THE STANFORD UNIVERSITY DESIGN SYSTEM (SUDS)

OVERVIEW

Opening

- Engineering Information Control

Introduction

- What is SUDS?

- How is it used?

- Available Services

The Environment

- System Configuration (CADNET System description)

- Programmable Graphics Processor (SUDS User Station)

SUDS Library

- General Description

  a. .DRW file
  b. DIPS.DIP file
  c. Public Libraries

SUDS

- SUDS Overview

- SUDS (Drawing Program)

Wirelister Program

- General Description

- Design Aids

  a. Logic Simulation
  b. Placement Optimization
  c. Delay Calculation

- SUDS to other DEC programs

  a. Wirewrap
  b. Multiwire
  c. KPL

The SUDS Processes

- SUDS to Idea
- SUDS to Caldec

Conclusion

- SUDS vs Manual

THE STANFORD UNIVERSITY DESIGN SYSTEM (SUDS)

INTRODUCTION

What is SUDS?

The Stanford University Design System (SUDS) was originally developed at the Stanford Artificial Intelligence Lab by Dick Helliwell. It is a schematics/logic drawing package which not only offers hardcopy capabilities, but is also the nucleus of a data base which provides input to the IDEA and CALDEC automated printed circuit board design systems operative here at Digital. The data entered into SUDS for generating logic schematics will also generate wirelist information, and layout verification for use by the layout designers. In short, SUDS allows you to draw a schematic using interactive graphics, and then feed this drawing directly to CALDEC, IDEA/PCLS, SAGE 2 (simulator), multiwire, or wirewrap without ever hand coding a wirelist. For the user, the most significant factor is that the product of the system is data files that can be handled by a computer.

How is it used?

The Stanford University Design System (SUDS) is configured to run on either DECsystem 1Ø or 2Ø operating systems with graphics terminals. SUDS was specifically designed for schematics, but its data structure and command language are general enough to support a veritable multitude of applications from flow charts, floor plans, and block diagrams to logic and circuit schematics. In addition, SUDS has an extremely powerful Macros (command repetition) facility and a convenient set mode that allows a subset of schematic elements to be manipulated as a single item. The SUDS single key-stroke command type language may initially feel cumbersome, but the SUDS user will quickly become adapted to the Stanford keyboard and the total Stanford University Design System.

Reduction of High Design Time and Cost

In general, SUDS (or most any automated schematics system) reduces the high design time and cost associated with getting a product out. It accomplishes this by:

- Eliminating the manual re-translation of data represented by a schematic (parts list, wire list) into a machine readable format for subsequent CAD processing.

- Eliminating the manual re-drawing of print sets to standards.

- Eliminating the disagreement between the print sets and the physical design data.

- Supplying control over the print set to the Engineer.

- Reducing the hassles involved in discerning the incremental difference between an ECOed design and it's base rev.

SUDS Interface

Specifically, many interfaces exist in SUDS to allow you to communicate with other CAD processes used at DEC. Some examples of these interfaces and other capabilities follow:

- Draw and post process plot diagrams.

- Extract the wirelist from the SUDS drawing data base that is suitable for input to the CALDEC and IDEA P.C. (and I.C.) layout systems.

- Extract wirelist for input to the SAGE 2 simulator.

- Extract a wirelist for input to the CALMA I.C. layout system.

- Obtain reports on data such as gate-loading, unused pins, naming violations, parts lists, used on information and more.

- Combine data together from many modules for wirewrap of a backplane.

- Obtain a wirelist for wirewrap of prototype board.

- Create procedures (Macros) for sequences of often used or complex drawing commands.

- Create and maintain cover sheets to print sets (contents, etc.)

- Create simple mechanical diagrams.

- Draw auxiliary data such as Capacitor Drawings, Parts Lists, etc.

- Do flow charts and architectural diagrams using SUDS.

SUDS is actually only the starting point in a whole array of computer-aided (CAD) tools under development at DEC.

Personnel and SUDS Usage

We assume that most engineers will be interested in running SUDS.... at least at the beginning. In fact, it is recommended that the engineer become quite conversant in SUDS as it is a complex system and you can only get the best use out of a tool you know well. However, we expect the majority of the actual use to come from a technician or draftsperson as schematics are currently their responsibility, anyway. Most important, however, is the need for at least one person in each group to become the SUDS "guru" to answer questions about it's operations, interface with library support and software support and direct the module release process (a non-trivial set of procedures.)

OK, So How Do I Actually Use It?

We have given much thought to exactly how to use all the components of SUDS and if it would be possible to document this. We have come to the conclusion that there can be no one document to serve all of the general Engineering communities. Each group has their own personnel, their own area and their own way of doing a design. Although there are points in the general process where certain data is needed or certain functions ought to have been performed, the way to get there can vary greatly.

What we are suggesting is that each group sit down with their SUDS "guru", Engineering Manager, the local Engineering Services Site Manager and perhaps, the SUDS trainer and come up with a definitive document of their own process flow (refer to D-FD-SUDS-Ø-FLOW). This should include the checks to be performed and the functions with long lead times (e.g. library additions, purchase specs, etc.) along with the normal processing. It may take a bit of time, but we are confident this time spent up front will more than compensate for itself through lack of lost time later.

So, What is the Suggested Module Design Process Using SUDS?

A. The best time to capture the logic information (schematic) is during it's creation when you are scribbling on envelopes and changing logic significantly. At this time, SUDS can be operated by the Engineer or technician. The best environment to allow this is having the SUDS terminal right in or near the lab.

B. When schematics have been developed to the breadboard stage, a wirewrap board can be produced in a matter of days or a "quick and dirty" P.C. layout in weeks... both using the data base created in (A) by the original designer.

C. During debug, groups of changes are entered to the on-line SUDS data base. This assures the data base always matches the breadboard.

D. After debug, the final data base from the debugged schematics is sent directly to P.C. layout with no hand coding. At this time, the print set may have to be brought to DEC Standards. However here, as in any stage in the game, two versions of the data base can be compared automatically and a wirelist of the changes reported.

E. Finally before Limited Release (LR), the MIF (data like artwork for manufacturing) can be compared to the wirelist from the Engineer's print set assuring a match.

Available SUDS Services

Library Support

Library support means a central group that enters and helps define new bodies (parts) into the SUDS library and causes it to be distributed to all sites. All user library parts requests (see Appendix 1) are to be submitted to Engineering Information Control (E.I.C.) library group with preferred drawing shape, specification for the part (DEC Spec, if it exists), and a project name and charge number.

Training

Training is provided by Engineering Information Control. Please contact the Training Coordinator, ML4-2/E90, DTN: 223-9710 to receive a course catalog and/or related periodicals on a continuous basis.

Documentation

Currently, the only documentation is SUDS.TXT written by Dick Helliwell (Digital - Marlboro). The LCG Group (Marlboro) is in the process of writing a total SUDS User Guide with complete definitions of all SUDS commands. During training classes, a comprehensive training/user manual will be made available to all trainees to hasten the learning process. This text is only an introduction in how to use the SUDS Design System.

KEY SUDS PEOPLE

The following is a list of key people and their areas:

Name	Function	Location
Bob St. Cyr	CAD Tool Training Manager	ML4-2/E90 223-8083
Cindy Pekkala	Training Coordinator	ML4-2/E90 223-9710
Tig Richardson	CADSE Support Supervisor (SUDS Software Release)	ML3-5/T28 223-3325
Norm Rheault	SUDS Trainer	ML4-2/E90 223-8789
Tom Witowski	Library Support (SUDS/ALM)	ML4-2/E90 223-4242
Jim Fleming	Engineering Process Systems	ML4-2/E90 223-2287
Pat Barry	IDEA Trainer	ML4-2/E90 223-6167
Nancy Moore	ECO Process Systems	ML4-2/E90 223-3172
Moe Marchand	IDEA Trainer	ML4-2/E90 223-5235
Don DiMatteo	IDEA Trainer	ML4-2/E90 223-2438
John Hartling	SUDS Support/ E.S. Manager	CX 522-2203
Dick Helliwell	SUDS Support (Father of SUDS) Software Engineer	CX 522-2009
George Bourbeau	SUDS Support	CX 522-2116
Irv Prais	SUDS Support	CX 522-2015

THE ENVIRONMENT

SUDS - SYSTEM CONFIGURATION (CADNET SYSTEM DESCRIPTION)

All the SUDS programs are run on CADNET systems and so it is necessary to understand how it functions and how to communicate with it. The following information is a brief and simple explanation of the CADNET system and how it serves the SUDS user.

