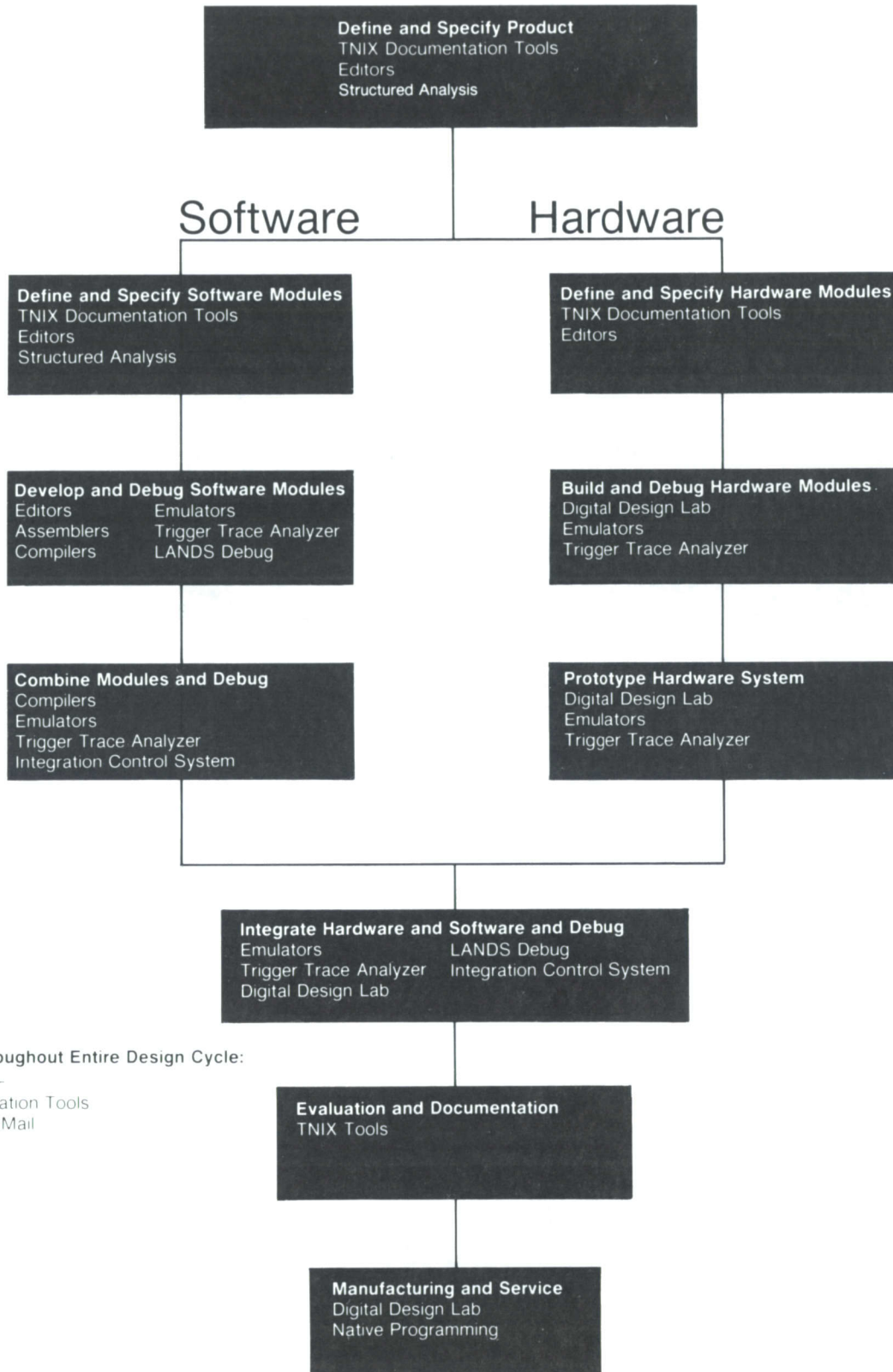


Microcomputer Development Cycle



Used Throughout Entire Design Cycle:
ColorKey +
Documentation Tools
Electronic Mail

Tek Tools Help You be More Productive Throughout the Entire Microcomputer Design Cycle

Over time, microcomputer designers have developed a systematic microcomputer design cycle which guides a microprocessor-based product from concept and definition through to manufacture and service. Each step in the cycle presents a unique set of problems and challenges to the engineer. Also, each step is critical to building a successful product.

Every phase in the cycle presents you with different demands. These demands are growing more complex as microcomputer hardware moves from the 8-bit to the 16-bit level and beyond. The only practical solution is more powerful design tools.

Tek's 8500 Series Automates The Design Cycle.

Only the Tektronix 8500 Series can give you design support broad enough to embrace the entire design cycle. No matter where you are in the cycle, Tektronix has design automation tools that dramatically increase the effectiveness of the engineer in developing a better product in less time and at a substantially lower cost.

Tek's 8500 Series design tools cover more design cycle tasks with greater effectiveness than any other engineering tool set on the market today. And they do so with a level of quality that has made Tektronix the leader in digital design instrumentation.

MICROCOMPUTER DEVELOPMENT

The ColorKey+ User Interface
 is divided into 4 scripts
 shown below. Transition time
 between scripts is about 10 sec.

COLORKEY+

<u>analysis</u>	<u>design</u>	<u>edit</u>	<u>code</u>	<u>debug</u>	<u>files</u>	<u>sys.ops</u>
<u>edit</u>		<u>ed</u>	<u>assemble</u>	<u>compile</u>	<u>listfiles</u>	<u>sys.info</u>
<u>evaluate</u>		<u>ace</u>	<u>asm</u>	<u>compile</u>	<u>showfile</u>	<u>mail</u>
<u>fix</u>		<u>lde</u>	<u>link</u>	<u>icsprompt</u>	<u>file info</u>	
<u>show</u>	<u>crt</u>	<u>format</u>	<u>libgen</u>	<u>ics</u>	<u>file mgmt</u>	
<u>etc.</u>	<u>copier</u>				<u>flex disk</u>	
<u>listfiles</u>	<u>plotter</u>				<u>dir mgmt</u>	
<u>add TBD</u>						
<u>list data</u>						
<u>list proc</u>						
<u>look DD</u>	<u>configure</u>	<u>asm debug</u>	<u>TTA debug</u>	<u>HLL debug</u>	<u>DDL debug</u>	<u>prom prog</u>
<u>sort DD</u>	<u>new setup</u>	<u>status</u>	<u>status</u>	<u>pdb</u>	<u>setup</u>	<u>setup</u>
<u>overview</u>	<u>select</u>	<u>run pgrm</u>	<u>bkpt</u>		<u>DDL cntrl</u>	<u>read</u>
<u>listfiles</u>	<u>emu mode</u>	<u>memory</u>	<u>view acq</u>		<u>DAS cntrl</u>	<u>program</u>
	<u>memory</u>	<u>registers</u>	<u>trace prg</u>		<u>diags</u>	<u>compare</u>
	<u>load</u>	<u>load/save</u>	<u>counting</u>		<u>send cmd</u>	<u>status</u>
			<u>run code</u>			<u>listproms</u>

Tek Microcomputer Development Systems Expand with Your Needs

No matter what the size and scope of your engineering operations, Tek provides a solid hardware and software foundation for your microcomputer design tool set.

The 856X Family consists of the 8561, 8560, and the 8562

The 856X Series is designed to be 100% software compatible and hardware upgradable. Each member offers a specific combination of features with reasonable price/performance.

