VERSATEC 8242 PLOTTER

package will drastically reduce IC plot backlogs. For example, with CALMA's new software package, the Versatec 8242 can produce a shaded IC plot in less than half the time it takes the fastest pen plotters to produce a multicolor plot. These benefits in output speed increase as circuit complexity increases.

CALMA's Matrix Plotter Package is ideal for any IC plotting job. Special applications include:

- 1) Initial and final verification of Plot-vs-Composite data checks.
- 2) PG blow-up verification
- 3) E-beam raster pattern verification.

Exclusive high speed software sorts and formats the data for output. Data can be represented by up to 16 different line codes and 48 different fill codes.

This software package runs under the GDS-II Background Job System. For the Printronix plotter, the raster data is spooled from the disk. For the Versatec plotters, the raster data can be output to magnetic tape. This tape can then be plotted on-line by the CALMA system, or used to drive any Versatec off-line controller. For smaller plots, data is directly spooled from the disk to the Versatec.

Since the Versatec can use magnetic tape for transfer, plot size is determined by tape density. 800 bpi density will provide a plot of approximately 41 x 100 inches, regardless of data complexity. 1600 bpi density will double plot length.

In addition, for optimum performance, 32K of additional core allocation is recommended.

Versatec Hardware

The Versatec 8272, 8242, 8236, and 8224 high-speed electrostatic plotters provide up to 72-inch-wide plots for a variety of IC applications. Special features include:

- 200 spot-per-inch resolution
- Plots up to 72 inches wide
- Proven design providing reliable operation
- High resolution for detailed output

- High data density handling
- Plots with an accuracy of 0.2% with exclusive shaft encoder and servo-drive mechanism
- Exceptional repeatability

The large plot area allows for complete, single-sheet plots while retaining full, visible definition of details. With operating speed 10 to 30 times faster than pen plotters, full-scale, E-sized or larger drawings can be plotted quickly, accurately, and reliably.

Because electrostatic plotters output one complete raster scan of data at a time, high density and plot complexity do not reduce speed. Versatec models can plot up to 2.0 inches per second. A multiple input buffering system insures maximum printer/plotter speed without loss of data.

An exclusive shaft encoder and servo motor drive assure required plot accuracy. The shaft encoder determines timing and writing location, while the servo motor responds for precise control of paper movement. Together they provide an overall accuracy of 0.2%, or 15 mils, at maximum speed. Repeatability is absolute since all data is plotted at the same time for each scan line.

Horizontal accuracy is provided by a precisely engineered stationary writing head.

Specifications

Writing method	Electrostatic
Paper drive	Servo motor
Timing control	Shaft encoder
Controls:	
Switch panel	ON-OFF Paper advance Low paper indicator Low toner indicator Contrast adjustment Reset On-line/off-line switch
Frame mounted controls	
Concentrate adjustment	Automatic
Toner supply	6-gallon system, supplied in 2-gallon containers.

Concentrate supply	16-oz. bottle		
Operating environment:			
Temperature	60° to 80° F (Fahrenheit)		
Relative humidity, non-condensing	10% to 70% relative		
Paper width (inches)	22, 24	36, 42	72
Plotter Dimensions (inches)			
Height	40	40	40
Width	45	63	100
Depth			
Standard	34	34	34
with take-up	46	46	46
with extension table and take-up	70	82	100
Weight (net, pounds)	750	900	1250
Power (watts)			
Maximum	2000	2200	3300
Operating	1400	1600	1900
Voltage, all units	100, 115, 200, or 230 VAC±10%, 48-62Hz. Single phase, Specify at time of order.		

The Printronix Printer/Plotter

The Printronix printer/plotter can function in any or all of these capacities:

- Line printer
- Plotter
- Hardcopy unit

Line Printer:

The basic unit is a 300-LPM line printer capable of 132 characters per line. This raster matrix line printer uses a 9x7 dot matrix.

The print font is determined electronically, rather than mechanically. By using leaf-spring hammers and having 50 percent fewer mechanical components, its simple mechanical

design ensures uniform impact dot for dot. Thus all characters have uniform intensity, unlike the characters on mechanical font printers, some of which are much lighter than others (for example, w's are lighter than periods).

Many features that are optional on other printers are standard on COPI. Some of these are electronic VFU, plot mode, quick interchangeable fonts using PROMS, minimal mechanical maintenance, high MTBF and low MTTR.

Plot Mode:

In plot mode, COPI plots at 16-2/3 ipm, with 60 dots-per-inch horizontal and 70-dots-per-inch vertical. Plot widths can be up to 12.8 inches. Software selection allows graphics and text to be printed on the same page. Impact printing techniques with incremental paper feed allow plotting without spooling. Also, the printer can be interrupted without creating a toner bar, as happens with electrostatic printers.

Hardcopy Mode:

In hardcopy mode, COPI can support two CRTs. No chemicals, special paper or special handling are required. COPI converts from CRT data to line printer data and can still be used for printing and plotting.

Advantages

- Lower service requirements
- Variable font capability
- Multifunction capability for approximately the same cost as a line printer
- Higher reliability

Applications

- In-house reports
- Quick-look plots

BACKGROUND FUNCTIONS

Introduction

GDS-II offers the largest array of background functions in the interactive graphics industry. This extensive collection includes optical pattern generator and e-beam formatting, design rule checks, data manipulation programs, on-line and off-line plotter support, tape library, and output format handlers. Since the background processor never intrudes into the foreground, all these programs operate on a minimal-interference basis.

Background Job System

All background jobs are controlled through the GDS-II Background Job System (BJS). Based on two unique CALMA software products, Multiground RDOS and the Job Management System, BJS provides both a high degree of background job independence and a repertoire of job management facilities.

User management of the various background operations available on GDS-II is possible through a special interface between RDOS and the GDS-II command language, called the Job Monitor. A variety of background job control functions is available, which allow the user to: enter or add a job, change the priority of a job, suspend or abort a job, pause a plotter, and obtain information about jobs which are queued, active or terminated.

Specifically these functions include:

- **JOB CREATE:** This function allows the user to define a background job and enter it into the job system. The Job Create command collects and validates the parameters, prompting the operator for the required information. This interactive mode of job definition is used for plots, pattern generation, rules checks, and other jobs whose parameters are fairly complicated. Job definitions can also be saved for execution at a later time.
- **JOB INFO:** This function displays the status and related information for all jobs known to the system. The following information is displayed for each job:

Job Name (both user and system name)
Station that initiated the job
Status: waiting, active or complete; whether the job is suspended; whether it is an on-line plotter job; etc;
Priority Class
Time of Job Entry, Initiation and Termination
Elapsed CPU Time for Job

- **JOB LOG:** This function displays the status and related information for a specific job. If the job is still active, this output can be displayed in real time.
- **JOB PRIORITY:** This function allows the operator to assign a different priority class to a job already entered.
- **JOB SUSPEND:** This function suspends the execution of an active job, or prevents a waiting job from becoming active.
- **JOB START:** This function also reverses the effect of the Suspend Job command and allows the job to be executed.
- **JOB ABORT:** This function causes a job to be immediately terminated. A queued job is removed from the queue. All spooled output generated by the job is purged.

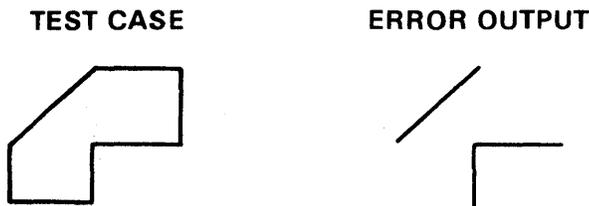
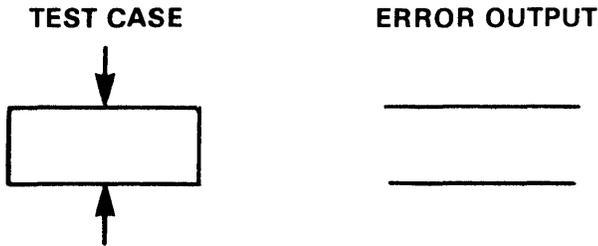
Background Job Efficiency

Background jobs effectively use CPU time not needed for foreground activities and compete with each other on a priority basis. For example, the operation of dumping a graphic library to tape is scheduled as a high-priority background job. Because the job is independent of the station that initiated it, that station is available for interactive work while the dump proceeds. At the same time, the dump, because of its high priority, can run at once without being delayed by a lengthy rules check or other lower priority jobs.