The CADNET system on which the SUDS run on is a DEC system-1Ø time sharing system that contains XXXK of core (main memory) and a monitor program that schedules and allocates processing time and memory required for each user. Time sharing enables multiple users to have access to a computer system at the same time by sharing the resources of the computer between users. In addition to main memory, the DECsystem-1Ø provides secondary memory in the form of Disk Packs. These disk packs are used for storing many types of data as well as allocating each user a maximum amount of unique disk blocks for files that may be created while running programs (see figure 2-1).

In order to use the CADNET sytem, all SUDS users must first obtain a project programmer number (PPN). A PPN provides the user with three important functions:

1. A working area on disk where the user can store data files created by running programs. Other portions of the disk will be used for other PPN's to store programs and data files.

2. The user will identify this area by a unique PPN; a number representing a project and the idividual's number seperated by a coma (e.g. 362,12Ø)

3. To protect each user from others, a unique password is given to each user when the PPN number is assigned.

Each time a user runs on the CADNET system, they are required to log into the computer. This process will cause the computer to ask the user for a PPN, password, cost center, and charge number. This is necessary to tell the computer where your files are stored, and what department and project to charge for the computer time you use.

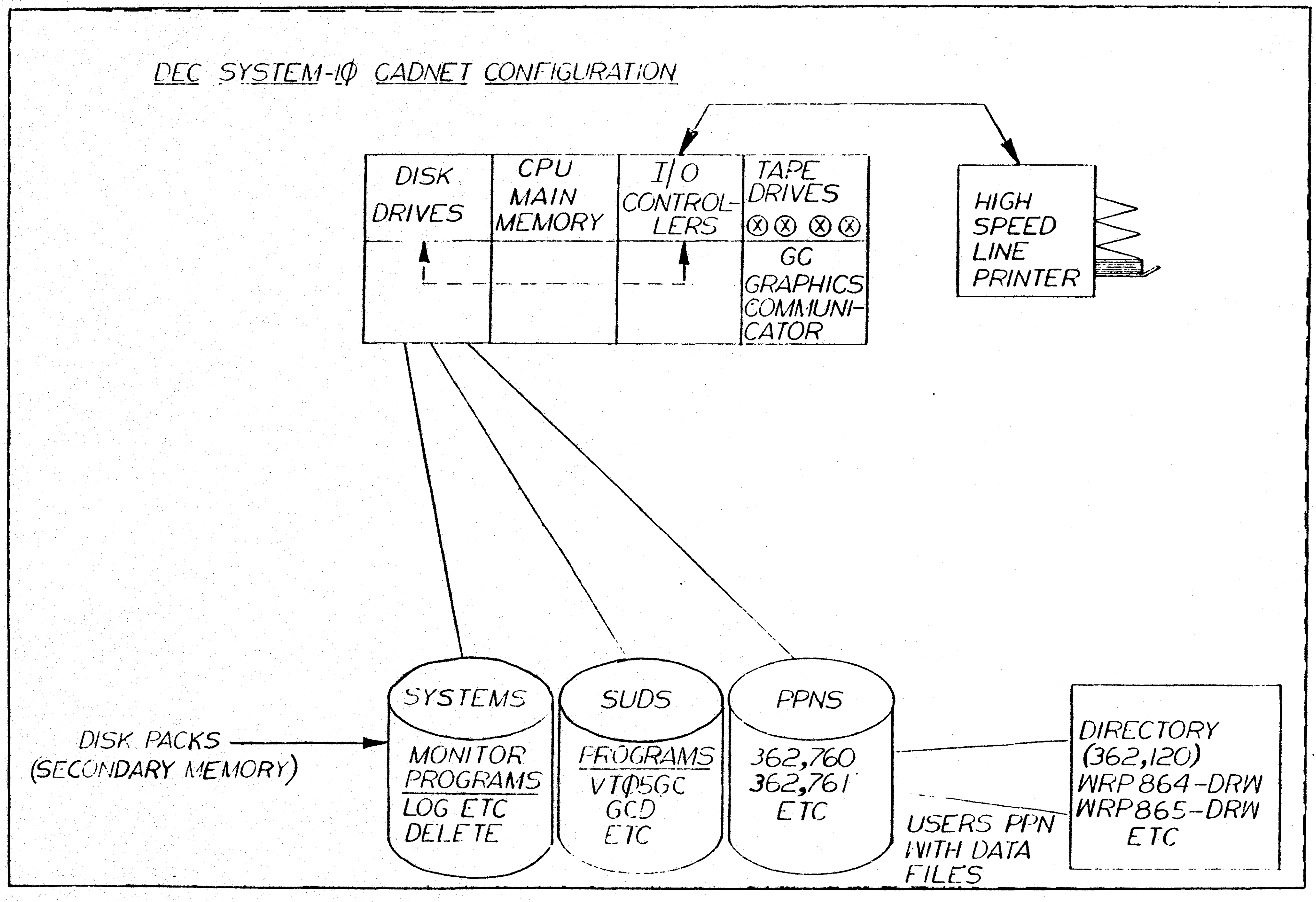


FIGURE 2-1 SUDS SYSTEM CONFIGURATION

PROGRAMMABLE GRAPHICS (SUDS USER STATION)

A. Hardware

The basic SUDS system user station is a subsystem comprised of a program that links with internal system programs to communicate through the DECsystem-1Ø to display a SUDS designer's layout on a programmable graphics processor (PGP6Ø). The PGP6Ø is comprised of an 11/34 minicomputer attached to a VS6Ø graphics display and an LA36 teletype. The SUDS designer also has the option of using a lightpen and button box, which are attached to the PGP6Ø as a means of interacting with the graphics display tube (see figure 2-2).

Currently there are two basic configurations supported by CADSE: a PGP configuration using DMC-11 communications and a GT4Ø-like system using DL11's and an asynchronous communications system. In either case, a GT4Ø or GT6Ø graphics terminal may be used.

The host system must be DECsystem-1Ø in CADNET for DMC11 configuration or any DEC1Ø or 2Ø for the DC11 communications.

B. Software

SUDS is written in MACRO-1Ø assembler language. It is heavily conditionalized to allow for different configurations and operating systems (ie: TOPS-1Ø or TOPS-2Ø). The only special systems software needed is the PGP DMC-11 communications system standard on all CADNET machines.

In both communication configurations, there is piece of software that is resident in the SUDS terminal (VTØ5G or VTØ5GC) that handles the keyboard and display. This is down line loaded from the host (-1Ø or -2Ø) or loaded from local disk storage. The current drawing program is GCD. Also, data bases are all stored as disk files.