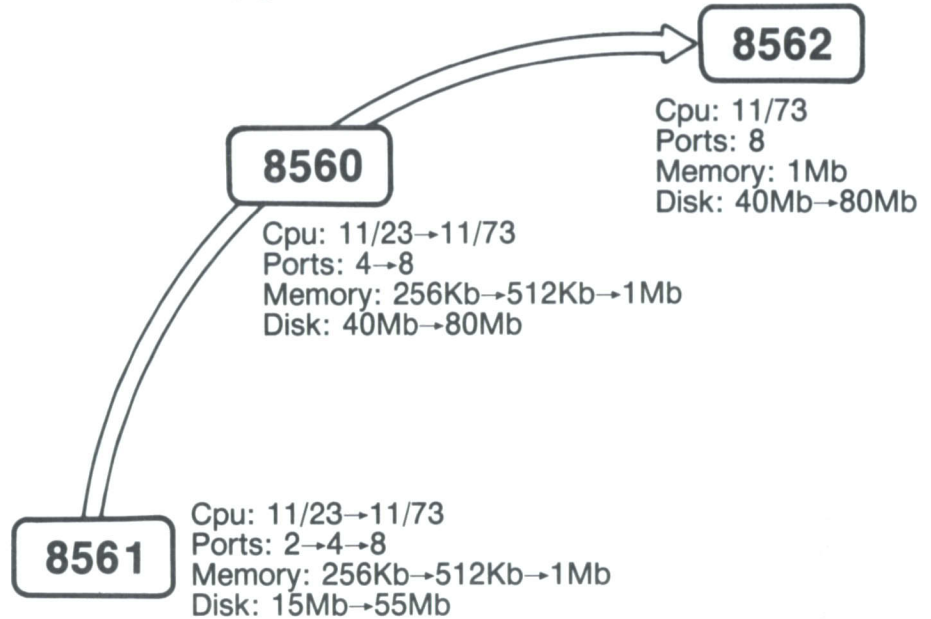
8561

The 8561 is a powerful, low cost entry-level software development system supporting one or two users. It provides a complete set of microcomputer design tools to the smaller design team while accommodating future expansion through a simple, cost-effective upgrade path.

The 8561 offers an LSI-11/23 CPU, a 15 Mbyte hard disk, a 1 Mbyte flexible disk, 256 kbyte of RAM memory, two user ports, and two line printer ports. This basic system can be easily upgraded within the same mainframe to an LSI-11/73 CPU, 55 Mbyte of hard disk, 512 kbyte or 1 Mbyte of RAM memory, a GPIB interface for disk backup, and four or eight user ports. All upgrades can be implemented through any Tektronix Service Center, and all Tektronix software development and integration tools remain fully functional after the upgrade.

Each 8561 user port can support either a standard CRT terminal, a Tektronix 4105 Color Terminal, or a Tektronix 8540 Integration Unit designed specifically to handle hardware/software integration tasks through realtime emulation.

856X Series Upgradability



8560

The 8560 offers a medium cost/performance software development system for four users. The same software design tools offered for the 8561 are available on the 8560 and the 8560 is also upgradable. A standard 8560 offers an LSI-11/23 CPU, a 40 Mbyte hard disk, a 1 Mbyte flexible disk, 256 kbyte of RAM memory, four user ports, and two line printer ports.

The 8560 can be easily upgraded within the same mainframe to an LSI-11/73 CPU, 80 Mbyte of hard disk, 512 k or 1 Mbyte of RAM memory, a GPIB interface for disk backup, and 8 user ports. Upgrades are handled similar to the 8561. The 8560's user ports, like the 8561, support standard CRT terminals, the 4105, and the 8540.

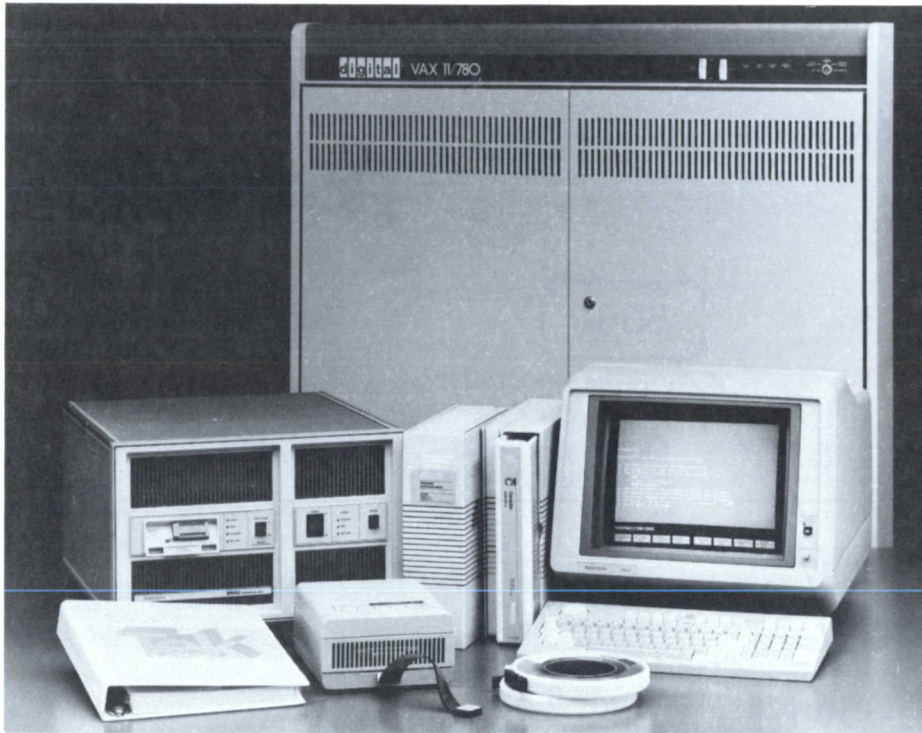
NEW 8562

The 8562 is a high performance software development system which supports up to eight users. Compatible with the 8561 and 8560, the 8562 offers an LSI-11/73 CPU, 40 Mbyte hard disk, a 1 Mbyte flexible disk, 1 Mbyte of RAM memory, eight user ports, and two line printer ports. The 8562 can be upgraded with a GPIB interface for disk backup, or 80 Mbytes of disk storage within the same mainframe.

For large design teams, the 8562 offers a complete solution, at a very good price/performance ratio.

NEW

MICROCOMPUTER DEVELOPMENT



TNIX Operating System

The TNIX Operating System present on all members of the 856X Series is derived from AT&T Bell Laboratories UNIX Operating System Version 7. TNIX is ideally suited to the team-oriented software design environment and contains many features targeted specifically at improving the productivity of individual engineers participating in a collective design effort.

Multi-Tasking

TNIX increases user productivity by allowing several tasks to be performed simultaneously. For example, a source code file could be assembled or compiled while another is being edited. Or a file could be printed while the user views a document, etc.

Hierarchical File System

TNIX provides a powerful hierarchical file system, where files can be arranged in a "tree" structure with many levels. Each level accommodates either files or file directories pointing to more files or directories at deeper levels. This file organization has several benefits. Files can be organized by project or function, which makes them easier to find and manipulate. Sublevel files can be easily protected by restricting access to top level directories. TNIX allows read, write and execute permissions to be assigned to files belonging to the owner, members of the group or all others.

User can also be assured of using the latest version of software by "sharing" (linking) to the master file. As the master file changes, all linked copies change also. Linked copies of the master file do not take up file space, another advantage.

Electronic Mail

TNIX Electronic Mail provides a very easy way to transmit messages between individual users. Messages can be sent or received 24 hours a

day, allowing users to communicate effectively. Mail can be sent between users on the same 856X or another 856X networked to it. Users are notified at login when mail has been sent to them.

When two or more users are actively using an 856X, they can send messages directly to each other's screen without leaving their desk. Also, the 856X system manager can "broadcast" a message to all active users at the same time to relay important information.

Automated Software Creation

To simplify generation of large programs composed of many modules, TNIX features a tool called "make" which ensures that all interdependent source code modules have been reassembled or recompiled to reflect updates entered after debugging. Make minimizes recompilation by regenerating only those modules which have changed since the last build. It also eliminates build errors and provides a single, linked, executable object module with one command. Make also reduces support costs by documenting the build process, and can be used with assemblers, compilers and text processing tools.