Design Rule Checks (DRC)

GDS-II supports a comprehensive set of data checking and validation programs that operate in the background. Specifically these DRCs include:

Interior Spacing Check: This all-angle check tests closed polygons for violations of specified minimum internal spacing. All closed polygons on a specified layer within a specified window are tested. Errors are noted by an output of the two lines creating the violation. These lines may be parallel, convergent or indicating a corner to corner spacing.



Exterior Spacing Check: This check notes all spacing violations between the edges of any two distinct polygons. Whenever the distance between two polygons is less than the specified minimum, these violations are output. This check may be run within a single layer or between two layers.



Efficient search procedures make these tests feasible for even the largest integrated circuits. Because these checks execute rapidly, the user can run them at the time of data creation (i.e., storage), and quickly modify any errors as they occur. Errors can be either viewed on the screen or plotted (alone or in combination with original data) for easy inspection.

(More extensive DRCs are discussed in the Future Products section of this manual.)

Database External Format

The OUTFORM and INFORM programs create and read back magnetic tapes (or other files) in a standard external format. Because of this standard output format, any modifications to the main database manager will be invisible to the user. This insures that all input/output archival tapes remain totally compatible.

User Program Access

Users who have the DG/L compiler can do database access from their DG/L programs by invoking standard database access routines that are provided with the DG/L library.

Pen Plotters

GDS-II plots either on-line in background mode or by generating control tapes that drive off-line plotters. In both cases, the operations proceed without interfering with digitizing, designing, editing, or display functions. The following functions are provided:

- **Layer Select:** At job creation time, the operator can select which layers are to appear on a particular plot. Normally, each layer is plotted in a different color. Multiple-pen plotters can be made to change pens automatically for each layer and pause when all pens have been used, or continue using the same pens.
- **Scale:** Plot scales are also selected at job creation time and require no complex factor computation. The user simply inputs his desired scale and the system does all the computation.
- **Plot Area:** A plot can be of an entire circuit or a user-selected portion of the circuit. The area to be plotted is selected by defining a rectangular window. Multiple windows can be queued for sequential plotting.
- **Filling:** This option, available on raster scan plotters, permits automatic filling of polygons and offers 48 fill codes, as well as 16 different line codes.

Recent developments in GDS-II software have made it possible for the system to drive the Versatec 8272, 8242, 8236, and 8224 matrix plotters. A combination software, plotter and controller package is now available. (See New Products.)

Pattern Generation (PG)

GDS-II pattern generator output programs make it possible to move quickly from a finished design to actual circuit production. PG tapes contain data from designated layers of a specific library structure. Closed polygons, which may contain edges of arbitrary orientation, are efficiently fractured into rectangles. The entire set of rectangles for each layer is sorted and optimized for the particular pattern generator.

The following optical pattern generators are supported: Mann 1600, 2600, 3000 and 3600, Electromask 2000. All pattern generators are supported to their full resolution over the full range of stage travel.

The time to generate a tape (in the absence of other system activity) will not exceed the limits given below, for typical integrated circuit layers, whether of regular or random layout.

Number of Flashes	Time to Produce Tape
25,000	30 min.
50,000	1 hr.
250,000	5 hrs.
500,000	10 hrs.

For all these cases, the sort optimization will achieve an average pattern generator flash rate of at least 2 flashes per second for the Mann 3000, and at least 6 flashes per second for the Electromask 2000.

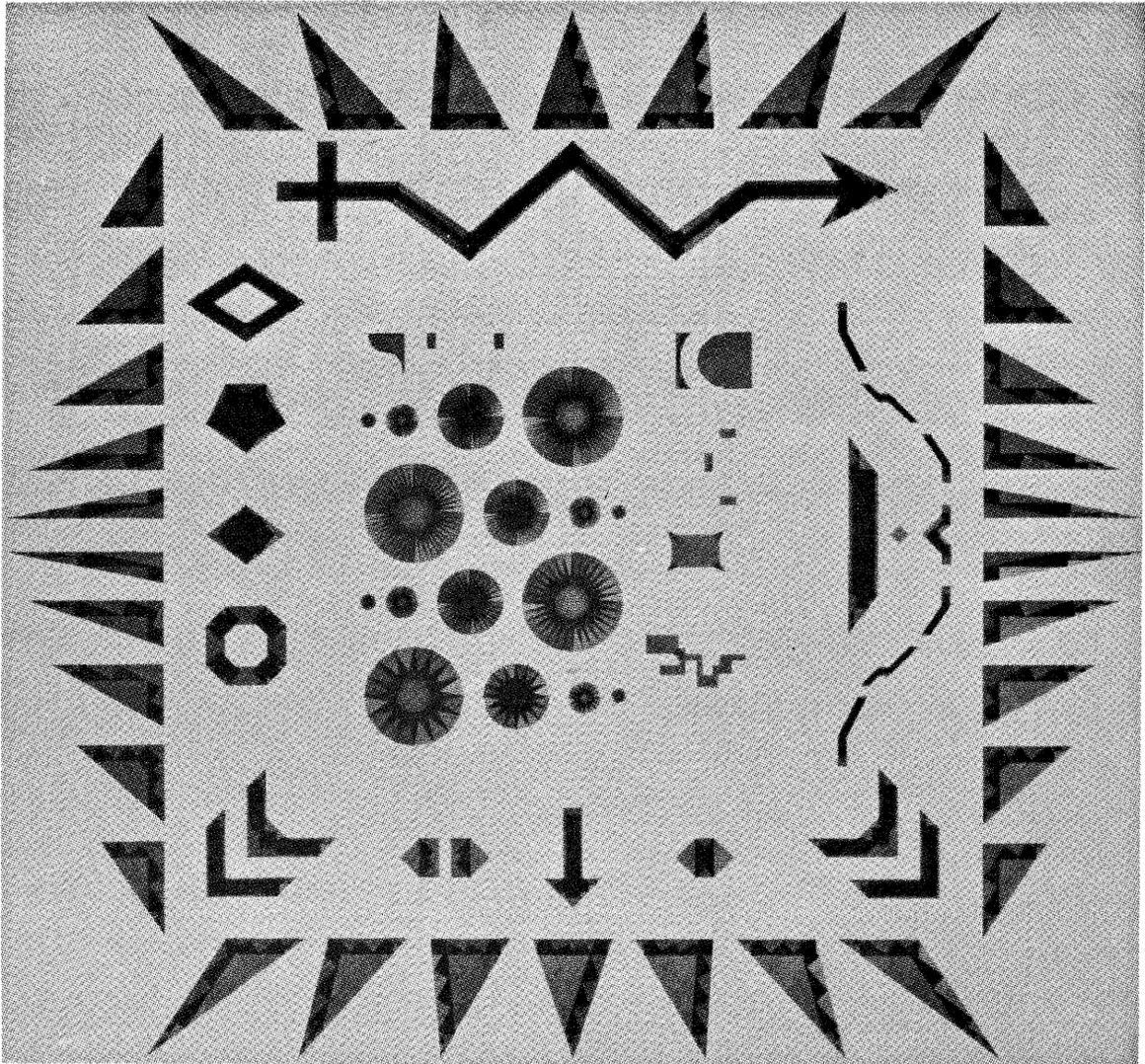
The pattern generator read-in program reads any CALMA-generated pattern generator tape into the data base for verification purposes. The pattern generator read-in models supported are the same as those supported for output.

All-angle software to drive electron-beam pattern generators (including the Etec MEBES) is also available.

The main requirement for e-beam PG is full healing capability. Since the raster pattern of the e-beam usually cannot scan an entire circuit, the chip must be divided into sections. GDS-II software automatically windows these sections and closes all shapes that intersect the window at any angle.

CALMA e-beam software provides the precision necessary for supporting any machine to the point that the actual machine resolution will always be the limiting factor.

In all cases, GDS-II PG and e-beam software is completely all-angle and offers no restrictions in the design of the database. Output machine limitations are calculated and incorporated at the time of tape creation.



These complex shapes, shown in this enlargement of an all angle test pattern plotted on an Electromask Pattern Generator, demonstrate the highly sophisticated nature of CALMA's pattern generator

software. The reticle was underdeveloped to illustrate how each shape has been fractured into a set of rectangles acceptable to the pattern generator.