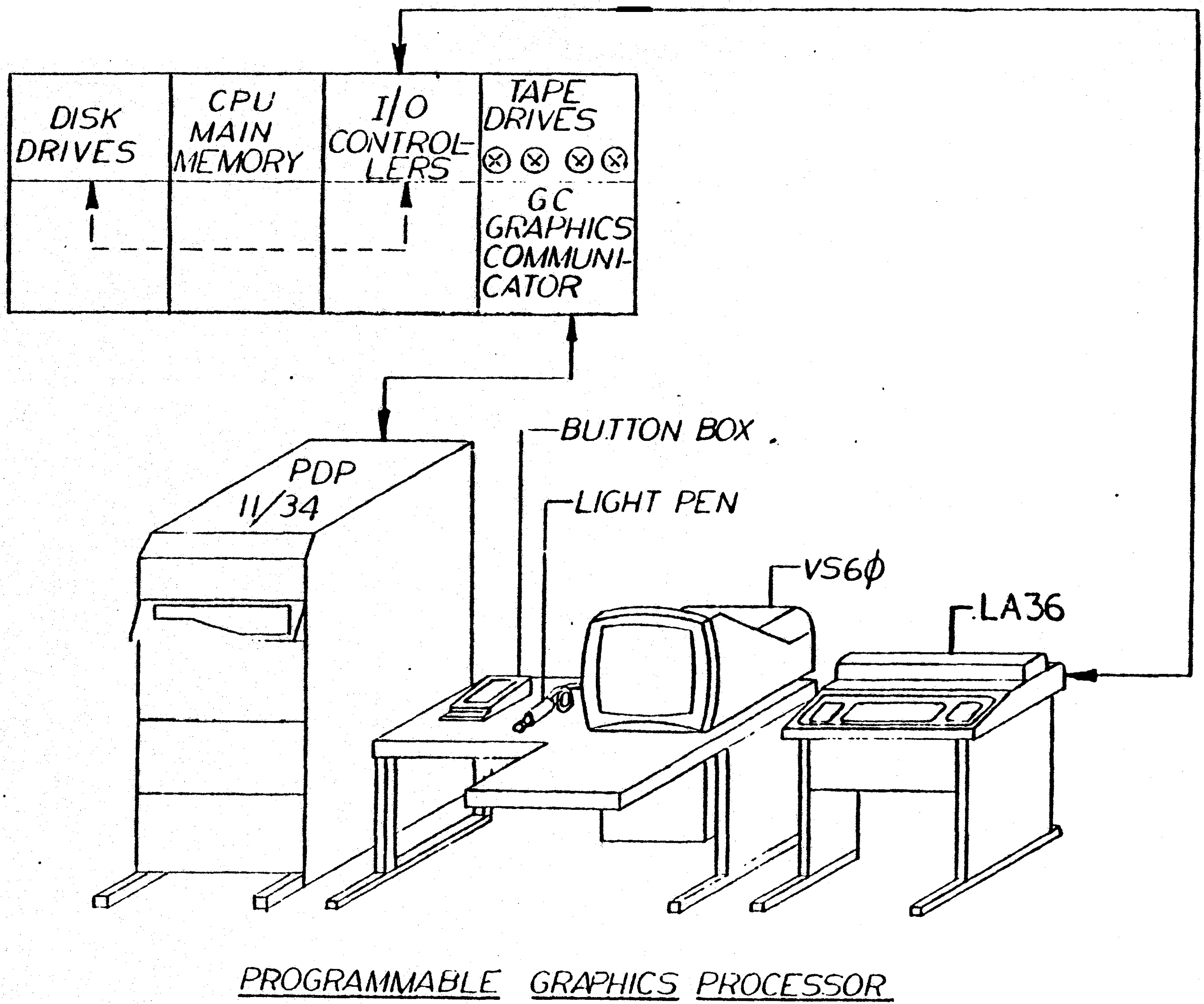


FIGURE 2-2· SUDS SYSTEM USERS STATION

HARDWARE COMPONENTS OF CAD TOOLS AND APPROXIMATE COST OF TOOLS

GEMS

PDP8E, 24K memory, DECtape, VT01-A scope, Bendix digitizer	$38K

CALDEC

PDP15, 96K MEMORY, DECTAPE, VT04-A SCOPE, RF15/RS09	$47K

SUDS

PDP11/34, 32K memory, communication link, VR17-LC/VT11 (SUDS only)		$ 9K
PDP11/60, 64K memory, communication link, VS60 (SUDS or IDEA)	Old price-	$24K
(Must have a PDP10 or 2020 host)	New price-	$37K

IDEA

Terminal -	PDP11/60, 64K memory, communication link, VS60-AA Old price-$24K	New price-	$37K
Host-	PDP10, 512K memory, communication link, 4-RP06 disks, LP10, 2 magtapes (1010) DECtape		$293K
	2020, 512K memory, communication link, 2-RM03 disks, 1-magtape (TU77)		$53K

UNIGRAPHICS

PDP11/70, 512K memory, communication lines, 4 work station graphic terminal, 2-RM03, 1-magtape, 1-plotter, unigraphic software*, 2-copiers, spares	$310K

*Unigraphic licenced software price being negotiated for substantial savings on total cost of system.

* per graphic terminal	$22K

APPLICON

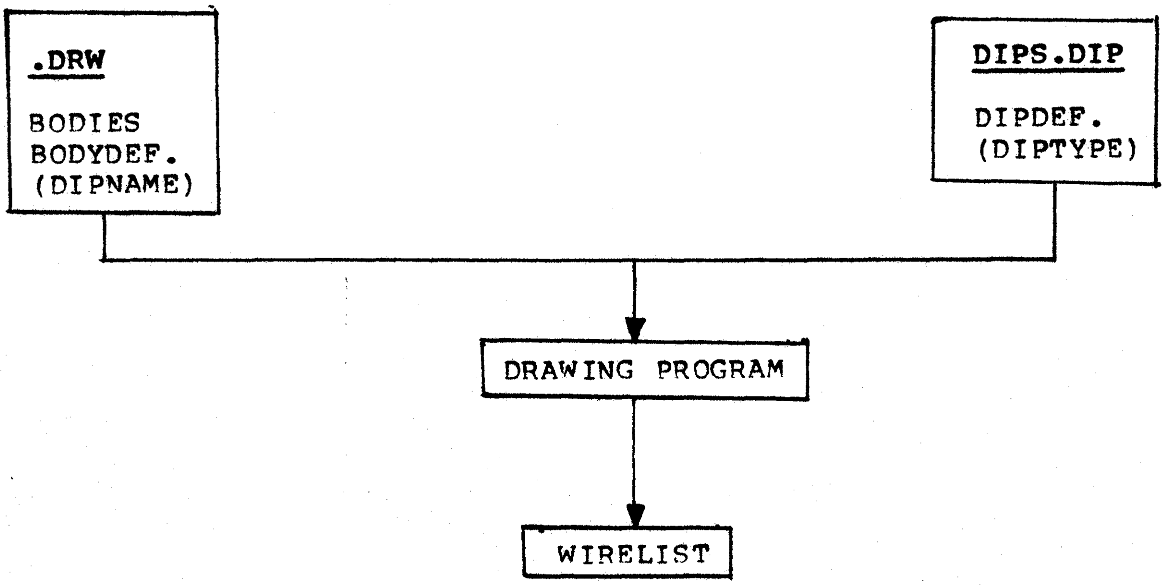
PDP11/34, 64K memory, Applicon software, 1 magtape,
1 large disk, 2-graphic/raster display,
2-12x12 tablets, 34x44 tablitizer terminal $250K

VAX BASED LAYOUT SYSTEM

Development system only - VAX/11/780 (STAR), 2 megbyte
memory, 2-large disks, 1-magtape, 1-line printer,
communication lines, VS60 and interface $100K

Production layout system is aimed at a low end VAX C.P.U. (Comet, Nebula) with a total cost in the neighborhood of $50K ± $10K.

SUDS LIBRARY



SUDS LIBRARY

The SUDS Library is comprised of two sub libraries: .DRW and the DIPS.DIP, which are used in tandem with each other in preparing drawings and listings necessary in the designing of P.C. boards.

SUDS.DRW

The SUDS .DRW library consists of a collection of .DRW files which contain well defined graphic symbols and their pin identifications. These symbols are used by the designer to create graphic presentations on the VS6Ø display. Depending on the file selected from the library, the designer may have access to electrical symbols, or symbols used in designing flow charts, block diagrams, and floor plans. Figure 3-1 contains some of the electrical symbols stored in the .DRW library.

DIPS.DIP

The purpose of the DIPS.DIP libary is to provide information on the electrical characteristics of the components represented by the .DRW library symbols.

Each file within the DIPS.DIP library is automatically associated with a specific electrical graphic symbol, from the .DRW file, whenever that symbol is used. This means that the output will contain not only the .DRW symbols, but the associated DIPS.DIP files, other computer programs (SAGE, IDEA, CALDEC, MULTIWIRE, etc.) will be able to test and check the electrical properties of the completed design for accuracy.

When the designer is using an electrical symbol, he has the option of referencing the associated DIPS.DIP file to add or extract additional information. For example, assume the designer is working with a 14-pin IC, part #19-Ø5575-ØØ. In the .DRW library, this part is represented as four gates with the same symbol (see Figure 3-2). To distinguish one gate from another, the designer can reference the associated DIPS.DIP file #74ØØ (the number shown within the symbol) to acquire the listing in Table one. The designer can then use the next-to-last column of the listing (called section) to label the gates in their appropriate order. In this case, the gates would be numbered as shown in Figure 3-2. For another example, assume the designer wishes to obtain the part number for a 1ØØØHMS resister he is using (shown in top row of Figure 3-1). By referencing the DIPS.DIP RES file (the file for all standard resistors), the designer can obtain the standard type listing shown in Table 2. The designer can then reference the "P" sublisting to obtain the 13-ØØ229-ØØ part number.