8540 Integration Unit Support

The 856X Series is designed for easy interfacing to the Tektronix 8540 Integration Unit, which provides real-time emulation for the entire range of Tektronix 8-bit and 16-bit chip support. The entire 8540 command set is built into TNIX, allowing 8540 and 856X commands to be intermixed and invoked from a single workstation. Code developed on an 856X is downloaded to the 8540's program memory, up to 256 kbaud, for execution on the emulator processor. Transfers occur at 153.6 kbaud using the high speed interface between the 8540 and the 856X Series.

Code execution takes place under control of debug software, and the resulting data can be uploaded and analyzed using powerful data manipulation tools available in TNIX. This provides the H/W engineer with the UNIX productivity tools previously available only to S/W engineers.

856X Software Support for Microprocessor Design

The 856X Series supports a wide variety of software development and integration tools for over thirty of the most commonly used 8-bit and 16-bit microprocessors. Software available ranges from assemblers to high level support such as Pascal LANDS, the first Pascal microcomputer software development package that supports all phases of the design cycle.

At the source code entry stage, LANDS includes an easy to use Language-Directed Editor which understands Pascal syntax. As source code is entered, syntax errors are immediately identified so the user can easily correct them using the screen editing capability. LDE minimizes the edit/compile/edit cycle time associated with syntax errors, which are most common.

The LANDS Pascal compiler contains many enhancements aimed specifically at microcomputer programming. This includes features such as bit level data manipulation, assignment of variables to specific hardware addresses and direct access to I/O ports. Also, a separate optimizer pass can significantly reduce code volumes and boost performance.

LANDS also includes a unique tool for implementing the hardware/software interface, called the Integration Control System. The user simply responds to a menu or creates a brief file outlining parameters for the software/hardware interface. ICS automatically handles the details, such as creating linker command files, interrupt handling code, and reset/initialization code.

For hardware/software integration tasks, LANDS includes Pascal Debug, allowing all debug operations to be performed at the Pascal source code level. This completely eliminates the time-consuming requirements of translating assembly-level debug information into its high level counterpart.

Tek/DEC Software Compatibility

Tek also offers powerful High Level Language tools and many popular assemblers on Digital Equipment Corporation's VAX Series of computers using either the VMS or UNIX based operating systems. This Tek/DEC combination gives you a powerful series of options when configuring your design environment. You can run Tek software development tools on the VAX to produce executable object code, and easily download the code to a Tek 8540 Integration Unit to perform emulation/debug tasks. Tek supplies you with the software needed to make your VAX/8540 combination function as a turn key system. In more advanced configurations, you can interface entire 8560/8540 systems to the VAX to create a powerful distributed processing system with the VAX acting as the central manager.

Regardless if you start with the entry level 8561 or a VAX 780, the Tek software tools will allow you to transport the programs you generate from one computer to another.

Single Key Interactive Color Interface to Tek Design Tools

Colorkey+ and the Tek 4105 Color Computer Graphics terminal combine to give you simple, single key stroke access to the powerful TNIX operating system and Tek's microcomputer design tools.

The color coding and graphically defined "soft" keys guide you through the system with a minimum of effort or knowledge.

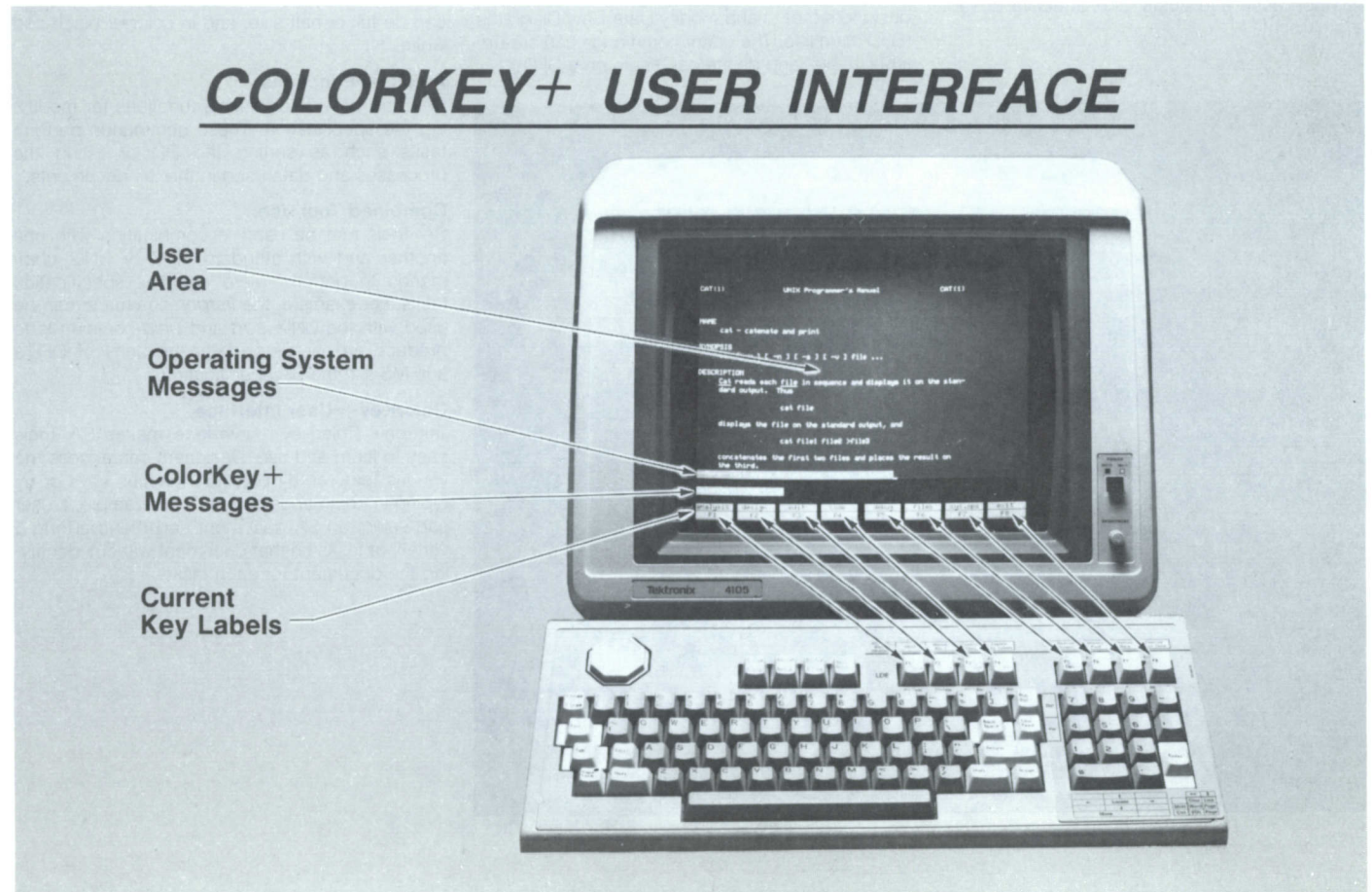
The use of color and softkeys minimize the time it takes new users to learn TNIX and use the emulation/debugging tools provided by the 8540 Integration Unit. Softkeys simplify command entry and minimize keystrokes. Softkey labels appear on the bottom of the screen and dynamically change as different operations are performed. Functions are arranged in a hierarchical manner, from major tools down to specific options for a selected command.

ColorKey+ is an intelligent interface. Parameters entered by the user are remembered and displayed as softkey options later whenever necessary. Previous commands can also be recalled, modified, and reissued. If a user requires more information about a softkey option, ColorKey+ will offer a longer explanation. And, if users "forget" where they are in the softkey command "tree" a "Where-AM-I" command graphically displays the command sequence and softkey levels.

Furthermore, as softkey commands are entered, ColorKey+ displays the actual TNIX commands required to perform the operation. Eventually, the user learns most of the system commands and if desired, can directly issue commands to the system, bypassing ColorKey+.

Color is used by ColorKey+ to reduce user fatigue and to increase productivity. It maximizes readability of complex information. Different types of information being displayed can be highlighted using color. As an example, ColorKey+ uses red to display messages which require immediate attention, and brown for prompting the user.