Background Specifications

- All background operations are controlled by the Background Job System (BJS).
- BJS is compatible with standard RDOS and supports the same languages that RDOS supports.
- Functions available for user control of background operations include:

- Job Create
- Job Save
- Job Info
- Job Log
- Job Start
- Job Suspend
- Job Priority
- Job Abort

- Designated background job priorities insure efficient use of system time.
- Specific background operations include:

- Design Rule Checks
- Database External Formatting

- Plotter Support
- Pattern Generation
- GDS-I to GDS-II
- GDS-II to GDS-I
- User Programs

- Design Rule Checks include:

- Internal Spacing Checks
- External Spacing Checks

- Pen Plotters supported by GDS-II software include:

- Calcomp 748, 936, 960, 1039, 1055
- Xynetics 1050, 1100, 1200, 2200
- Versatec 8224, 8236, 8242, 8272

- Pattern generation software is available for:

- David Mann 1600, 2600, 3000, 3600 Pattern Generators
- Electromask 2000 Pattern Generator
- Cambridge Instruments EBMF Electron-Beam Plotter
- ETEC MEBES Electron-Beam Plotter
- Extrion Electron-Beam Plotter

MULTIGROUND RDOS

Introduction

In most current computer installations, real-time and program development functions are performed sequentially. In other words, a new function is not begun until the current function is completed. Because the collection of data and its analysis require a fraction of the total available system resources, many parts of the system are idle for significant periods of time. For example, monitoring four terminals on a system requires only a portion of the memory and input/output devices in the system. It also requires only intermittent use of the central processor.

For most applications, sequential processing is adequate. However, IC design involves manipulation of large amounts of data as well as high speed background throughput, two tasks that require special system modifications.

To meet these needs, CALMA has developed its own Multiground Real Time Disk Operating System, called Multiground RDOS, which can handle up to 15 simultaneous background modes while running up to 6 stations in the foreground. By permitting multitasking, CALMA's Multiground RDOS insures that as much of the total system is kept performing productive work as much of the time as possible.

RDOS Organization

GDS-II's Multiground RDOS is actually an enhanced version of Data General's Real Time Disk Operating System. This more powerful system is known as Multiground RDOS because it supports additional jobs beyond the "foreground" and "background" offered by standard Data General RDOS.

CALMA's Multiground RDOS insures complete isolation for each of the up-to-16 grounds being run. This permits, for example, several different interactive graphic systems to operate at the same time, (i.e., GDS-II and DDM), or an interactive graphic system to operate concurrently with program development and debugging.

User specified priorities determine the CPU processing time allowed each task. The higher priority job is run in the foreground while the remaining jobs are run in the background.

RDOS includes a program overlay system which gives users great flexibility in overlaying programs from disk into main memory. Large user programs can be segmented easily into disk-resident overlays to allow efficient, orderly use of main memory. Tasks stored on the disk occupy main memory only when they are ready for execution.

RDOS Flexibility

A modern real time operating system must be geared to change and diversity. CALMA's Multiground RDOS system can exist in an almost unlimited variety of machine configurations; different installations will typically have different configurations as well as different applications. Moreover, the configuration and scope of the project at a given installation may frequently change. Thus the operating system must be adaptable to a number of different environments. All of this puts a premium on the system modularity and flexibility which have been designed into RDOS.

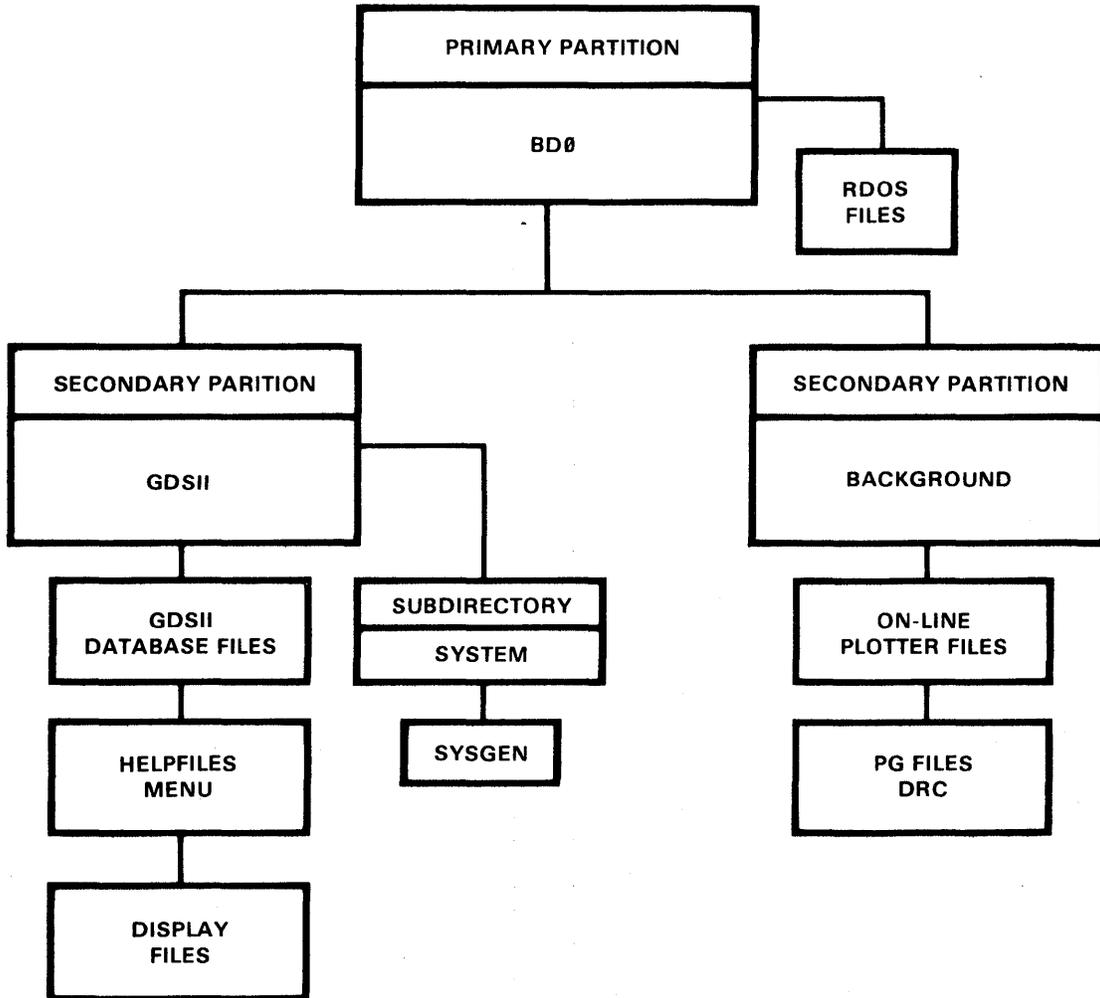
Foreground/Background Programming

To increase system utilization, RDOS permits several programs to be core-resident and run with apparent simultaneity. That is, the operating system switches control between task schedulers according to a predetermined priority assigned to each by the user. Foreground/background programming permits unrelated collections of tasks to be performed, sharing basic system resources.

Data from a background job is processed while waiting for an event or for data from a foreground job. The background mode can also be used to develop new programs without interrupting on-going jobs. Foreground and background programs can be hardware-protected from each other and from the operating system, so a debugged program can run in the foreground, while a lower-priority program can be constructed and debugged in the background.

Although the foreground and background programs are truly independent, the operating system does permit them to act in concert if desired. The foreground and background programs can issue messages to one another via certain system calls.

GDSII RDOS DISK AND RDOS BACKGROUNDS



Memory Management and Protection

Foreground/background programming with software memory partitions is unquestionably a more efficient use of computer resources than is single-program execution. Nonetheless, even this improved scheme has drawbacks which limit its usefulness. The first drawback is that there is no guarantee that if one program experiences failure (e.g., by accessing cells outside its allocated address space) it will not cause the other program to fail also. Secondly, foreground/background programming requires that great care be exercised to insure that each program is tailored to fit into available memory.

Multiground RDOS overcomes these limitations by supporting a Memory Management and Protection device. This device, specially developed for Eclipse computers, provides a hardware separation of the background, foreground and operating system areas.

Memory Management and Protection extends the maximum memory configuration for a central processor from 32K to 256K words.

Within the framework of an executing program, two modes exist. In the first mode, absolute mode, only the lower 32K is directly addressable and the mapping device is not used. RDOS resides in these low physical memory locations and executes in absolute mode.