FIGURE 3-1

Samples of Electrical Symbols Found in the SUDS .DRW Library

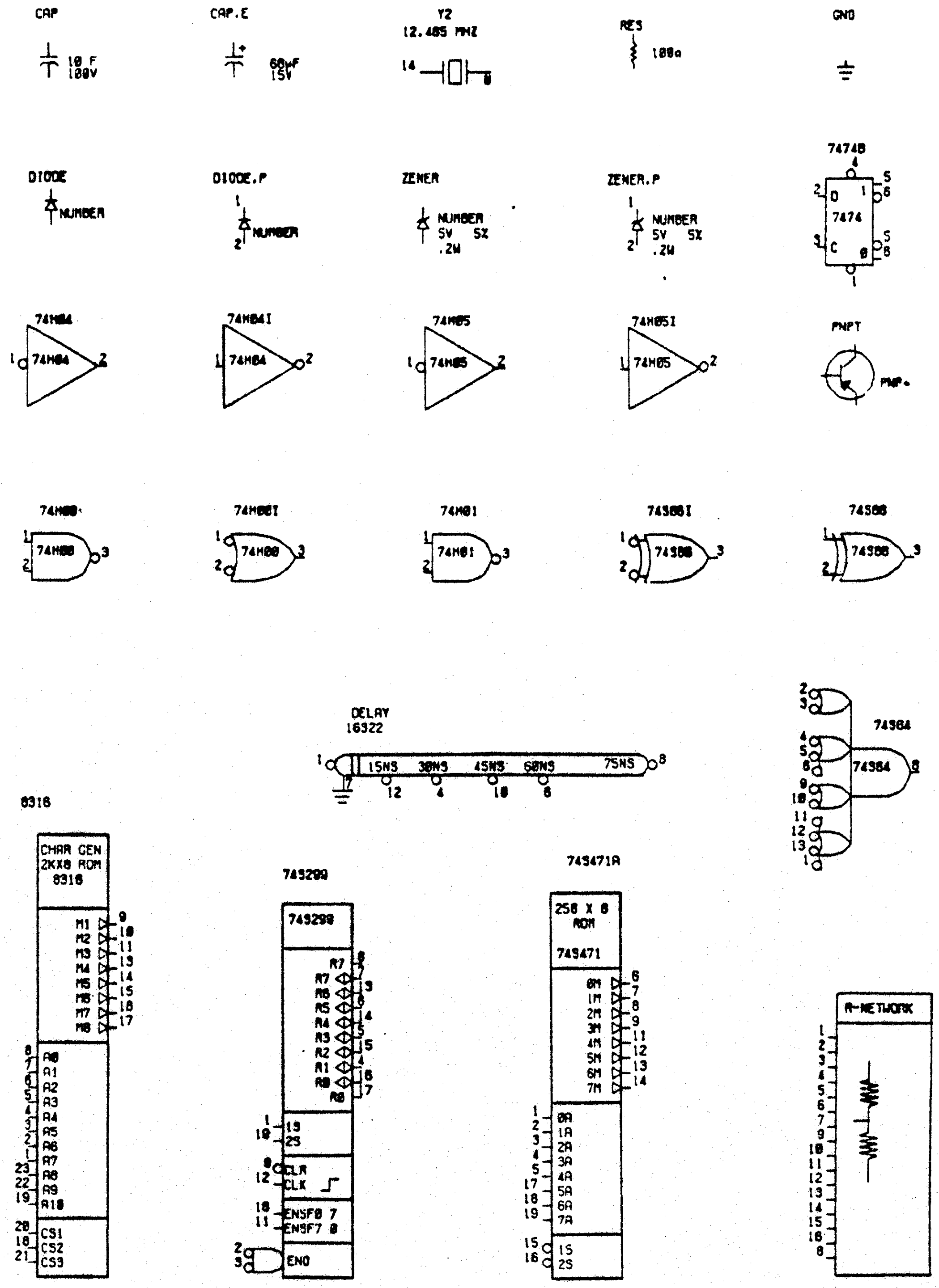


FIGURE 3-2

.DRW Presentation of Part #19-Ø5575-ØØ

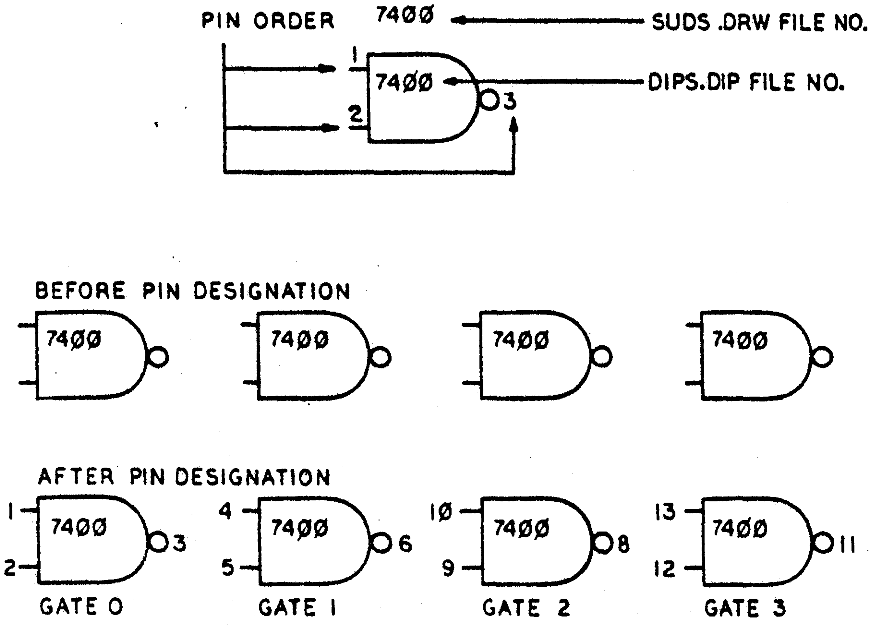


TABLE 1 **DIPS.DIP File Listing for .DRW Symbol #74ØØ.**

```
 P7400
7400     14 PINS, VG PACKAGE TYPE = 3
         PIN #    TYPE     LOW      HI       USE      PS       SECTION RULE
         1        TIS      -1.60    0.04              1        1/0
         2        TIS      -1.60    0.04              1        2/0
         3        TO       16.00    -0.40                      3/0
         4        TIS      -1.60    0.04              2        1/1
         5        TIS      -1.60    0.04              2        2/1
         6        TO       16.00    -0.40                      3/1
         7        TG
         8        TO       16.00    -0.40                      3/2
         9        TIS      -1.60    0.04              3        2/2
         10       TIS      -1.60    0.04              3        1/2
         11       TO       16.00    -0.40                      3/3
         12       TIS      -1.60    0.04              4        2/3
         13       TIS      -1.60    0.04              4        1/3
         14       TV       5.00V    22.00MA

*A7400
DIP PROPERTY SUB-MODE!
**P
PART NUMBER

*19-05575-00
**E
DIP SUB-MODE!
*
```