Colorkey+ User Interface



Tek Structured Analysis Tools Specify Conceptual Requirements Graphically

Tektronix Structured Analysis (SA) Tools, a set of graphically oriented, software development facilities, automate the method of specifying system requirements. Used in the first stage of a design cycle, they assure completeness and consistency in a specification.

SA methodology is based on a systematic, graphic, top-down technique of describing what a system does. Using SA, designers can develop an accurate and complete system specification before design begins.

The most important benefit of using SA methodology is an improved understanding of the system at an early stage in the project. This is largely due to SA's graphic notation which encourages feedback and thorough analysis. Without SA tools, manually maintaining the mechanical details of the methodology are tedious and error-prone.

SA Tools automate those functions that a computer can do best—editing the specifications and diagrams, detecting omissions in SA documents, maintaining consistency, and displaying the documents on screen and in hardcopy. While the computer handles the routine details, the designer is free to concentrate on system analysis. This results in improved product quality, reduced development time and costs, and effective project management.

SA Tools Functions

SA Tools support the SA methodology by automating the routine tasks of specifying a system. They let designers create, analyze, modify, and display a specification of the system to be developed.

COMMAND	FUNCTION
Main	
edit	Edits an SA document.
evalsa	Evaluates an SA document.
fixsa	Corrects an SA document.
showsa	Shows an SA document.
Auxiliary	
addtbd	Adds new entries to the DD.
dfdtplot	Converts a DFD to the UNIX plot format.
listdf	Lists data flow names from SA documents.
listpnn	Lists process numbers and names from SA documents.
lookdd	Looks up an entry in the DD.
mono	Produces a monochrome copy of a DFD.
sortdd	Sorts DD entries alphabetically.

Editing

A special-purpose interactive graphics editor lets designers create and modify Data Flow Diagrams (DFD's) on-line. The editor commands can create, label, move, and delete each item on a DFD.

Evaluation

The evaluation function verifies consistency and completeness in the SA specification. The *evalsa* command analyzes the SA documents both individually and with respect to one another. It identifies inconsistencies and omissions in documents. It also evaluates a document for adherence to SA guidelines.

For example, the *evalsa* command checks consistency of data flows between a DFD and its parent, checks that the Data Dictionary (DD) defines all data used in DFD's and no other data, and checks DFD's for unconnected and read- or write-only data items.

Correction

The correction function preserves consistency in the SA specifications as it is developed. The *fixsa* command automatically maintains consistency of process names and numbers in the specification. This command creates new documents with the proper name and number, and corrects documents with incorrect names and numbers. It also maintains consistency between DD entries and the data names in DFD's by adding or deleting DD entries to match new or changed DFD's.

Display

The display function is used to show the SA specification on-line or to produce a hard copy. The *showsa* command displays formatted SA documents on the terminal.

Designers can also produce DFD hard copies quickly on Tektronix copiers and plotters. Copies can be full or half size, and in color or black and white.

Auxiliary Functions

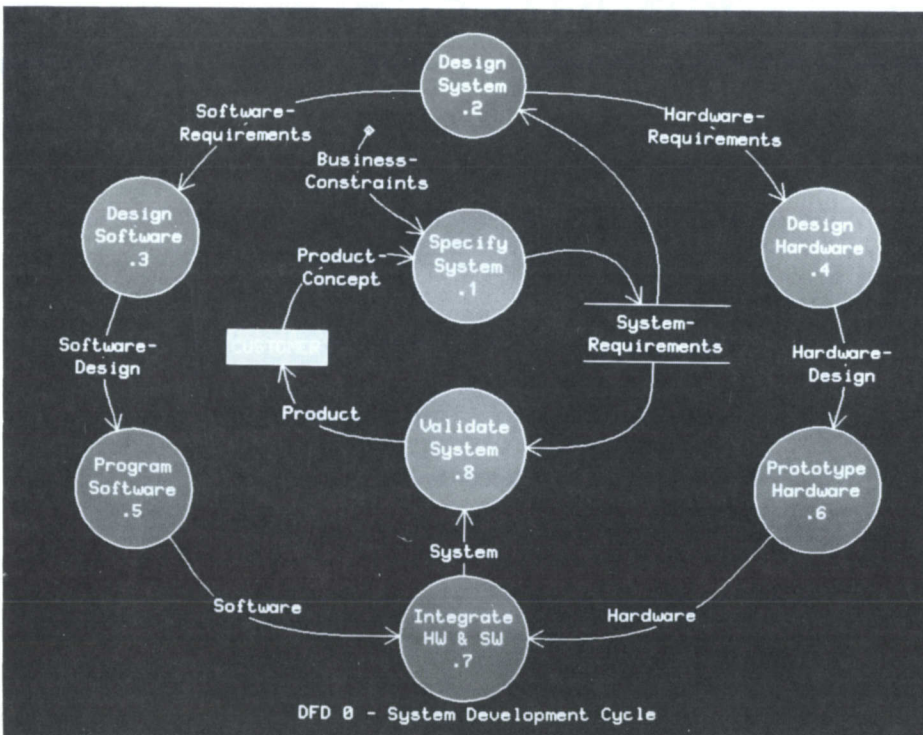
SA Tools provide additional functions for modifying the specification. These commands perform tasks such as sorting the DD or listing the processes and data used in the SA documents.

Combined Tool Use.

SA Tools can be used in combination with one another and with standard TNIX or UNIX commands to perform more complex specification tasks. For example, the *listpnn* command can be used with the UNIX *sort* and *uniq* commands to produce an overview of the hierarchy of DFD's and MS's in the SA specification.

ColorKey+ User Interface

Tektronix ColorKey+ interface makes SA Tools easy to learn and use. Designers can access the various features by pressing function keys or by typing in commands directly. For example, a user can select an SA document and then perform a variety of tasks on that document without identifying the document for each task.



Tek Editors Expedite Code and Text Entry

During software development and document preparation, high performance editors become critical tools in the design effort. Different tasks within the design cycle can be made more productive by using an editor specifically designed for the task. Also individual users have different preferences as to the type of editor they like to use. Tek offers a selection of editors to handle the different tasks and individual preferences.

General Purpose Editors

The TNIX line editor, "ed", lets you make sweeping changes in your code or documentation with a minimum of effort. With the streamlined command set of "ed", updates are easily accomplished.

The Tek Advanced Screen-Oriented Editor (ACE) is an ideal choice for text entry and cursor-directed screen editing at specific locations within your code or text.

High-Level Source Code Editors

Tek's Language-Directed Editor (LDE) is an integral part of the Tek LANGUAGE-oriented Development System (LANDS). LANDS allows editing, compiling, integrating, and debugging a

program in the same high level for the entire cycle. The LANDS package consists of LDE for PASCAL or "C", Compiler, Integration Control System, and Debug, thus allowing the user to work through the S/W design and debug cycle while remaining at a high level.

LDE is actually able to understand the syntax of the Pascal or C language. Thus, it will flag any syntax errors in the source code you've entered while you are still in the editor. LDE's syntax error detection eliminates the need to run lengthy compilations just to locate syntax errors, LDE is also an excellent general purpose screen-oriented editor.

Special Editing Tools

The TNIX Auxiliary Utilities Package contains several specialized tools that allows sweeping changes to multiple files automatically. For example, if you have several assembler source files and you wish to change all the assembler directives, you could use these tools to accomplish this task automatically.

Tek Gives You the Choice of Editor That Will Best Accomplish the Task You Wish to Perform.

C LDE

Checks Syntax Of Code During Edit Session

```
# include <stdio.h>
# define FALSE 0
# define TRUE 1
pedestrian ( )
{
    char ped;

    printf ("Press space bar to cross street.");
    scanf ("%c", & ped);

    if (ped == ' ')
        return — (TRUE);
    else
        return (FALSE);
}
```

Syntax error: detected at 'return'

Tek Compilers and Assemblers

Microcomputer software design demands a highly sophisticated set of programming tools that maintain high productivity while permitting logic manipulations all the way down to the hardware level. In response, Tek compilers and assemblers bring you an advanced feature set that streamlines your coding effort while retaining the power you need to implement machine-level operations.