The second mode is called mapped or user mode. In user mode, up to thirty-two 1024 word blocks of memory are mapped by the management unit to produce an apparent (logical) 32K continuous address space. Both the foreground and the background programs execute in mapped mode and are essentially unaware of their actual memory locations.

Command Line Interpreter

The principal means for a user to communicate with RDOS is via the Command Line Interpreter (CLI). The CLI is a system program, executable in either the foreground or background, that accepts command lines from the console and translates this input as commands to the operating system. The CLI acts as a console interface between the user and the system; it permits user programs to be executed and utilities to be run, it performs certain file housekeeping chores, and it provides a diversity of other functions for the user acting at the console keyboard.

RDOS restores the CLI to main memory whenever the system is ready to execute a program--after initialization, after a disk bootstrap, after a console break, after the execution of a save file, etc. The CLI, in turn, indicates to the user that the system is ready to accept commands by typing a ready message on the console.

RDOS Features

Task Scheduling:

A task is a logically complete path through a program, which can be executed independently of another task within the same program.

RDOS permits "multitasking" within each ground; this means that within the same address space there may be more than one asynchronous process (task) that competes for CPU time. The Scheduler for Multiground RDOS is interleaved among grounds end tasks. In other words, a priority is computed for each task in each ground. This permits high- and low-priority tasks to co-exist in the same ground. The fast response of Multiground RDOS is attributed to this advanced scheduler, which does not allow CPU bound background operations to degrade the basic interactive response of foreground graphics.

The Multiground RDOS scheduler insures that all tasks with the same priority progress at the same rate. The scheduler keeps statistics on which tasks are using the most CPU time and temporarily lowers their priorities until there has been an equal allocation of time.

Program Swaps and Chains:

Any program running under RDOS can suspend its own execution and either invoke another distinct program or call for a new section of itself. The current program is overwritten in both cases.

The *program chain* facility permits programs to be run which require more core storage than is ever available at one time. To use chaining, a program must be written in serially-executable segments, each of which calls the next segment.

The *program swap* facility, by contrast, permits distinct programs to call one another in much the same manner as subroutines are called. The primary difference is the size of the routine and the manner of argument passing. Whereas subroutines are always

smaller than the total amount of resident core, program swaps are as large as total user program space (or as large as multiples of user address space if chaining is performed). The program whose core images are overwritten during a program swap operation is stored temporarily on disk. Information saved enables the program to be restarted upon return from the swap.

System Input/Output Buffering:

All data transfers to or from disk files and hardware devices are buffered in the operating system before the data is transferred into the user's buffer. Each system device handler has a small buffer associated with it, depending on the speed of the device. The other area is the system buffer area which is organized into blocks of 256 words each. When data is transferred from a disk file, it is first read into this buffer area before accessing the data within the block. This allows smaller transfers of data to occur to or from the user area, and frees the user from having to provide sector buffers in the program.

Spooling:

When messages are output on a slow device like a teletypewriter, and its buffer fills up, the calling task is suspended until the buffer is emptied. Spooling allows messages to be temporarily stored on disk. The messages are later returned to main memory when space in the buffer is available. The significance of spooling is that output messages or information can be queued without putting excessive loads on the user main memory for buffering. This also frees the user from having to optimize his message requests, and thus permits more effective use of the device.

RDOS also controls the plotter spooling, which is specially adapted to plotter needs. The spooler incorporates not only plotter data but also operator communications necessary to allow changing of pens and paper. This enables plotter spooling to be truly independent of job execution. Available operations include pausing the output from a spool and purging spooled data on a job-selective basis.

File System:

The RDOS file system provides a generalized means of accessing files on disk. Filenames in RDOS may be from one to ten (10) characters in length. Optional filename extensions of one or two letters may be added. Filenames are referenced through directories,

which may be nested up to three deep in RDOS. The top level, called the Primary Partition, is equivalent to the entire disk pack. Files may exist at the primary partition level or may be associated with nested directories.

The first level below the primary partition is called the secondary partition. It is characterized by a fixed amount of disk space. The secondary partition may either have files directly accessed from it or contain one additional level of nested directories. This additional level, called a subdirectory, may have files but no additional nested directories.

RDOS offers 3 disk file structures Sequential, Random, and Contiguous, which enable fast and flexible access to bulk storage.

Information in sequentially organized files is stored in groups of disk blocks. The last word of each 256 word block is used to store a link to the next block in the file. This link is invisible to the user, and is solely for system use.

Random file organization provides the best combination of flexibility and accessibility of data. In randomly organized files, a master index of all physical block addresses is created. The master index blocks, themselves, are sequential files.

Blocks of data storage in random files utilize all 256 words for information storage. Each block is assigned a sequential, positive integer by its position within the master index, indicating the block's logical position within the file. In processing randomly organized files, two disk accesses at most are required for the reading or writing of each block: one to access the file index and one for the block of data itself. If the index is main-memory resident (having previously been read into a system buffer), only one access is necessary.

Contiguous file organization has a rigid structure, but it provides the quickest access to data. Contiguous files are composed of a fixed number of disk blocks, which constitute an unbroken series of disk block addresses. These files can neither be expanded nor reduced in size, since, by definition, they occupy a fixed series of disk blocks. Contiguous files may be considered as files whose blocks may be accessed randomly, but without the need for a random file index.

All I/O operations that can be performed on randomly organized files can be performed on contiguously organized files. Contiguous files have the advantage of usually requiring less time for accessing blocks within the file.

Programming Facilities

Multiground RDOS retains complete compatibility with Data General's RDOS, which allows all the software developed by Data General to be run without modification. Compilers for FORTRAN, ALGOL, BASIC, Assembly, and DG/L are available, along with Text Editors and Macro Assemblers. Data General's complete selection of support programs, such as editors, loaders, and debuggers, are also compatible with Multiground RDOS.

Multiground RDOS includes custom I/O drivers for all of CALMA's hardware.

Specifications

- Capability to handle up to 15 simultaneous background modes while running up to 6 stations in the foreground.
- Complete isolation of grounds
- Concurrent production and program development
- User specified job priorities

- Interaction with Command Line Interpreter (CLI)
- Support for Memory Management and Protection
- Independent task scheduling
- Program Swaps and Chains
- Input/Output Buffering
- Spooling
- Ability to access to all hardware peripherals and disk data files
- Three-deep nesting of filenames
- Random and Contiguous file structures
- Programming Languages:

DG/L
FORTRAN
ALGOL
BASIC
Assembly
Macro Assembler
GDS-II
GPL-II

HARDWARE

The CPU Base System

The minimum GDS-II system has the Eclipse S/230 Central Processing Unit (CPU) with 300 megabyte disk, 75 inch-per-second dual density tape drive, keyboard printer, digitize or edit station, and on-line plotter. The maximum system can support up to four disks (which allows up to 1200 megabytes of storage), adds a modem for CPU/CPU link, and supports up to six design stations, three on-line plotters and a line printer. All stations can operate simultaneously with no serious degradation in system response time.

The Eclipse Computer

At the heart of the CPU base system is a high performance Data General Eclipse Computer with memory mapping and high-speed floating point processor. The Eclipse S/230 accommodates up to 256K words of core memory. It also has a microprogrammed architecture that incorporates a comprehensive instruction set. Instructions include:

- 16 bit and 32 bit instructions
- Direct, indirect, indexed, immediate and extended addressing
- Block, word, byte, and bit manipulation
- Fixed point signed and unsigned multiply/divide arithmetic
- Decimal arithmetic
- Extensive logical operations
- Single- and double-word shifts, including hexadecimal shifts
- Single and double-precision floating point instructions
- Extended operation (XOP) for subroutines

Single instructions do work that is normally done by subroutines in operating systems and high-level language compilers. For example, the Eclipse computer DISPATCH ABSOLUTE instruction does

most of the processing typically done by a FORTRAN computed GO TO statement. The LOCATE AND RESET LEAD BIT instruction lets ECLIPSE operating systems quickly identify and allocate an available disk block. A BLOCK MOVE instruction moves data blocks between memory buffers. A complete set of logical operations and bit manipulation instructions is provided for data communications applications.

Eclipse computers use an extended last-in, first-out hardware stack facility to do operations complicated software algorithms normally do, but the stack does them faster and more directly. An Eclipse stack is a series of variable-length temporary storage areas called frames, each easily assigned and randomly addressed. A single SAVE instruction allocates the frame, saves the entire machine state, and sets a pointer to the frame area. A single RETURN instruction reverses the entire procedure. This Eclipse random access stack is far more efficient than traditional serial access stacks. It results in fast subroutine linkages, rapid context switching, and a high system throughput in reentrant high-level language and operating system environments.