TABLE 2 - Excerpts from RES Listing

```
 PRES
RES       2 PINS
          PIN #    TYPE    LOW     HI      USE     PS      SECTION RULE
          1        I       0.00    0.00                    1/0
          2        I       0.00    0.00                    2/0

*ARES
DIP PROPERTY SUB-MODE!
**P
VALUE           RATING          TOLERANCE       PART NUMBER

 .04 OHMS        2W              2%             *13-17515-00
 .10 OHMS        3W              1%             *13-15240-00
 .20 OHMS        5W              3%             *13-13358-00
 1 MEG OHMS      1/4W            5%             *13-09595-00
 1 OHMS          1/4W            5%             *13-17522-00
                 2.0W            10%            *13-14350-00
 1.1K OHMS       1/4W            1%             *13-02645-00
 1.2K OHMS       1/4W            1%             *13-14491-00
                                 5%             *13-01320-00

     ⬇             ⬇              ⬇                   ⬇

 110 OHMS        1/4W            5%             *13-11523-00
 113K OHMS       1/4W           .10%            *13-09296-00
 120 OHMS        1/4W            5%             *13-00247-00
 121 OHMS        1/8W            1%             *13-02957-00
 128K OHMS       1/4W            1%             *13-05516-00
 130K OHMS       1/4W            1%             *13-14252-00
 137K OHMS       1/4W            1%             *13-05422-00
 140 MILLIOHMS   5W              1%             *13-10189-00
 147 OHMS        1/4W            1%             *13-02874-00
 150 OHMS        1/2W            1%             *13-12699-00
                 1/4W            5%             *13-00250-00
                 1W              10%            *13-00256-00
                 2W              5%             *13-00257-00
 150K OHMS       1/4W            5%             *13-02396-00
 178 OHMS        1/4W            1%             *13-11422-00
 180 OHMS        1/4W            5%             *13-01322-00
 196 OHMS        1/4W            1%             *13-02956-00
 200 OHMS        1/2W            5%             *13-02381-00
                 1/4W           .10%            *13-02962-00
                                 5%             *13-11522-00
 215 OHMS        1/4W            1%             *13-05123-00
 220 OHMS        1/4W            5%             *13-00271-00
                 1W              5%             *13-13347-00
 237 OHMS        1/4W            1%             *13-04857-00
 261 OHMS        1/4W            1%             *13-02873-00
 270 OHMS        1/4W            5%             *13-01972-00
```

PUBLIC LIBRARIES

The user can get the body definitions from any library by using the GETLIB command. He can also inspect the contents of any library that has been gotten by using $L to list the names of the bodies on the terminal or $$L to list them in a file to be printed. He can also make a visual inspection of the symbols contained in the libraries by getting a set of show drawings for them from the site librarian.

The presently available libraries are these:

PTTL	All plain TTL dips (7404)
HTTL	All high speed TTL dips (74H04)
STTL	All Schottky ITL dips (74S04)
LSTTL	All low power Schottky TTL dips (74LS04)
PARTS	Discrete components (resistors, switches, etc)
NCOMP	Hash mark and ground symbols
CMOSBD	All CMOS dips (4044)
ECLBOD	All 10K ECL dips (10124)
CKECL	All 2080 100K ECL dips (CK124)
MCABD2	All Venus MCA cells (C330)
MOSBOD	All MOS dips (36179)
BlOCK	Block diagram symbols
FlOW	Flow chart symbols
PERT	PERT chart symbols
MECH	Mechanical parts (fan, etc)
UA	Symbols for Unit Assembly drawings
SHAPES	Forms for constructing bodies
MISC	Miscellaneous shapes
MOSCAP	Special MF20 capacitors
GNDCAP	Special KL10 capacitors

To draw circuit schematics the user invariably requires more than one library. Hence for convenience there are top level libraries that contain only pointers to sets of libraries, so that getting a top level library is equivalent to getting all of the libraries in the set. This is the preferred method of accessing standard libraries. Present top level libraries and the libraries they point to are these:

TTL1	PTTL STTL HTTL PARTS NCOMP CMOSBOD
CMOS1	CMOSBOD PARTS NCOMP
ECL1	ECLBOD PARTS NCOMP
MCA2	MCABD2 NCOMP
MOS1	MOSBOD (expansion expected)

Also, MOSCAP and GNDCAP contain pointers to PARTS and NCOMP.

SUDS

OVERVIEW

SUDS OVERVIEW
(Refer to D-FD-SUDS-Ø-OVV1)

Engineering - Is where the design of a P.C. board starts. The Engineer starts by doing rough sketches of circuit schematics. When he has completed the sketches, he submits them to SUDS for processing.

SUDS - The SUDS operator (Engineer, Technician, Drafter) will convert the above sketches to formal drawings, using the SUDS automated drawing program.

Once the drawings have been generated, the operator will then run the wirelist program which will automatically generate the following printouts used first by the designer to check out his design and correct errors and then by CALDEC/IDEA people for reference purposes.

WL - Wirelist....A list of net runs.

WLU - Wirelist Utilization....how components and pins are utilized.

WLS - Wirelist Summary....indicates documentation and logic errors.

PRT - Components Part List....lists all the part numbers and their descriptions that are being used on the P.C. board.

Besides automatically generating drawings and wirelist the program will also create the multitude of files that serve as input to other CAD processes.

Functional Verification - At this stage in the flow the Engineer has a choice of either going the "software" or "hardware" route.

Simulation/Layout Preparation (Software) - There are three programs that an Engineer can run before going to layout.

1. Logic Simulation - Helps the Engineer determine whether the logic on a board or chip really works before going ahead with layout.

2. Placement Optimization Programs - These programs help the designer determine the optimal position of circuit elements from critical parameters supplied by the designer and known characteristics of the materials, including the capacitance of metal runs.

3. Delay Calculation - Allows the designer to determine the physical delays between individual signal points in a circuit design, either a single board or a set of boards in a backplane.

P.C. Design - The use of more up to date interactive Computer Aided Design (CAD) System by P.C. Designers at DEC. A Computer Aided Design (CAD) System is a combination of Hardware (computers) and Software (programs) developed to save designers the time and effort spent on tedious manual tasks. Presently IDEA and CALDEC are the two CAD systems being used by the CAD designers.

Verification - The process by which the designer insures that the drawings and P.C. board are compatible. The P.C. Designer generates files that will automatically update SUDS drawing files. Once the SUDS Designer has completed the update, he generates a file that is submitted to P.C. Designer. The Designer then uses this file to verify that the drawings and the board are compatible. Once the verification has been completed the board is released to production.

SUDS

SUDS
(Refer to D-FD-SUDS-Ø-OVV2)

SUDS is comprised of three basic steps, the SUDS Input, Drawing Program and the Wirelister.

SUDS Input - All projects start with a parts list and sketches. The first thing to do is check to see if all the parts lists are in the SUDS library. If not, submit a SUDSER (form for submitting new parts) to the SUDS library group for implementation of new parts. Then check sketches to make sure that all information to do the drawing is on them.

Once the parts list and sketches have been reviewed then go on to the Drawing Program.

Drawing Program - Is mostly for actually creating drawings, but it also has some features that help in the design process: e.g. it automatically keeps track of signal polarities and nomenclature (I/O pins), and it has commands that allow the user to trace runs. It also checks for polarity and dangle errors.

Closely associated with the Drawing Program is the Plot and Wirelist Programs.

Plot Program - Creates the input file (.DPL) for various hardcopy plotters.

Wirelist Program - Creates the input file (.WD) for the wirelist. It recieves input generated in the Drawing Program and automatically generates wirelist files (WL, WLU, WLS, PRT).

WL - Wirelist
WLU - Pin Utilization
WLS - Wirelist Summary
PRT - Component (parts) List

The **Wirelist Program** has two main functions:

> It is used to create the multitude of files that serve as input to other CAD processes. (See Table 4)
>
> It has many facilities to assist in catching design and documentation errors. These errors will appear in the wirelist files. (See Table 5)

The final stage in the SUDS process (just before release) is the generating of a cap drawing (CAP.DRW) and prefixing all the signal names. There are two programs that will create them automatically, the Cap and Great Name Change programs.

Cap Program - Automatically generates a cap drawing showing all the filter caps that are on a P.C. board.

Great Name Change Program - Automatically assigns prefixes to signal names.