High-Level Amenities for Assembly Coding

With Tektronix assembler/linkers, you get many features that are normally only associated with high-level coding. You can create sophisticated macro statements that provide high-level coding power.

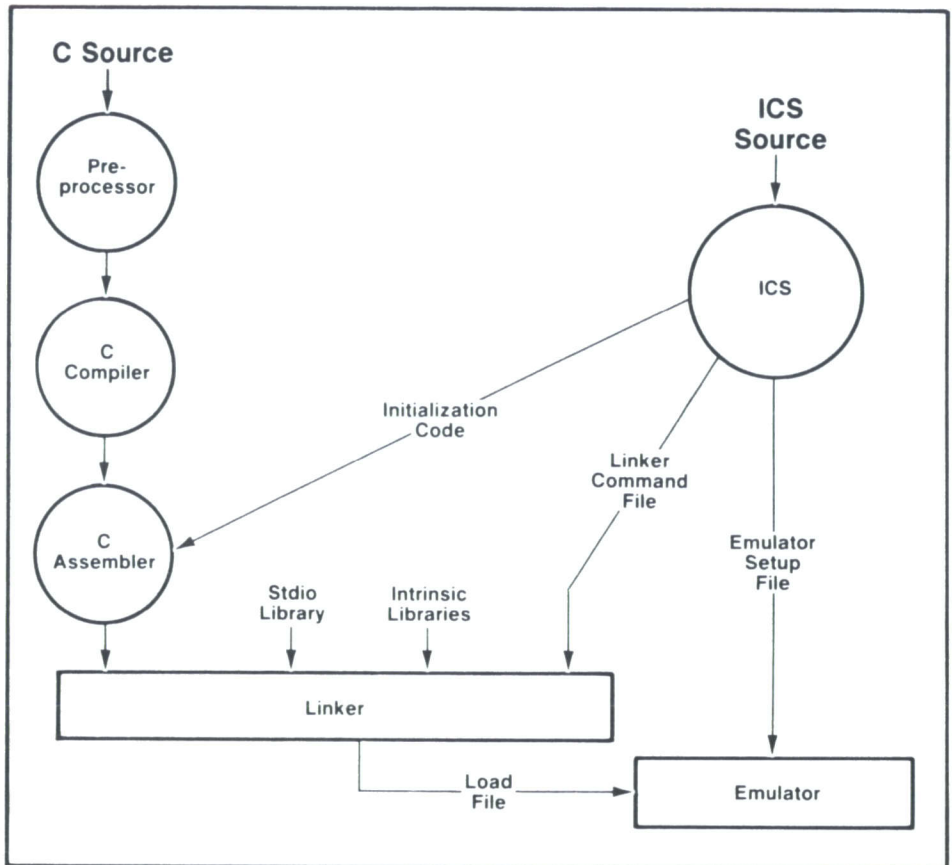
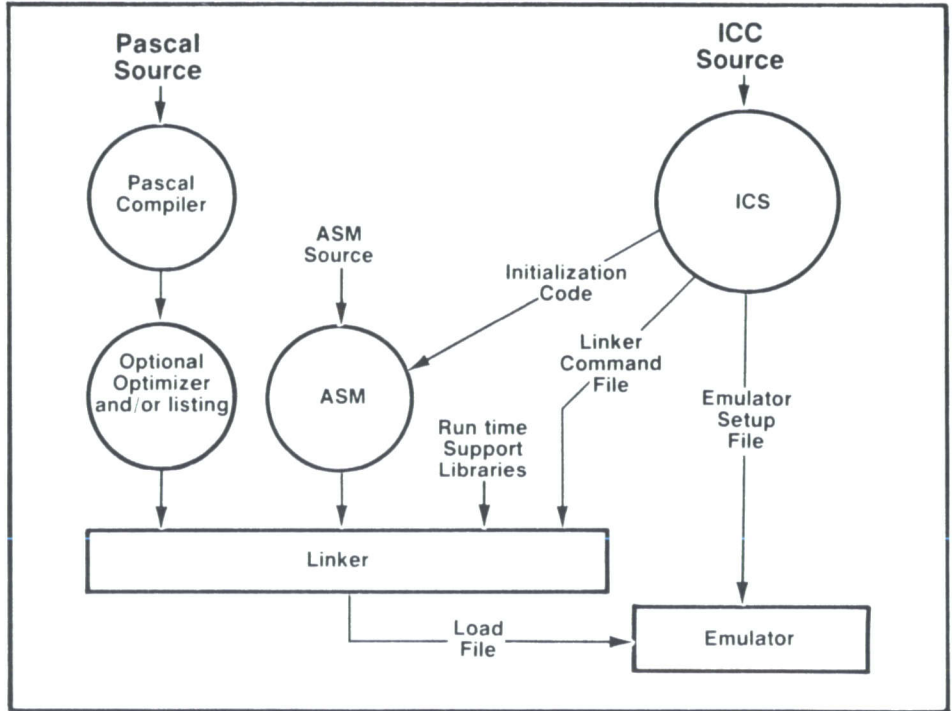
The INCLUDE directive can be used to include other files containing assembler source, data types, constants and variables. Conditionals, using Boolean expressions, are available to help you control the assembly process. In addition,

Tek assemblers all share the same base, which means once you learn a Tek assembler you can move from one microprocessor to another with a minimum of learning time. All the MACRO commands, expression handling and assembler directives are the same.

Tek Compilers Offer More Than Generation of Quality Code

For high-level language programming, Tek's PASCAL and "C" compilers give you all the established benefits of structured languages plus the ability to do operations such as interrupt handling, I/O port access and bit manipulation without ever having to resort to assembly language code.

To insure compatibility for porting existing programs, Tek's PASCAL-Compiler adheres to the ISO standard and Tek's "C" Compiler adheres to Kernighan and Ritchie's standard.



MICROCOMPUTER DEVELOPMENT

LANDS Automates Hardware/Software Interface Definition with ICS

One major task associated with microcomputer design is to correctly interface the software with the specifics of the prototype hardware. When the software has been generated in a high-level language like PASCAL or "C", this task can become quite complex and time consuming. However, with Tek's Integration Control System (which is a standard part of Tek's compilers) the hardware/software interface definition is almost entirely automated.

From a source file that specifies your hardware configuration, RAM and ROM locations, interrupts, etc., ICS generates the necessary code and command files to execute your code in your prototype.

Linker Command File Automatically Generated

ICS creates a linker command file which adheres to the memory parameters you've specified. Constants, instructions and global variables are all automatically assigned to their correct locations within the prototype address space.

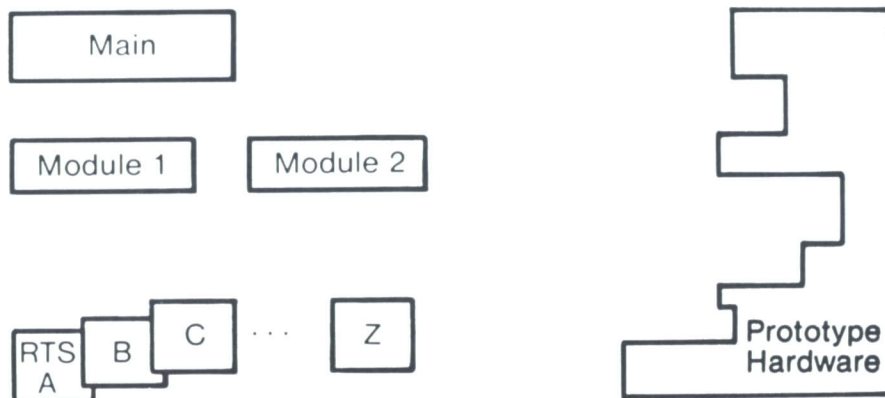
Reset and Interrupt Handler Code Supplied

ICS generates any code needed to link low-level interrupt vectors to your service routines. It also generates the object code needed to handle the initialization/reset operation. Normally, interrupt and initialization/reset code would have to be manually programmed in assembly language and then linked with the high-level code.