The Eclipse computer's fast interrupt facility handles even the most time-critical events. Real-time operating system interrupt servicing is in hardware, rather than slower software routines. The servicing can be tailored to individual peripheral devices and applications with the powerful vector instruction.

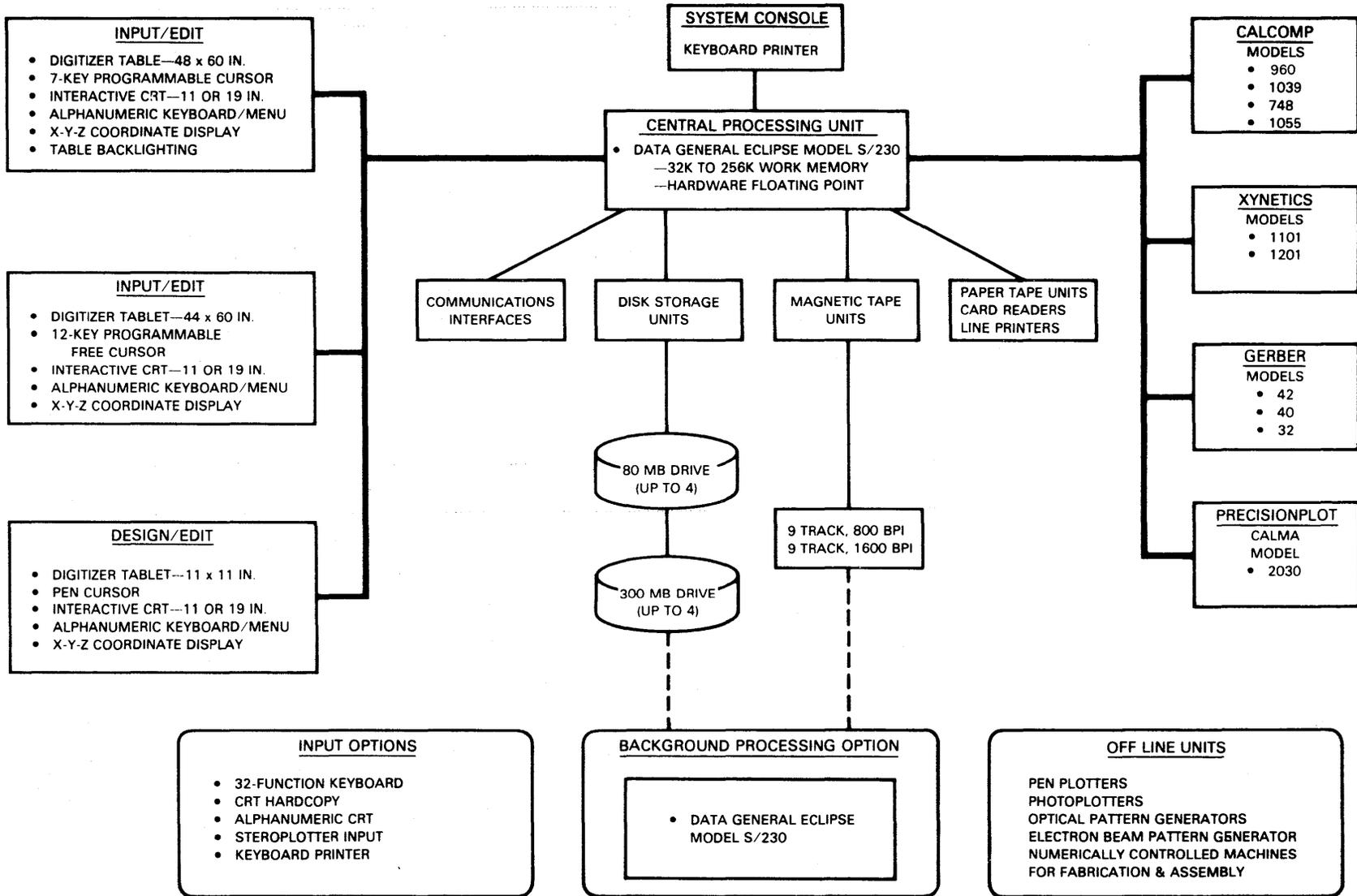
For simple interrupt processing, VECTOR combines fast service with minimum system overhead. The instruction identifies the interrupting device, then directly transfers control to the device handler, all in about 2.7 microseconds (4-way interleaved core). For the most complete interrupt processing, VECTOR identifies the interrupting device, switches from user stack to interrupt stack, saves the machine state, establishes a new priority, reenables interrupts, then transfers control to the device handler. VECTOR typically performs even this most complex function in only 18 microseconds (4-way interleaved core).

The Eclipse Memory Allocation and Protection (MAP) provides program and data integrity in multi-user environments. It provides hardware protection between user programs, and between a program and the operating system.

INPUT

PROCESSING

OUTPUT



HARDWARE CHOICES

MAP manages up to 512K bytes of main memory resource. It allocates memory to each user in 2K byte blocks, up to 32 blocks at a time. The blocks are small enough to make efficient memory use, yet large enough to keep system overhead to a minimum. The same block of physical memory can be allocated to more than one user, allowing procedure and data sharing.

MAP simultaneously holds three dynamic address translation maps--two user maps and a data channel map. The two user maps speed context switching. The separate data channel map lets I/O activity of one user overlap with the program execution of another user.

In addition to address translation, MAP provides several kinds of protection: Blocks can be write-protected so users cannot alter them--an important feature when physical memory is shared. Input/output devices can be made accessible or inaccessible to a user. Each MAP protection function is enabled separately and easily, letting the operating system handle users with widely differing requirements.

The Eclipse Floating Point Processor performs extremely fast single-precision and double-precision floating point arithmetic. High-level languages like Data General's FORTRAN V make extensive use of the Processor, producing high system throughput in computational applications.

The Processor has four separate 64-bit floating point accumulators for floating point arithmetic. Operands stored in these accumulators are quickly available for floating point manipulation, especially in iterative processes like sine and cosine calculations.

The Floating Point Processor operates in parallel with the computer's central processor. High-speed floating point arithmetic can therefore be performed simultaneously with other instruction processing.

There are 56 comprehensive floating point instructions that perform single- or double-precision floating point arithmetic.

Instructions PUSH or POP the entire Floating Point Processor state on or off the user stack, an extremely useful feature for rapid context switching.

FIX and FLOAT instructions allow convenient double precision integer arithmetic with the Floating Point Processor.

The performance of ECLIPSE's Floating Point Processor is comparable to that of large computers. For example, a double-precision floating point ADD takes only 2.4 microseconds, maximum, with all but 1.0 microseconds overlappable with other CPU operations.

ECLIPSE computers interleave and overlap memories to increase speed and system throughput. Interleaving puts sequential memory locations on different modules. Core memories can be interleaved two, four, and eight ways. In four-way core interleaving, for example, each of four consecutive memory locations is on a separate memory module.

Since successive locations are on separate memory modules, and since ECLIPSE computer micro-instructions are overlapped, ECLIPSE accesses the next sequential location before it completely finishes processing the previous location.

The effect on instruction execution time is dramatic. A four-way core-interleaved system reduces a 1400 nanosecond LOAD ACCUMULATOR instruction to 800 nanoseconds. The more modules interleaved, the faster the memory cycle time. Users get higher performance when they need it most, in large memory configurations.

ECLIPSE SPECIFICATIONS

Computer	ECLIPSE S/230
Instruction Length	16-bit and 32-bit
Hardware Accumulators	4
Index	2 hardware, 16 memory
Address Modes	Direct, Immediate, Indexed, Extended, and Multi-level Indirect Addressing.
Operator Control	Control panel with switch register
Memory Cycle Time	800 nanosecond

Standard Memory Size	128K, 16-bit words 2 stations + 1 alpha-numeric display
Memory Fully Expanded	256K, 16-bit words, up to six stations
Direct Memory Access channel	1.25 MHz input, 0.714 MHz output
Standard Arithmetic Element	High speed, single-and double-precision floating point processor.