TABLE 4

```
 XHELP
COMMAND ! ********* MEANING *****************
?        TYPE THIS LIST

TOP MODE:
APINRU   WRITE 'PRL' FILE TO SET AUTOMATIC PIN RULES FOR PULLDOWNS
ASIMPL   WRITE 'SMP' FILE, BUT ASSIGN REAL PIN NAMES IN PLACE OF "U" PIN NAME
BCS      OUTPUT 'CS' CIRCUIT SCHEMATIC NETWORK FILE FOR WRLCOM (BACKPANEL)
BFILES   PRINT FILE LIST FOR BACKPANEL
BLCOMP   WRITE 'BLC' FILE COMPARING TO BACK PANEL WIRELISTS
BLIST    BACK PANEL WIRE LIST
BMAKAL   WRITE 'BL', AND 'BLS' FILES
BMULTI   OUTPUT BACKPANEL 'NET' FILE TO MULTIWIRE SYSTEM
BPGMUL   OUTPUT BACKPANEL INNER LAYER INFO AS 'NET' FILE TO MULTIWIRE SYSTEM
BPRINT   PRINT BACK PANEL SIGNAL
BSELVL   WRITE 'BSE' FILE SELECTING ERROR LEVEL ('BS' INFO)
BSIMPL   WRITE 'SMP' FILE INCLUDING ALL PWR PINS
BSSEL    WRITE 'BS' FILE AND SELECT CATEGORY(S)
BSUM     BACK PANEL SUMMARY
CALDEC   OUTPUT CALDEC CONNECTION LIST
CONSUM   MAKE CONNECTORS NEEDED SUMMARY
CS       OUTPUT 'CS' CIRCUIT SCHEMATIC NETWORK FILE FOR WRLCOM (BOARD)
DEFDEV-  SET DEFAULT DEVICE FOR INPUT ONLY
DEFDIR-  SET DEFAULT DIRECTORY FOR INPUT ONLY
DEFPPN-  SET DEFAULT DIRECTORY FOR INPUT ONLY
DIPMUL   OUTPUT ALL DIP LOCS WITH POWER AND GROUND PINS TO 'MPG' FILE
DIPUML   OUTPUT 'UML' FILE TO REDISTRIBUTE REFERENCE DESIGNATORS BY DIPTYPE
DLASAR   OUTPUT 'STG' D-LASAR STIMULUS WIRELIST
DLYLST   OUTPUT 'WL' FILE AND INCLUDE WIRE DELAY FILE DATA
ECO      OUTPUT 'CMD' FILE FOR ADD/DELETES TO TLE
ERRSUM   LOGIC ERROR SUMMARY BACK TO D
FILES    PRINT FILE LIST FOR CARD
FIXCON   OUTPUT LIST OF CONNECTOR PINS FOR UPDATING DRAWINGS
HEAVY-   SET CURRENT AT WHICH "HEAVILY LOADED" OCCURS
IERROR-  ENABLE WIRE LIST INPUT ERRORS TO GO TO FILE
LAYOUT-  ENABLE/DISABLE CHECKING OF LAYOUT DATA
LPART    MAKE PARTS LIST (SAME AS "PART")
MAKALL   WRITE 'WL', 'WLS', AND 'WLU' FILES
MARGIN-  SET MARGIN FOR RUN OVERLOAD ERROR
MERLIN   WRITE 'S2M' FILE FOR INPUT TO MERLIN PLACEMENT SYSTEM
MODEL    WRITE 'MDL' FILE FOR NEWFUN FUNCTIONAL MODEL GENERATION
MPART    MAKE MASTER PARTS LIST (FROM BAC FILES)
MULTIW   OUTPUT BOARD 'NET' FILE TO MULTIWIRE SYSTEM
NETSUM   OUTPUT SUMMARY OF PINS IN NETS TO 'WNS' FILE
OUTSUM   WRITE OUTPUT PIN TERMINATION SUMMARY
PART     MAKE PARTS LIST (SAME AS "LPART")
PPART    MAKE 'T2P' PARTS LIST FILE FOR INPUT TO TXT2P
PRANCE   WRITE 'ASI' FILE FOR ASI'S PRANCE WIRE WRAP SYSTEM
PRECHK-  ENABLE SIGNAL PREFIX CHECKING
PSIMPL   WRITE 'SMP' FILE WITH ONLY POWER PINS IN IT
PGMULT   OUTPUT BOARD INNER LAYER INFO AS 'NET' FILE TO MULTIWIRE SYSTEM
REDAC    OUTPUT REDAC CONNECTION LIST
REFERE-  ENABLE MULTIPLE SIGNAL NAME REFERENCES
SAGE2    OUTPUT 'WIR' FILE OF SINGLE CARD FOR INPUT TO SAGE2 SIMULATOR
SAGALL   OUTPUT 'WIR' FILE OF ALL CARDS FOR INPUT TO SAGE2 SIMULATOR
SIGCHK   CHECK FOR AMBIGUOUS SIGNAL NAMES FOR PARTICULAR COMMAND
SIGSUM   MAKE SIGNAL SUMMARY
SIMPLE   OUTPUT 'SMP' FILE TO DEC WIRELISTER
SIMULA   OUTPUT 'WIR' FILE FOR INPUT TO SAGE2 SIMULATOR
SPFILE   READ 'SFT' SPECIAL FILE TEMPLATE AND GENERATE OUTPUT FILE
SPREFI   OUTPUT SIGNAL LIST FOR "GREAT SIGNAL NAME CHANGE"
SR       OUTPUT 'SR' SCHEMATIC REPRESENTATION FOR MODULE TEST GENERATOR
SSIMPL   WRITE 'SMP' FILE, ASK ABOUT SPECIFIC FEATURES
SSPREF   OUTPUT SIGNAL LIST FOR "GREAT SIGNAL NAME CHANGE", SELECT OPTION
STATUS   PRINT CURRENT STATUS
STUFF    MAKE DIP STUFFING LIST FOR [illegible]
```

```
THPRG   OUTPUT THPR-G CONNECTION FILE
TERMLI  WRITE TERMINATOR TEST POINT LIST
TEST    WRITE CARD TESTER FILE
TLE     OUTPUT 'CMD' FILE FOR INPUT TO TLE
TRMWDC  MAKE 'UML' FILE FOR UPDATING TERMINATOR REFERENCE DESIGNATORS BY COM
NG 2 DRAWING WIRELISTS
TRMWLC  MAKE 'UML' FILE FOR UPDATING TERMINATOR REFERENCE DESIGNATORS BY COM
NG DRAWING WIRELIST TO PC WIRELIST
USAGE   COUNT FREE STORAGE USAGE
USED-   USE D WIRELIST
USEPC-  USE PC WIRELIST
VGSIMP  OUTPUT BOARD TO DEC WIRELISTER
WCAL    OUTPUT 'CAL' FORMAT FILE FOR DLYED AND DLYSRT
WLELVL  WRITE 'WLE' FILE SELECTING ERROR LEVEL ('WLS' INFO)
WLID    CHANGE ID OF CURRENT WIRELIST (RENAME)
WLSSEL  WRITE 'WLS' FILE AND SELECT CATEGORY(S)
WRESIS  OUTPUT FILE FOR RESISTOR DRAWING

TOP MODE OR DIP SUB-MODE:
AVAILA  LIST AVAILABLE NOMENCLATURES
CLEAR   CLEAR CORE EXCEPT FOR RESIDENT DIP DEFS
DDT     CALL DDT
DSKIN   ACCEPT TTY INPUT FROM DISK
DSKHLD  HOLD DISK INPUT(CLOSES FILE)
DSKCON  CONTINUE DISK INPUT(REOPENS FILE, READS TO LAST POSITION)
DSKSKP  SKIP OVER LINES IN DSKIN FILE
ECL-    DEFAULT PIN TYPES TO ECL IN "M<DIP>" COMMAND
HELP    TYPE THIS TABLE
IWRC    INPUT WIRE RULE CHECK DEFINITION FILE
LWIRER  LIST AVAILABLE WIRE RULES
NOMENC- SELECT NOMENCLATURE
PAGE-   SET LINES PER PAGE (FOR LISTINGS)
RESIDE- MAKE DIP DEFS RESIDENT
SAVE    SAVE A DUMP FILE
TTL-    DEFAULT PIN TYPES TO TTL IN "M<DIP>" COMMAND
WIDTH-  SET CHARACTERS PER LINE (FOR LISTINGS)
WIRERU- SET/RESET WIRE RULES
WRCREF  OUTPUT SOME WIRE RULE CHECK DATA IN USER READABLE FORM

DIP SUB-MODE:
IDIP    INPUT 1 DIP DEFINITION FROM DIP DEF FILE
LDIPS   LIST ALL DIPTYPES ON TERMINAL
PACKAG- SET/CLEAR VARIABLE GEOMETRY WIREWRAP PACKAGE TYPE
*
```