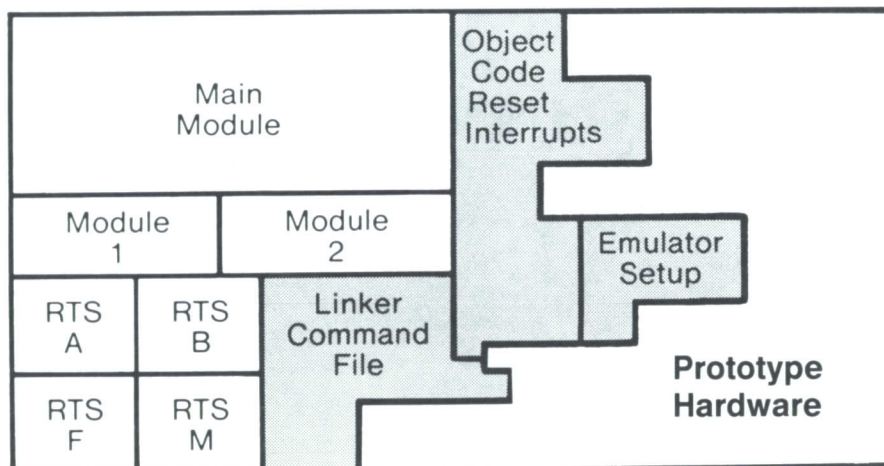
ICS Creates Emulator Command File

In addition to automating the hardware/software interface task, ICS also creates an emulator command file which allows you to download the linked object and execute it in a single command. You save both time and effort in the debug stage of the design cycle.

The Integration Puzzle



ICS Brings it Together



LANDS High-Level Debug Pulls Debug Operations Up To The Source Code Level

With most development systems, any trace information accumulated by the debug software is displayed in an assembly code format. Unfortunately, this code bears little or no resemblance to the original PASCAL or "C" source code. A great deal of time and effort is spent figuring out the relationship between assembly-level debug information and the original source code.

Tek's LANguage-oriented Development System solves this long-standing problem through High-Level Debug, which translates debug information back into its counterparts at the PASCAL or "C" source code level.

Track Real-Time Execution Bugs in Source Code

With High-Level Debug, all your debug commands can be entered using source code terminology. Breakpoints can be set on statement

numbers, procedure/function names or on variables to halt program execution. Values can be returned to calling functions.

You can even single-step through your high level language program statement by statement or reset the program to its original starting point.

Powerful Debug Display Features

High-level Debug also gives you the capability of displaying your program in high level source form, thus speeding the analysis. For instance, you can trace function calls and obtain a listing of each time the procedure is entered or exited and the value of any parameters. Also, variables can be displayed, modified and evaluated in their original source code terms. There is no need for the time-consuming task of translating hexadecimal responses to the corresponding high level language constructs.

```
# load "traffic.lo"
.....
# break wait
# go
Has walk button been pushed (y for yes, n for no)? y
Program breaks in wait at "mode2.c"#42
# source wait 9
"mode2.c"
 38:  wait(seconds)
 39:  int    seconds;          /* amount of time to wait */
 40:
 41:  {
 42:      intcounter = seconds;
 43:      /* enable(); */
 44:      while (intcounter);
 45:      /* disable(); */
 46:  } /* end wait */

# seconds
3
# seconds = 5
5
# go
-
```


Emulators and Trigger Trace Analyzer

Provide powerful insights into the interaction of software and hardware execution with emulators and trigger trace analyzer that reside in the 8540 integration unit.

Emulation is well accepted as the most effective method of debug during hardware/software integration, and Tektronix is largely responsible for establishing it. Since 1977, when Tek introduced its 8080, 6800 and Z-80 emulators, the company has been the undisputed leader in supplying the microcomputer engineering community with superior emulation systems. In recent years, this leadership position has been enhanced by advances such as real-time emulation with no wait states and support for 16-bit processors, such as the Motorola 68000 Series and the Intel 8086/80186 Series.

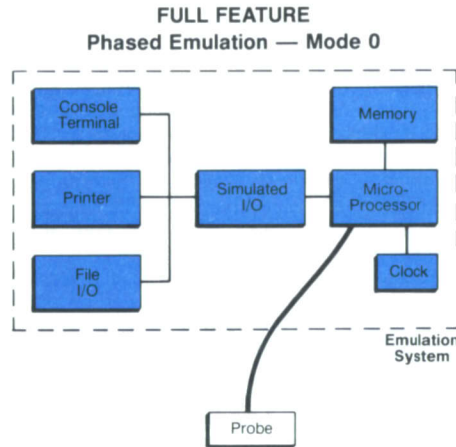
Real-Time Emulation

Real-time emulation is accomplished by executing code on a processor identical in function to the one targeted for the prototype hardware. The emulator processor is run under the control of powerful debug software that allows control and tracing of the code's execution. This debug software does not need to be linked into your code or use any of your memory or interrupt lines.

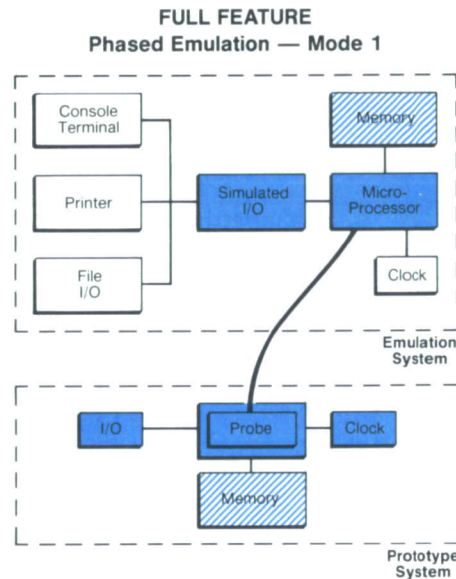
Tek's superior emulators allow your code to execute in real-time, with no wait states inserted or clock pulses stretched. This means the emulator is fully transparent to the user; therefore you do not spend time "working around" the development system.

Trigger Trace Analyzer Captures Real-Time Events

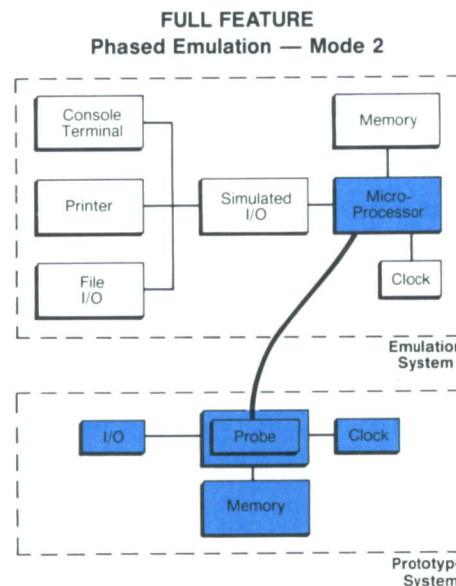
A powerful option to Tek emulators is the Trigger Trace Analyzer, which uses a high-speed buffer to capture real-time software and hardware logic events, with the prototype running at the design's full specified operating speed. Multiple word recognizers allow you to define sophisticated triggers when tracing code execution. And data qualification allows you to capture only the data you wish to see.