INSTALLATION DATA

Console Bay Dimensions	55 inches high x 24 inches wide x 30 inches deep (140 cm x 61 cm x 76 cm)
Service Clearances	Front: 36 inches (91 cm) Rear: 36 inches (91 cm) Left Side: 2 inches (5 cm) Right Side: 36 inches (91 cm)

OPERATING ENVIRONMENT

Humidity	10% to 70% relative
Temperature	60° to 80° F (15° to 27° C)
Power Requirements	208 VAC, 60 Hz, 3 phase, 5 wire, 15 amps or 230 VAC, 50 Hz single phase, 20 amps
Heat Dissipation	4970 BTU/hr. maximum

DISK MEMORY

CALMA has combined its own control system with a maximum-capacity storage module drive to create an efficient, reliable means of data collection and retention. This 300 megabyte module allows an almost 400% increase in data storage capacity. Along with increased storage capacity, standard module features include:

- Micro-processor control
- Daisy chaining of up to 4 drives
- Phase-locked oscillator/data separator

- 30 millisecond access time
- Power sequencing
- Acoustic cabinetry
- Proven reliability

Much faster transfer rates are achieved by CALMA's exclusively-designed microprocessor controller. This controller now allows many I/O functions that were previously controlled in software to be completed in firmware, instead.

During operation, the module outputs a variety of sophisticated signals to keep the controller informed of its progress. Along with other information, these signals tell the controller when the unit is up to speed, if the heads are on track, and whether a seek error had been committed or not.

The 300 megabyte drive shares similar styling, common interface, and identical transfer rates with CALMA's other storage units. This compatibility allows greater system flexibility and provides ease of maintenance.

DISK SPECIFICATIONS

Maximum Capacity	300 megabytes (16-bit words)
Rotational Speed	3600 RPM
Number of Heads Type	19 read/write, 1 servo Self-loading, single gap, no erase
Transfer Rate	1.2 million bytes per second
Access Time	
Full Stroke	55 milliseconds, maximum
Average	30 milliseconds
One Track	10 milliseconds
Disk Pack Characteristics	
Type	CDC 9883-91
Number of Disks	12 (10 recording, 2 protective)
Usable Surfaces	20 (19 read/write, 1 servo)

Diameter	14 inches
Coating	Magnetic Oxide

INSTALLATION DATA

Dimensions

Height	36 inches (91.5 cm)
Depth	34 inches (86.5 cm)
Width	23 inches (58.5 cm)

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Heat Dissipation

50 Hz:	4800 BTU/hr. (1200 Kg-cal/hr.)
60 Hz:	4550 BTU/hr. (1138 Kg-cal/hr.)

Power requirements

Voltage	208 VAC, 60Hz, 3 phase or 230 VAC, 5 wire, 50Hz, single phase
Current	25.0 amps turn-on, 8.5 amps normal

MAGNETIC TAPE TRANSPORTS

GDS-II supports both a standard 800/1600 and an optional 800 bpi magnetic tape transport. The 800 bpi transport features 45 inch-per-second tape velocity, while the 800/1600 transport features 75 inch-per-second tape velocity.

Characteristics common to both tape systems include:

- Data read both in forward and reverse directions
- Optical sensing of servo arm
- Automatic servo arm retraction
- Straightforward tape threading path
- All controls on front panel
- Hinged tape deck to provide front access
- Immediate, online access to the CPU

MAGNETIC TAPE SPECIFICATIONS

THE 800/1600 BPI TRANSPORT

Data Density	Nine track, dual-mode, 800/1600 bits per inch
Tape Velocity	75 inches per second (190.5 cm)
Rewind Speed	300 inches per second nominal (762 cm/sec)
Total Speed Variation	± 3% max
Head	Dual or single gap
Number of Tracks	9 IBM/ANSI-compatible
Recording Mode	NRZI and PE compatible
Tape Specifications	Computer grade, 0.5 inches wide, 1.5 ml thick
Reel Size	Up to 10.5 inches (26.6 cm) diameter; IBM hub compatible

INSTALLATION DATA

Physical Size

Height	54 inches (137 cm)
Width	22 inches (56 cm)
Depth	32.5 inches (83 cm)

Power Requirements

208 VAC, 60 Hz, 3 phase,
450 watts max
or 230 VAC, 50 Hz, single
phase, 450 watts max.

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

THE 800 BPI TRANSPORT

Data Density	800 bits per inch
Tape Velocity	45 inches per second (113 cm/sec)
Rewind Speed	150 inches per second nominal (381 cm/sec)

Total Speed Variation	±4% max
Head	Dual or single gap
Number of Tracks	9 IBM/ANSI-compatible
Recording Mode	NRZI and PE compatible
Tape Specifications	Computer Grade, 0.5 inch wide, 1.5 mil thick Mylar base
Reel Size	Up to 10.5 inches (26.6 cm) diameter; IBM hub compatible

Line Length	132 characters maximum
Spacing	
Horizontal	10 characters per inch (3.9 characters/cm)
Vertical	6 lines per inch (2.4 lines/cm)
Characters	96 upper/lower case ASCII, 7 x 7 dot matrix (0.07 x 0.10 inches) (1.8 x 2.5 mm)
Paper	Variable width: 3 to 14.9 inches (7.6 x 37.8 cm)

INSTALLATION DATA

Physical Size	
Height	55 inches (140 cm)
Width	24 inches (61 cm)
Depth	30 inches (76 cm)
Power Requirements	208 VAC, 60 Hz, 3 phase, 5 wire, 2.5 amps or 230 VAC, 50 Hz, single phase, 1.5 amps
Operating Environment	
Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Single Part	115 pound (6.8 kg) paper minimum, card stock thickness of 0.01 inches (0.25 mm) maximum
Multi-Part	2- to 6-part, Thickness of 0.02 inches (0.51 mm) maximum
Keyboard	Standard ANSI keyboard layout, Multikey roll-over
Ribbon	DEC-specified nylon fabric (P/N 36-100558) Spool assembly: 0.5 inches (12.7 mm) wide x 40 yards (36.5 m) long

KEYBOARD PRINTER

CALMA's Keyboard Printer offers many functional and user-oriented features. Variable vertical forms adjustment, variable forms width, and multi-part forms capability are all standard. The 7-wire print head is designed for long life and clear character definition. Other keyboard features include:

- 132 column printing; 10 characters per inch horizontal
- 6 lines per inch vertical spacing
- 96-character ASCII upper/lower case set
- Fine vertical adjustment for accurate forms placement

KEYBOARD PRINTER SPECIFICATIONS

Printing Rate	30 characters per second
----------------------	--------------------------

INSTALLATION DATA

Dimensions:	
Width	27.5 inches (69.9 cm)
Height	33.5 inches (85.1 cm)
Depth	24.0 inches (61.0 cm)

Operating Environment	
Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Power Requirements	(Powered from Central Processing Unit)
---------------------------	--

WORK STATIONS

For varying customer requirements, CALMA provides three types of work stations:

Large Table Work Station--a 48 x 60 inch controlled cursor digitizing table with backlighting for production IC digitizing. This station comes equipped with twinstand, gantry-type digitizer and a button cursor. The resolution is 0.001 inches; the repeatability is ± 0.001 inches.

Large Tablet Work Station-- a tablet the size of a drafting table with pen-stylus or crosshair unrestricted cursor for onscreen design. The resolution is 0.001 inches; the repeatability is ± 0.001 inches; the accuracy is ± 0.005 inches

Tablet Design Station--a design terminal with 12 x 12 inch pen-cursor graphic tablet. The resolution is 100 lines per inch; accuracy is ± 0.01 inches.

The Large Table work station is generally used for bulk data entry. The Tablet Design Station is generally used to interactively generate new designs. The Large Tablet Work Station offers the advantages of tablet with the size of large digitizer table.

The GDS-II system supports dual displays at all of its terminals. The graphics screen, a 19-inch storage CRT, provides a visual presentation of all graphics. The alphanumeric screen, an 11-inch refresh CRT, continuously displays a variety of messages to the user. All dual displays are adjustable for maximum user convenience.

Along with the dual displays, each work station includes a movable alphanumeric keyboard. The unit has 20 function keys, and an A/N keyboard capable of generating 96 separate ASCII character codes. The keyboard is provided with audible indications to augment input/editing operations. An optional function keyboard may also be selected for special symbols, macros or commands not normally on the menu.