TABLE 5

LEVEL 0

- I/O runs that are heavily loaded
- Runs with more than one I/O pin
- Inactive inputs
- Unused extra outputs
- Runs with no unique prefix possible

LEVEL 1

- Runs that are overloaded
- Unused outputs
- Runs with more than one pull-up
- Runs with wire or warnings, all outputs on same DIP
- Runs with wire or warnings, all outputs on different DIPs I/O runs with wire or warnings
- Runs with incorrect or missing signal prefixes
- Runs with no polarity for signal names
- Runs with more than one terminator
- Runs with questionable terminator
- Runs with delay values not in recommended range
- Flip-flop output driving flip-flop inputs of other DIPs

LEVEL 2

- Runs with NO Drive
- Runs with no HIGH drive
- Runs with no LOW drive
- Runs with UN or NC pins
- Runs that need termination
- ECL runs with no terminator rules
- ECL runs that don't match terminator rule
- Runs with pull-down not needed
- Runs with missing pull-down
- Runs with pull-down value not found in delay data
- Runs with pull-down value found but no delay data for fan-in/fan-out
- Runs with more than 2 mA of pull-down
- Runs with more than four outputs
- I/O runs with lower level inputs
- Runs with no inputs or outputs
- Runs with inputs and/or outputs connected to power
- Runs with output connected to ground
- Runs with power connected to ground
- Runs with mixed voltages
- Runs with ECL connected to TTL

LEVEL 3

- Runs with signal property conflicts
- Unnamed I/O runs
- Runs with multiple termination rules

WIRELISTER

THE WIRELISTER PROGRAM

The Wirelister produces material for Wirewrap, for CALDEC or IDEA to generate P.C. board layouts, for the Merlin automatic placement optimization system, and for many other procedures. The Wirelister also has many facilities to assist the user in both drafting and design: in the former, by such things as checking for inconsistent signal names, and in the latter, by keeping track of loading, power consumption, and other physical characteristics. Drawing files of the schematics for a circuit board or chip constitute a permanent database that serves as the central information source for all of these other activities.

Layout - The principle output of the wirelister is the materials needed for input to CALDEC and IDEA. The basic item is a connection file (.CON for CALDEC and .TLE for IDEA), which list all the wiring connections on the entire board (sometimes with power, grounds and terminators, sometimes without, depending upon the stage of the process). This file should usually be on paper tape for CALDEC and on DECTAPE for IDEA, but requirements may vary.

The other materials generated by WL are printouts used first by the designer to check his design and correct errors, and then by CALDEC/IDEA people for reference purposes. These include a wirelist (.WL), a wirelist summary (.WLS), a wirelist pin utilization (.WLU), and a component parts list (.PRT).

With this package, the user should of course supply a set of circuit schematics for the board. He must also prepare a cover sheet (CVR) listing any special directions, such as minimum or maximum lengths for certain runs and whether particular IC's must occupy specific positions on the board. In addition, CALDEC requires a list (CLK) of runs of which clock rules apply (for IDEA, this is taken care of by signal properties in the .TLE file).

The process of getting a layout completed may involve several transfers of material back and forth. If errors are found, the package must be returned to the SUDS user for correction. In any event, a preliminary layout is returned for the user to check that everything is where it should be, that no problems are caused by the way things got laid out, and to assign final component locations.

SUDS itself has a number of facilities for helping with these activities, and there are also associated programs to assist in checking, renaming and comparing. In some cases, the user returns to the drawing program using materials supplied by the wirelister.

When all the problems have been solved, a final routing and clean-up pass through CALDEC/IDEA generates the finished P.C. board layout.

For an ECO, the wirelister supplies a list of the changes (a wirelister compare (.WDC) file for CALDEC, and add/delete (.TLE) file for IDEA.

IDEA can be used for designing individual MCA chips (it is more advanced than CALDEC and can handle many more layers). Then with the input package limited to the external chip connections, CALDEC or IDEA can be used for laying out a P.C. board containing the chips.

There are three programs that the wirelister will generate input to, which can reduce time and cost of a design. They are the Logic Simulation, Placement Optimization, and Delay Calculation Programs.

Logic Simulation - (see D-FD-SUDS-Ø-oVV3) One of the products of the wirelister is a WIR. This is the input to the SAGE 2 logic simulator, which helps the user determine whether the logic on a board or chip really works before going ahead with layout. SAGE 2 simulates the hardware of individual gates and logic subsystems with the ability to inspect the interaction between individual gates in real time, to determine whether logic actually does what it was designed to do. An entire system can also be simulated, with inspection at levels higher than individual gates. A similar wirelister output (the NWS file) is used by the VOTE simulator, which tests the effectiveness of diagnostic test patterns, programs and microcode. In the next generation, both simulators will use the same input.

Placement Optimization - (see D-FD-SUDS-Ø-OVV4) Since both CALDEC and IDEA use placement procedures that are particularly automated, layout can be expedited by making prior use of the Merlin automatic placement optimization system. These programs help the designer determine the optimal position of circuit elements from critical parameters supplied by the designer and known characteristics of the materials, including even the capacitance of metal runs.

The original program, MINCUT, was specifically for placing the 400-gate-array chips used in the Comet project. The new program, FINCUT, is much more general in that it is not limited to the physical and logical characteristics of any particular technology. Instead, for each placement run, it gets the appropriate technological specifications from a pair of library files: the grid file (.GRD), which specifies the positions where packages can be placed on the physical unit, and the logical packages group file (.LPG), which defines the characteristics of the components.

FINCUT exists in several versions geared to specific uses, such as MCACUT for MCAS and ECLCUT for PC and multiwire boards with ECL parts.

The wirelister supplies the necessary information about the design to Merlin in the S2M file, and Merlin returns the optimal dip and connector pin positions in the M2S file. With this information, the designer can go back to SAGE 2 to get real delays and begin a more confident layout.

Delay Calculation - (see D-FD-SUDS-0-OVV5) This software package allows the designer to determine the physical delays between individual signal points in a circuit design, either a single or a set of boards in a back plane.

From the SUDS wirelist file and the CALDEC/IDEA output, DLY creates a database that represents the physical hardware as it would be built, and from that CAL calculates all signal propagation times taking into account gate delays, wire links, and even stubs. Then with DLYED, the user can determine the delay structure of his design by inspection of propagation times across individual elements in each signal path.

For MCA inspection, the CAL file, which is equivalent to the output of the CAL program, can be generated directly by the wirelister.

The wirelister also generates files that input to programs such as MULTIWIRE, WIREWRAP, and KPL.

MULTIWIRE - (see D-FD-SUDS-0-OVV6) Utilizing the P.C. program, parts of the drawing program, multiwire software (the MDT package), outputs of the wirelister (NET and MPG files), and the routing rules built into the module wirewrap program (see below) or FINCUT, the user can layout his own multiwire boards and generate the materials for the multiwire vendor to produce it. Sometimes these serve as prototypes while one waits for the CALDEC/IDEA process, but where expected volume is small, they may be used for the finished product.

Designing an etch board is much more expensive and time consuming than multiwire, but etch boards are cheaper to manufacture. Thus for only a few dozen units, it is probably better to use multiwire, whereas for volume production, it pays off to layout etch boards.

WIREWRAP - (see D-FD-SUDS-Ø-OVV6) Another product of the WL program is a SMP file, which serves as input to the module wirewrap system. These programs have built-in routing rules and make use of three major inputs: a placement file (PLC) that defies the coordinates of pin 1 of each dip position, a package file (PKG) that gives the coordinates of all other pins relative to pin 1 for each package type, and a WL-produced assignment file (ASG) that tells what style package occupies each position.

A similar system using the SMP files for all of the boards, but with internal connections deleted (i.e., containing only I/O pins) is employed for laying out the background wirewrap.

KPL - (see D-FD-SUDS-Ø-OVV6) The parts information on the SUDS database (.PRT) is used as input to the KPL program. The benefits of using this program are:

A. Eliminates manual input to KPL (human errors).

B. Minimizes document errors (part number, quantity and reference designator) on KPL.

C. Reduced time and cost in generating a KPL.

D. Information sent to the KPL will be consistent.

SUDS PROCESSES

SUDS TO IDEA

SUDS TO CALDEC

SUDS TO IDEA/CALDEC

The wirelister generates the materials need for input to CALDEC and IDEA. The basic item is a connection file (CON for CALDEC, TLE for IDEA), which list all the wiring connections on the entire board (sometimes with power, grounds and terminators, sometimes without, depending on the stage of the process). This file should usually be on paper tape for CALDEC and on dectape for IDEA, but requirements may vary.