- Software Tested On Actual Microprocessor
- No Prototype Hardware Required
- Prototype I/O Interactions Are Simulated
- All Emulator Functions Are Available



- Prototype Software & Hardware Tested Together
 - All Clock, I/O & Interrupts Tested
 - Real-Time Environment
- Memory Substitution
 - No PROMs To Program
 - Can Debug Without Working Memory System
- Program is Transferred To Prototype Memory In Steps
- All Emulator Functions Are Available



- Final Test Of Hardware & Software
- All Prototype Memory Tested
 - PROM, ROM, Etc.
 - Dynamic Or Static RAM
- All Emulator Features Available

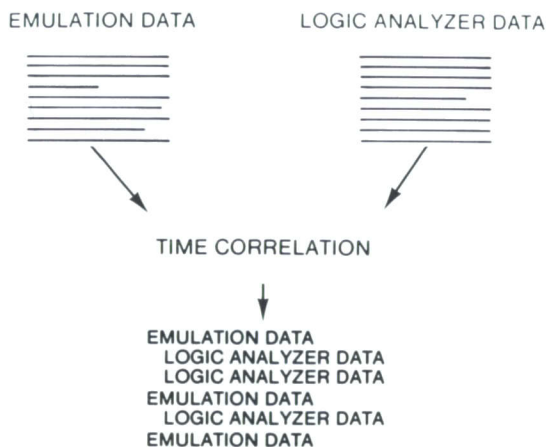
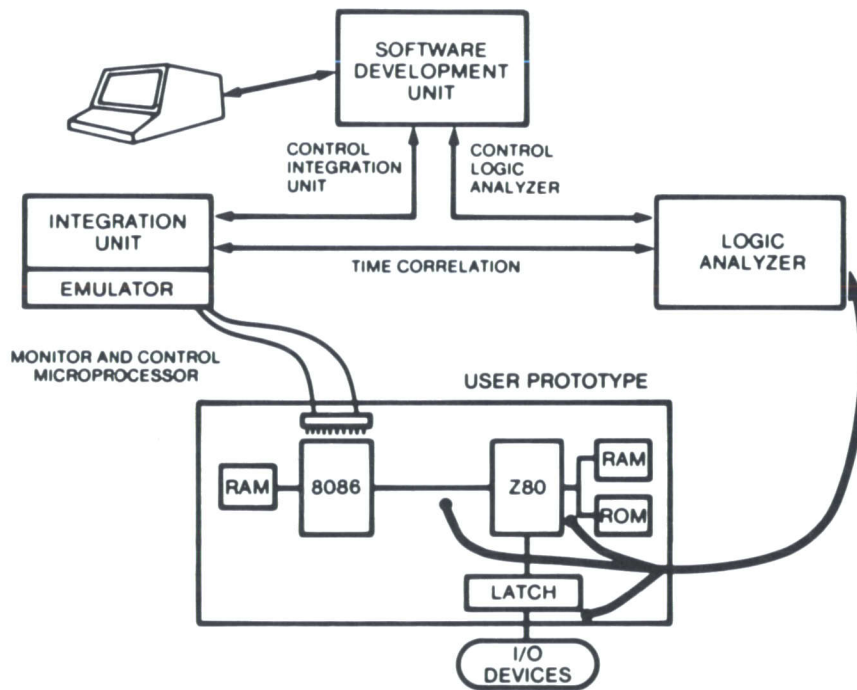
Simultaneous Debugging of Multiple Processors

Digital Design Lab (DDL)

A greater number of microcomputer designs now involve either dual processors or a single processor closely coupled to other intelligent hardware within the prototype. Tek meets the challenge of these designs through the Digital Design Lab, which combines the power of the Tektronix 8560/8540 Microcomputer Development System with the Tektronix DAS 9100 Digital Analysis System.

Through a single terminal interface, the user has complete control of both systems. An 8540 Emulator traces one processor's execution while the DAS 9100 traces the second intelligent chip. The real-time data flow from these two sources is time stamped as it is stored in memory. The data can then be formatted in a time-synchronized manner for display and analysis. This time correlation capability allows you to see the cause and effect relationships between the two circuit elements being debugged. The result is a powerful insight into separate, but related data flows within the same prototype system.

THE SOLUTION: TIME CORRELATION





These systems consist of assembler, emulator, probe and firmware, 8561 Multiuser Software Development Unit, 4105 Color Graphics terminal and 8540 Integration Unit with 64 kbytes of memory.

ORDERING INFORMATION

856140A Z80 Development System .	\$29,700
856140B 8085 Development System .	\$29,700
856140C 6800 Development System .	\$29,700
856140E 6809 Development System .	\$29,700
856140F NSC800 Development System	\$31,500
856140H 7809/8/7 Development System	\$31,500
856140J 7810/11/16 Development System	\$31,500
856140K 78C05/06 Development System	\$31,500
Option 01 — Replace 8561 with 8560	+ \$11,000
Option 02 — 512 kbyte Memory for 856X	+ \$2,000
Option 03 — 1 MEG Memory for 856X	+ \$5,900
Option 06 — Trigger Trace Analyzer with F/W	+ \$4,800
Option 10 — LSI 11/73 CPU with Floating Pt	+ \$5,000
WARRANTY-PLUS SERVICE PLANS—REFER TO PAGE 14	
N1 — Service Plan + 9 Months Service for any of the above systems	+ \$1,750