WORK STATION SPECIFICATIONS

Graphics CRT

Screen Area	15 inches (38 cm) horizontal by 11 inches (27 cm) vertical, 19 inches (48 cm) diagonal.
Addressable Points	1024 x 1024
Erase Time	0.5 seconds
Writing Rate	4,000 inches per second (10,160 cm/sec)

Alphanumeric CRT

Screen Area	8.5 inches (21 cm) horizontal by 6.5 inches (16.5 cm) vertical, 11 inches (27.4 cm) diagonal
Display Format	25 lines of 80 characters each
Character Type	5 x 7 dot matrix, 0.08 x 0.18 inch character size.
Refresh Rate	60Hz

Large Table Work Station

FEATURES

- Eight-digit X and Y continuous coordinate display
- Half-grid indicator lights
- Programmable function cursor
- Choice of programmable menu and/or 32-function keyboard
- Powered height adjustment
- Counter-balanced table angle adjustment from horizontal-to-vertical

OPTIONS

- 32-Function Keyboard
- Foot-Operated Function Switches
- X, Y Coordinate Display

INSTALLATION DATA

Digitizer Dimensions

Width	76 inches (193 cm)
Depth	58 inches (147 cm)
Height	57 inches (145 cm)

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Total Heat Dissipation	1775 BTU/hour (444 Kcal/hr)
-------------------------------	-----------------------------

Power Requirements	208 VAC, 60 Hz, 3 phase, 5 wire, 20 amps or 230 VAC, 50Hz, single phase, 20 amps
---------------------------	---

Large Tablet Work Station

FEATURES

- 8-digit X and Y continuous coordinate display
- Half-grid indicator lights
- Programmable function cursor
- Choice of programmable menu and/or 32-function keyboard
- Powered height adjustment
- Counter-balanced table angle adjustment from horizontal-to-vertical

OPTIONS

- 32-Function Keyboard
- Foot-Operated Function Switches
- X, Y Coordinate Display

INSTALLATION DATA

Digitizer Dimensions

Width	70 inches (178 cm)
Depth	60 inches (152 cm)
Height	57 inches (145 cm)

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Total Heat Dissipation 1775 BTU/hour 444 Kg-cal/hr)

Power Requirements 208 VAC, 60 Hz, 3 phase 5 wire, 20 amps
or 230 VAC, 50 Hz, single phase, 20 amps

TABLET DESIGN STATION

Tablet

Tablet Area	11 inches x 11 inches 27.9 x 27.9 cm)
-------------	--

Performance

Accuracy	±0.01 inch (0.25 mm)
Resolution	100 lines per inch

Input Device Pen Cursor

INSTALLATION DATA

Dimensions

Width	64 inches (162.6 cm)
Depth	59 inches (149.9 cm)
Height	56 inches (142.2 cm)

Operating Environment

Temperature	60° to 80° F (15° to 27° C)
Humidity	10% to 70% relative

Heat Dissipation 1420 BTU/hour
(355 Kg-cal/hr.)

Power Requirements 208 VAC, 60 Hz, 3 phase, 5 wire, 5 amps
or 230 VAC, 50 Hz, single phase, 5 amps

ON-LINE PLOTTERS

On-line plotters supplied with GDS II include:

Calcomp 1055, 1039, 936, 960, and 748 Plotters

Xynetics 2200, 1200, 1100, and 1050 Drafting Tables

Versatec 8272, 8242, 8236, and 8224 Matrix Plotters

Printronix Printer/Plotter

PERIPHERAL OPTIONS

GDS II provides a complete selection of peripheral options to meet varying user requirements. Included are a paper tape reader, a paper tape punch, a punched card reader, a line printer, and a communication interface.

Paper Tape Reader:

8 channels, 300 cps, supply and take-up reels, panel mounted.

Paper Tape Punch:

8 channels, 75 cps, supply reel, panel mounted.

Punched Card Reader:

80 columns, 285 cpm, table mountable.

Line Printer:

132 columns, 64 ASCII characters, 300 lpm, free standing unit.

Keyboard Printer:

Full ASCII 96 char keyboard 30 cps, 132-column matrix printer, free-standing console.

Keyboard Display:

Full ASCII 96 character keyboard, 30-480 cps, 2000 character video display table mountable.

Synchronous Communication Interface:

This synchronous interface enables Remote Job Entry emulation. With the appropriate software GDS-II can emulate IBM's 2780 RJE Terminal, or IBM's HASP Terminal or CDC's UT200 Terminal, or Univac's 1004 Terminal.

FUTURE PRODUCTS

The VMD

Recent progress in CALMA research has brought several important and encouraging results.

The most significant of these concerns CALMA's work with the Vector Memory Display (VMD). By combining the benefits of both vector and raster memory, CALMA's VMD will revolutionize on-screen design. This innovative display features a high-resolution, flicker-free image along with instantaneous data identification and lightning-quick refresh speed.

The VMD uses a vector memory to store data, plus a raster memory to scan the vector memory and drive the screen. Data comes from the Eclipse CPU memory into vector memory. To load a portion of a circuit into the VMD, the user simply windows all desired data. Once loaded, the CPU is not needed except to update the disk. A large amount of data can be transferred without throughput degradation.

The VMD has an 8085 command processor, so all commands will be sent to both the VMD and the Eclipse CPU.

Standard VMD features include:

Large Vector Memory

The vector memory stores up to 128K words, enough space to accommodate approximately 60K vectors. (Each side of a polygon is a vector.)

With the VMD, all or any one of four command menu levels can be displayed on the graphics screen. Menu configuration can be altered so as not to interfere with the actual drawing. For example, a 8 x 5 matrix of buttons, located along the bottom of the screen, can be broken up into two 2 x 10 matrices and placed perpendicular to each other in any corner of the display. On-screen menus may be any size (up to 512 buttons) and are limited only by screen space.

To execute commands, the user simply has to touch the desired on-screen button with the cursor. He never has to look away from the screen.

Selective Erase

Because the VMD stores its data in a vector memory, any shape on the screen can be instantly deleted or altered without forcing the entire image to be redrawn. Any change or deletion is instantly displayed as it happens.

Multiple Viewports

The VMD allows different magnifications of the same image to be displayed on the screen at one time. In this mode, most of the screen is taken up by a view of the entire circuit. However, the portion of the image being edited is displayed in a magnified inset window. Cursor position determines what data is contained in the inset window. As the cursor is moved, the corresponding magnified data will move, also.

With this magnified inset window, the user can see, close-up the area he is editing while also being able to see his position with respect to the entire circuit. This feature is especially useful in routing signal traces, where the designer needs to be aware of both the details of local obstacles and the general direction in which he is headed.

Insert format can be reversed with the entire database viewed in the inset window and the magnified editing area occupying the rest of the screen. The menu on the screen remains the same regardless of magnification. Multiple viewports will support multiple cursors.

Continuous Pan and Zoom

Pan allows the designer to quickly change the location of the select view window. This effectively removes screen size limitations, because the designer can pan over the entire database at any magnification. Pan can be implemented in the horizontal, vertical, or diagonal direction.

Zoom changes the magnification of the displayed data and can be operated by continuous command or by a predefined magnification number.

Menu size and configuration remain constant during these operations.

Context Switching

The VMD's context switching feature allows the user to switch instantly from one viewing layer of an IC design to another. Line coding can be instantly changed, menu levels switched, and the designer can even blink a layer on and off.

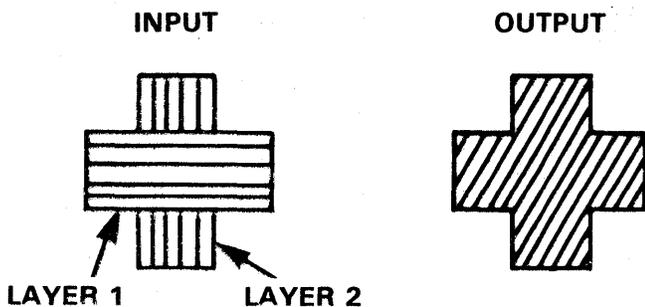
All context switching features are accomplished with minimum assistance from the CPU.

Advanced Design Rule Checks (DRCs)

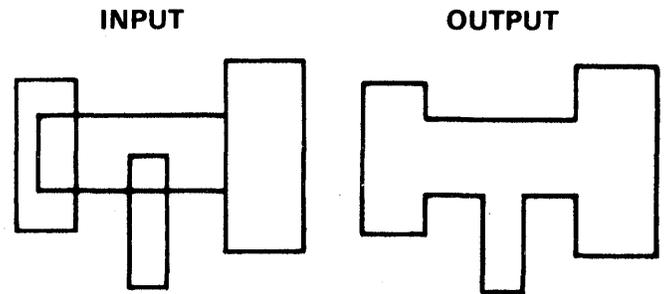
CALMA is currently working to augment its already extensive collection of DRC's. These new checks will perform a variety of sophisticated functions including:

Logical Operations Between Layers: This program will compute the geometric intersection (AND), union (OR), or difference (AND NOT) of two designated layers.