SUDS TO IDEA (see D-FD-SUDS-Ø-PRC1) - The wirelister generates the TLE (.CMD) file that inputs into IDEA. The P.C. designer lays out the board and assigns pin numbers and reference designations (which may already have been done by the Engineer). In the layout process the Engineer may submit design changes which will extend the completion date of the board.

Upon completion of the board the P.C. designer will generate a UML file that will contain all his layout changes. He will submit this file to SUDS. The SUDS operator will process the UML file with the .DRW files (circuit schematics). This process automatically updates the SUDS.DRW files. Once the SUDS operator has completed the above, he will process the .DRW files through the drawing and wirelist programs.

The wirelister is then used to generate a .CS file (circuit schematic representation) that is then submitted to IDEA. The P.C. designer will use the .CS file to varify that the SUDS drawings and board are compatable. If they are not, the above process has to be repeated. If they are compatable, then the board is submitted to production.

SUDS TO CALDEC (see D-FD-SUDS-Ø-PRC2) - Is the same process as SUDS to IDEA except for the UML process. At that stage the P.C. desginer generates an RL8AH tape. The tape contains the old file (original file submitted to designer) and new file (file with designer's changes). The tape is submitted to SUDS. It is then processed using the RENAME program which automatically updates all the .DRW files. The drawings and board varification process is the same as SUDS to IDEA.

SUDS PROCESS
(See D-FD-SUDS-Ø-PRC3)

OBTAIN PPN - A PPN number can be obtained by filling out a CADNET operations access form and submitting it to computer operations.

REVIEW PARTS LIST - Review Parts List to insure that all information on the parts list is contained in the SUDS Library.

SUDS LIBRARY - Contains the defined graphics symbols and the electrical characteristics of components.

RUN GCD/GTD - Puts you into the SUDS drawing program.

CREATE .DRW FILE - The .DRW file represents a drawing (circuit schematic, flow and block diagrams, floor plans, etc) which is generated by calling bodies from the SUDS Library.

DANGLE POINTS - (.XDA) Checks for two types of dangle points.
a) A stand alone point, which has no logical function - it just clutters up the .DRW file.

b) A line that has a missing signal name will have a dangle point error message.

POLARITY CHECK (XPOLCK)- Checks for polarity errors. This command checks the .DRW file for polarity errors such as:

a) A high signal connected to a low signal

b) A signal name that has the high indicated in the name but connected to a low input/output of a body or in reverse.

PLOT XEP (.DPL)- Is the process by which the SUDS designer generates a .DPL file, which is used as the input into the "P" (plot) program.

WIRING DIAGRAM XEL (.WD)- Is the process by which the SUDS designer generates a .WD fiel, which is used as an input into the wirelister program.

WIRELISTER (WL)- A set of SUDS wirelist files (WD) is the source of information for the wirelister program (WL), which from it generates all sorts of option files for other programs.

XMAKALL - Is the wirelister command that will automatically generate the following files and listings:

Wirelist	(.WL)
Wirelist Summary	(.WLS)
Wirelist Utilization	(.WLU)
Parts List	(.PRT)

OPTION FILES - SUDS output files that are used for input to other programs.

XTLE - Generates a .CMD file for input to IDEA

XCALDEC - Generates a .CON file for input to CALDEC

XCS - Generates a .CS file for input to IDEA and CALDEC to do a wirelist compare (WLCOMP) to insure that the drawings and PC board are compatible.

SAGE 2 - Generates a .WIR file for input to the SAGE 2 logic simulator.

RENAME PROGRAM - The rename program receives an RL8AH tape from CALDEC, which contains an old and new file. These two files are processed using the rename program to generate an .UML file, which is used as an input to the GCD/GTD program to automatically update the .DRW file.

PLOT PROGRAM (RU P) - Running the plot program "P" on a set of SUDS plot files (.DPL) results in a .PLO file, which serves as an input to most of the different hardcopy units (Versatic, Varian, and CAL-COMP) to produce a plot of the finish drawing.

GREAT NAME CHANGE - Is used to put prefixes on all signal names. It generates a change file (.CHG) that is used to input to GCD/GTD programs.

CAP PROGRAM - Is a program that automatically generates a cap drawing (CAP.DRW) showing all filter caps that are on the board.

Conclusion

The days of the Engineer designing with rough sketches of circuit schematics, changing them frequently and scribbling directions to the drafter all over them are numbered.

An automated design system eliminates much of the:

-Effort required to produce a finished drawing.

-Sources of error in communication between Engineer, P.C. designer and drafter/technician.

-Drafter's/Technician's erasing or sticking an added circuit element way off at the edge, disconnected from its associated logic. Instead he just changes the file and plots anew.

-Inconsistencies in drawings (symbols, lettering, and line thickness).

To the user the most significant factor is that the products of the system are data files that can be handled by a computer; every hardcopy of a drawing is simply a copy - there is no original paper. This means much shorter turnaround time both for development changes and later ECO's. It means much greater accuracy, specially for repetitive operations and common elements used in a number of situations.

The advantages of SUDS over manual drafting are:

-Much less drafting is required to translate a sketch into an initial drawing. Manually it would take ten to twelve hours to do a drawing vs six to eight hours if done on SUDS.

-It automatically keeps track of signal polarities and nomenclature (I/O pins). Manually it would be very time consuming and costly to do the above.

-It allows the user to automatically trace runs. The drafter/technician would have to physically trace runs.

-Has many facilities to assist in catching design and documentation errors.

-Has many facilities to assist in catching design and documentation errors.

- Manually it would involve the Engineer and drafter/technician to manually check drawings and design to catch the above errors, where as SUDS will do it automatically. That leaves the Engineer and drafter/technician time to do more productive work. The major advantage of an automated system is not so much a saving in total drafting as an ability to do a great deal more useful work in the time available: just having more information on the circuit schematics makes them more valuable to Field Service, Manufacturing, and other users. In many cases using SUDS allows activities that would otherwise be impossible. Suppose for example that as a production deadline approaches, it is decided the signal prefix used on a group of a dozen drawings is misleading in light of the way the design has evolved. A manual drafting department would simply inform Engineering that nothing could be done about it. But an automated department would have just SUDS search through the files for all instances of the prefix and change them. Some Engineers have become adept enough at using the system so that they even make minor revisions (such as IC substitutions) at an office terminal with no graphic display at all.

I hope the SUDS Overview has enlightened you to what SUDS is and what it can do for the Engineering and Engineering Services Community, in reducing design time and cost.

Please feel free to call me, Norm Rheault, Ext. 223-8789, ML4-2/E90 for more information.

WHAT IS THE SUGGESTED MODULE DESIGN PROCESS USING SUDS?

A. THE BEST TIME TO CAPTURE THE LOGIC INFORMATION (SCHEMATIC) IS DURING IT'S CREATION. AT THIS TIME, SUDS CAN BE OPERATED BY THE ENGINEER OR TECHNICIAN.

B. WHEN SCHEMATICS HAVE BEEN DEVELOPED TO THE BREADBOARD STAGE, A WIREWRAP BOARD CAN BE PRODUCED IN A MATTER OF DAYS OR A "QUICK AND DIRTY" P.C. LAYOUT IN WEEKS... BOTH USING THE DATA BASE.

C. DURING DEBUG, GROUPS OF CHANGES ARE ENTERED TO THE ON-LINE SUDS DATA BASE TO ASSURE THE DATA BASE ALWAYS MATCHES THE BREADBOARD.

D. THE FINAL DATA BASE FROM THE DEBUGGED SCHEMATICS IS SENT DIRECTLY TO P.C. LAYOUT WITH NO HAND CODING. TWO VERSIONS OF THE DATA BASE CAN BE COMPARED AUTOMATICALLY AND A WIRELIST OF THE CHANGES REPORTED.

E. THE MIF CAN BE COMPARED TO THE WIRELIST FROM THE ENGINEER'S PRINT SET ASSURING A MATCH.