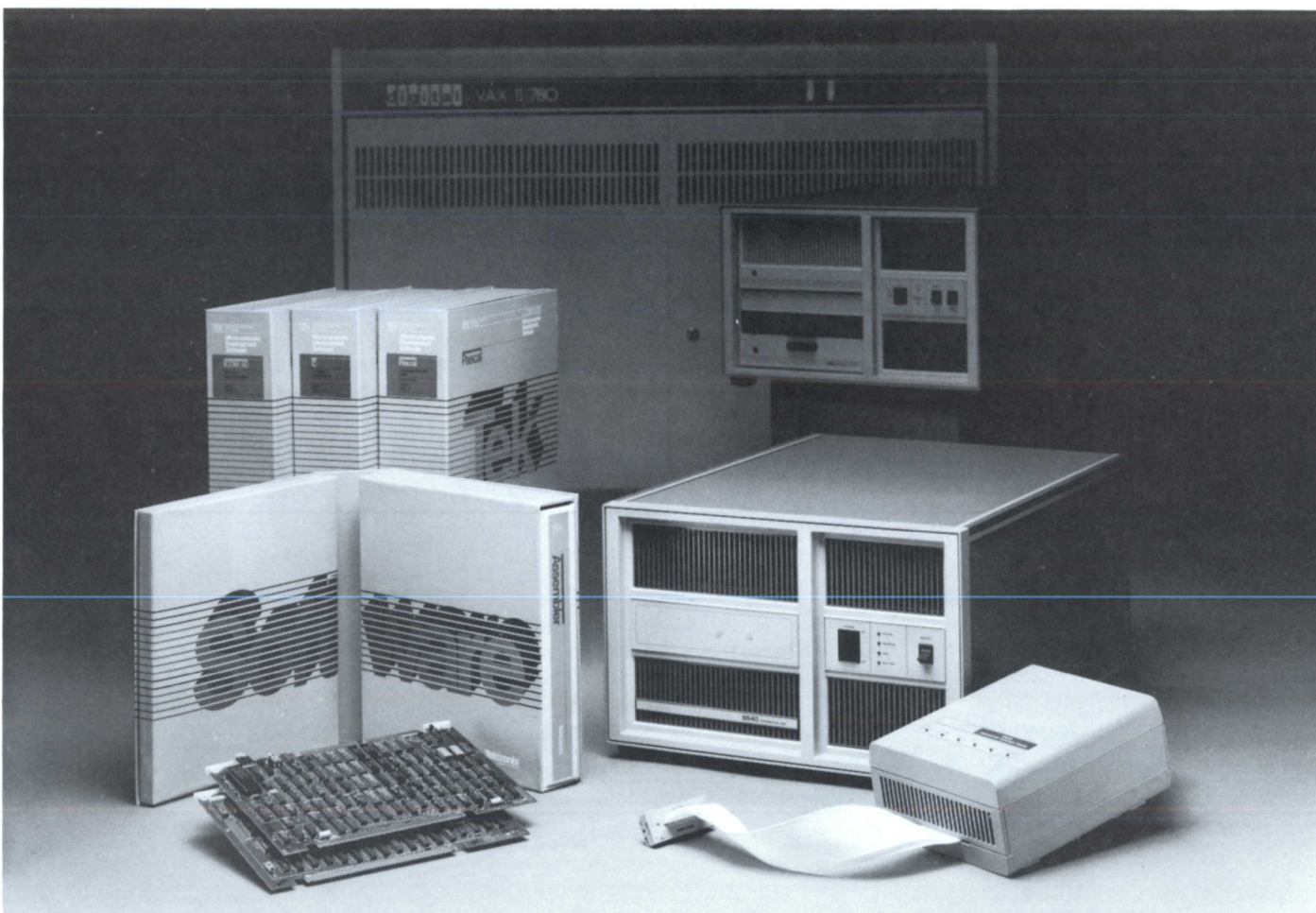
Microprocessor Development Systems

8560 Multiuser Development System	\$24,500
8561 Multiuser Development System	\$13,500
Option 02 — Floating Point Option	+ \$500
Option 03 (8560 Only) — 5-8 Ports	+ \$5,000
Option 04 — GPIB Interface	+ \$2,500
Option 05 — 512 kbyte Memory	+ \$2,000
Option 07 — 1 Mbyte Memory	+ \$5,900
Option 08 (8561 Only) — Four Total User Ports	+ \$2,500
Option 10 — LSI-11/73 CPU with Fltg Pt	+ \$5,000
Option 47 — Rackmount Option	+ \$125
8562 Multiuser Development System	\$36,000
Option 04 — GPIB Interface	+ \$2,500
Option 09 — 80 Mbyte Total Disk Storage	+ \$11,000
WARRANTY-PLUS SERVICE PLANS—REFER TO PAGE 14	
N1 (8560) — Service Plan + 9 Months Service	+ \$1,540
N1 (8561) — Service Plan + 9 Months Service	+ \$940
N1 (8562) — Service Plan + 9 Months Service	+ \$2,140
8540 Integration Unit	\$11,000
Option 01 — Communications Interface	+ \$400
Option 03 — Trigger Trace Analyzer and Control F/W	+ \$4,800
Option 04 — 64 kbyte Static Program Memory	+ \$3,000
Option 05 — 128 kbyte Static Program Memory	+ \$5,600

Option 08 — TTA W/O Data Acq I/F and Probe	+ \$3,800
Option 47 — Rackmount Kit	\$125
WARRANTY-PLUS SERVICE PLANS—REFER TO PAGE 14	
N1 (8540) — Service Plan + 9 Months Service	+ \$600
8540U04 Intel Interface Package	\$400
8550F06 Memory Allocation Controller	\$2,500

SOFTWARE

ACEDIT Ace Screen Editor	\$500
Option 1A — Support for 8560/8561	NC
PLDEDIT PASCAL Language Directed Editor	\$1,300
Option 1A — Support for 8560/8561	NC
8560U01 Text Processing Package	\$1,000
8560U02 Native Programming Package	\$1,000
8560U03 Auxiliary Utilities Package	\$500
These software packages require the Floating Point Option (Option 02 or 8560F02) in the 8560/8561.	
8560U04 Intel Interface Package	\$500
8560U05 Unicom: Communications Package	\$1,000
DDL Digital Design Lab Support	\$2,900
Option 1A — 856X Support	NC
Option 1B — VAX UNIX TU-58 Cassette	+ \$600
Option 1C — VAX UNIX Reel Mag Tape	+ \$600



The V-Systems from Tektronix are systems designed to provide complete hardware and software support for design engineers needing the highest quality design tools available. The V-Systems are configured to integrate with an existing host computer, either an 856X Series or a VAX computer and include all the hardware and optionally the software required to do so. Included with the V-Systems are Tektronix's 8540 Integration unit, 64 kbytes of memory, 16-bit emulation support (emulator and probe), Integrated Logic Analysis, and as an option Tektronix's unique LANDS for high level language support. LANDS is available for either PASCAL or "C" and includes a Language Directed Editor, Compiler, Integration Control System, High Level Debugger, Assembler, Linker, and for supporting VAX computers, ICOM40.

ORDERING INFORMATION

V68000A Emulation Support	\$24,900
V68008 Emulation Support	\$24,900
V68010 Emulation Support	\$24,900
Option 01 — 128 kbyte Memory	+ \$3,000
Option 02 — MAC Board	+ \$2,500
Option 03 — TTA 8-Bit External LA Probe	+ \$1,000
Option 1A — P-LANDS Support for 856X	+ \$9,000
Option 1B — P-LANDS VAX UNIX TU-58 Cassette	+ \$12,000
Option 1C — P-LANDS VAX UNIX Reel Mag Tape	+ \$12,000
Option 1E — P-LANDS VAX VMS TU-58 Cassette	+ \$12,000
Option 1F — P-LANDS VAX VMS Reel Mag Tape	+ \$12,000

Option 2B — C-LANDS VAX UNIX TU-58 Cassette	+ \$12,000
Option 2C — C-LANDS VAX UNIX Reel Mag Tape	+ \$12,000
Option 2E — C-LANDS VAX VMS TU-58 Cassette	+ \$12,000
Option 2F — C-LANDS VAX VMS Reel Mag Tape	+ \$12,000
V8086 Emulation Support	\$24,900
V8088 Emulation Support	\$24,900
Option 01 — 128 kbyte Memory	+ \$3,000
Option 03 — TTA 8-Bit External LA Probe	+ \$1,000
V1750A Emulation Support	\$33,500
Option 01 — 128 kbyte Memory	+ \$3,000
Option 02 — MAC Board	+ \$2,500
Option 03 — TTA 8-Bit External LA Probe	+ \$1,000
Option 1A — Assembler, 856X	+ \$2,000
Option 1E — Assembler, VAX VMS TU-58 Cassette	+ \$4,000
Option 1F — Assembler, VAX VMS Reel Mag Tape	+ \$4,000
Option 3A — PIA W/Flying Leads, Clips	+ \$750
Option 3B — PIA W/Flying Leads, Pigtail	+ \$650
WARRANTY-PLUS SERVICE PLAN REFER TO PAGE 14	
N1 — Service Plan + 9 Months Service for any of the above V-Systems	+ \$990
VZ80 Emulation Support	\$17,900
V8085 Emulation Support	\$17,900
V6809 Emulation Support	\$17,900
VNSC800 Emulation Support	\$19,700
Option 03 — TTA Back Panel and Probe	+ \$1,000
Option 04 — Communications I/F for 8540	+ \$400
Option 1A — Assembler for 856X	+ \$1,700
Option 1B — Assembler for VAX UNIX TU-58 Tape	+ \$2,500
Option 1C — Assembler for VAX UNIX Reel Mag Tape	+ \$2,500
Option 1E — Assembler for VAX VMS TU-58 Tape	+ \$2,500

Option 1F — Assembler for VAX VMS Reel Mag Tape	+ \$2,500
WARRANTY-PLUS SERVICE PLAN REFER TO PAGE 14	
N1 — Service Plan + 9 Months Service for any of the above V-Systems	+ \$820

INTERFACE SOFTWARE

ICOM40 Integrated Communications System for 8540	\$2,000
Option 1B — VAX UNIX TU-58 Cassette Tape	NC
Option 1C — VAX UNIX Reel Mag Tape	NC
Option 1E — VAX VMS TU-58 Cassette Tape	NC
Option 1F — VAX VMS Reel Mag Tape	NC
Option 3B — VAX UNIX ICOM40 Source TU-58	+ \$6,000
Option 3C — VAX UNIX ICOM40 Source Reel	+ \$6,000
COLORKEY ColorKey+ User Interface	\$1,500
Option 1B — VAX UNIX TU-58 Cassette Tape	NC
Option 1C — VAX UNIX Reel Mag Tape	NC
Option 1E — VAX VMS TU-58 Cassette Tape	NC
Option 1F — VAX VMS Reel Mag Tape	NC

STRUCTURED ANALYSIS TOOLS

STRUCTA Structured Analysis Tools ..	\$9,500
Option 1A — Support for 856X	NC
Option 1B — VAX UNIX TU-58 Cassette	+ \$5,000
Option 1C — VAX UNIX Reel Mag Tape	+ \$5,000

For information concerning other configurations, options, peripherals, available software, software subscription services, applications assistance, and customer training contact your local MDP Sales Engineer.