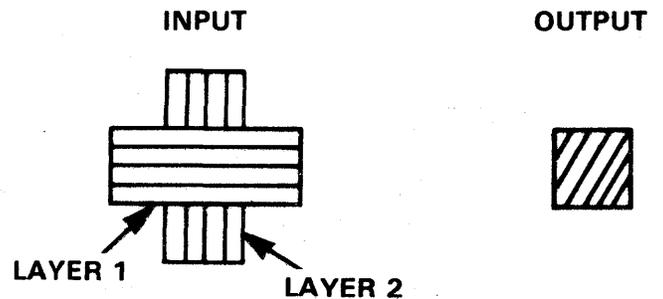
- **OR Operation** - An OR operation on 2 layers generates an output layer containing only boundary lines and enclosing all areas that were contained in either of the input layers. If the 2 requested input layers are the same, the output layer will contain only the boundary lines of the input layer; all lines involved in overlaps or abutting polygons are removed.



- **Use of OR** - Consider the case of large number of overlapping polygons on a single layer. Performing an OR on this layer will result in a considerable simplification, and will result in more rapid Design Rule Checks if the generated layer is used instead of the input layer.

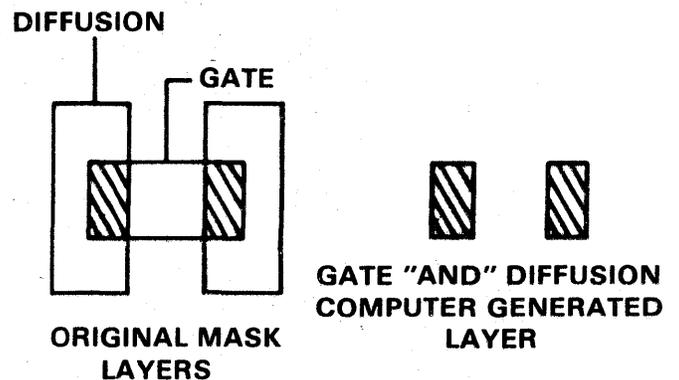


- **AND Operation** - An AND operation on 2 layers generates an output layer containing only the boundary lines of those areas that are common to both input layers.



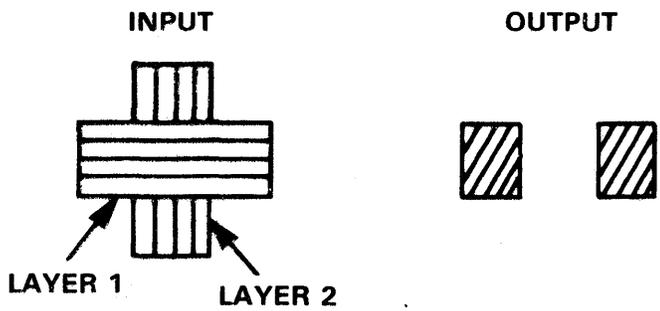
If the same layer is requested for both input layers, a special operation is performed, and the output layer encloses only those areas which are contained in more than one polygon from the input layer.

- **Use of AND** - Consider a process where the gate layer must overlap the diffusion by a given amount. To verify this the user will proceed as follows:

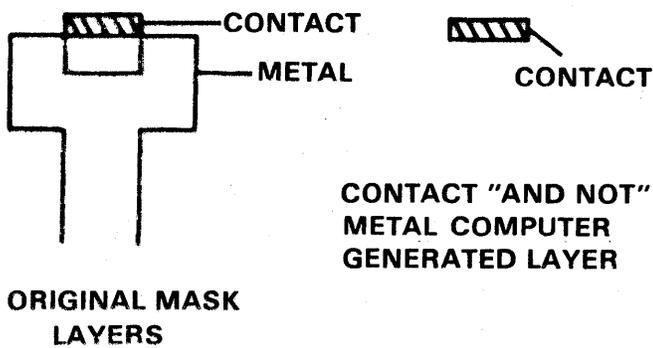


- A. Computer generate layer by "anding" diffusion with gate.
- B. Perform a standard internal width check for the computer generated layer, specifying correct width.
- C. Any widths that are less than specified will be flagged as errors.

- AND NOT Operation - Unlike the OR and AND operations, the AND NOT operation is dependent on the order of the input layers. The generated layer contains the boundary lines of those areas enclosed by the first requested layer and not enclosed by the second requested layer. The case when both requested inputs specify the same layer has no meaning.

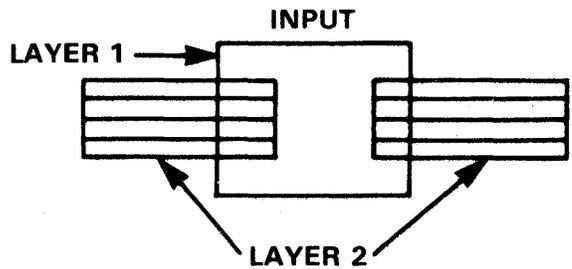


- Use of AND NOT - Consider a process where all contacts must be covered by metal.
 - A. Generate a mask layer consisting of contact "and not" metal.
 - B. The resulting mask layer will contain all violations of this design rule.



The above examples indicate how the various capabilities of the DRC system may be used for the designer to formulate a set of checks appropriate to his process.

Intrusion: This program checks for specified maximum or minimum intrusion of one polygon enclosed within another. Intrusion checks always involve two layers and only boundary lines are considered.



All widths that are less than a specified minimum or more than a specified maximum will be flagged as errors. (Exact definition of specifications in progress at time of publication.)

Connectivity Computation: This program will determine the connectivity relations among all the polygons in a layer or set of layers. The program assigns a connectivity number to each polygon such that two polygons have the same connectivity number if and only if they belong to the same connection region. (Exact definition of specifications in progress at time of publication.)

Edit-in-Place

As previously mentioned in the System Operation Section, before the user can implement design and edit commands to GDS-II data, this data must first be located by the system. While command selection and execution are extremely fast on the GDS-II System, significant improvements in throughput speed are always of concern to CALMA. Research and

Development is currently working on a high-speed database editor, called "Edit-in-Place," which will actually perform edits on any part of the selected database. When used in conjunction with on-screen command menus, alphanumeric feedback, and fast response, Edit-in-Place will give GDS-II the ultimate capabilities for true on-line IC design.

Right now, editing structures nested within other structures is not any easy task for designers using conventional graphics systems. With Edit-in-Place, however, the GDS-II System will be able to perform any edit to any structure nested up to 10 levels deep.

Suppose, for example, that the designer is working with a complex gate array or microprocessor with over 1 million vectors. Displayed entirely on the screen, a circuit like this would be nothing but a solidly-filled rectangle. Now suppose the designer zooms in 200X to a single contact within a gate structure nested 5 levels deep. To edit this contact with conventional systems, the user must first identify which structure the contact is in, then discard all the other data, call the specific structure containing the contact, bring it up to view and perform the edit.

In contrast, with GDS-II's Edit-in-Place, the user will be able to view the entire circuit all the time. After zooming to the contact in question, the user can then specify the final structure command and point at the

contact. No matter how deeply nested, the system will rapidly and automatically locate the structure. The user can then edit a specified item with the surrounding circuitry visible and editable as well. No reloading is required to edit any data displayed on the screen at any time.

Several options will accompany this new Edit-in-Place command. The user may:

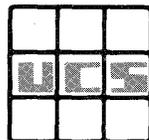
1. Edit all versions of a structure
2. Copy for edit any specified number of versions of a structure
3. Copy or rename any particular structure for editing

Edit-in-Place is designed to limit its search to only the areas specified. Once a particular structure is pointed out, all edits and searches are restricted to the structure in question. This limited-search feature provides for very high speed edits anywhere within the entire database. Moreover, edits can occur to copies of an original structure (where complex changes might require retrieval) or directly to the data in a library.

There are no restrictions to the size or quantity of data involved. High-level structure nesting is encouraged because smaller structures are edited faster.

CALMA

INTERACTIVE GRAPHIC SYSTEMS
A division of United Computing Systems Inc



527 Lakeside Drive ▪ Sunnyvale, Ca. ▪ 94086