History of the P?S/8 Monitor System for the PDP-8

First written by Charles J. Lasner, New York, NY, 1975
Last revised by Charles J. Lasner, New York, NY, 1982
Transcribed into XHTML, with permission, by Patrick J. Gleason, Syracuse, NY, 2005

Foreword: The document that follows is provided as a historical reference. It was given to me, printed by what appears to be an LA-30, or possibly an original Centronics dot matrix printer, on tractor-feed teletype paper in 1982. Charlie (Charles J. Lasner) gave it to me around the time when I first met him in 1982, to fill me in on events that transpired in the late 1960s and early 1970s, as I had become the third person in the known universe to own a personal minicomputer in 1977. During the years referred to by this document, big-metal computers had been available for more than a decade, but were very expensive (prices in the millions of dollars). Many large businesses and educational institutions had "mainframe" computers made by companies like IBM, Burroughs, National Cash Register (NCR), Honeywell, General Electric and Control Data Corporation (CDC). The CDC Cyber-170, with its sixty bit word size and the IBM 360 with a 32-bit word were the biggest of these. In the early 1960s, M.I.T.'s Lincoln Laboratory, using transistorized logic circuits made by Digital Equipment Corporation (DEC) had built what was the rough equivalent of the microprocessor for those days, a machine with 1,024 12-bit words of memory called the "LINC" (Laboratory INstrument Computer), which was the first "minicomputer". DEC, building on Lincoln Lab's minicomputer concept, built and marketed two minicomputer products: the PDP-8 (Programmable Data Processor Number 8) and the Linc-8, which was a combination PDP-8 and LINC constructed in one cabinet with a lot of shared circuitry. Linc-8's were very rare; only 157 ever being produced. Charles Lasner has one that is fully functional to this very day, and two more that could be made functional with some work, in his basement. I have a fourth one in my garage, which is in need of some minor repairs. When Charlie bought the first Linc-8, it was utterly unheard of for a private individual to personally own a computer, because their cost was so prohibitively high. What you are about to read here is the history of the earliest origins of "personal computing", written by the hand of the person who owned the first personal computer, in part, written on that first personal computer. Concepts and code originally developed by Richard Lary and other members of the Poly Question Society (P?S) have been carried forward through operating systems named OS/8, RT-11, CP/M, MS-DOS, and OS/2 to become part of the core of every personal computer operating system in use today. Unrelated to all this were the mainframe operating systems: Multics, then Unix (a fix for Multics from Bell Laboratories for the PDP-7, which they dumped into the public domain because it never worked, but was resurrected many years later as Linux by Linus Torvald) and RSX (Resource Sharing Executive) for the PDP-7, then PDP-9,

then PDP-15, then PDP-11, and fnally as VMS (Virtual Memory System) for the VAX (Virtual Address eXtension) PDP-11, then DEC Alpha, then Intel Itanium (a ressurection of a really bad Hewlett Packard mainframe CPU that was known to be a total loser in the 1960s, on a single chip, that's really slow and tends to burst into flames). As a sideways point of interest, we also now have Windows NT/2000/2003/XP/Vista, which is SCO Xenix (a stripped down, buggy, version of Unix) using Windows for Workgroups 3.11 as its shell, and disk device drivers and networking from OS/2, with a few memory management concepts and routines from RSX and VMS, all tied together with a lot of duct tape, epoxy and string.

1.1 Introduction

This document is presented as an overview of the events leading up to the current implementation of the P?S/8 Monitor system, including the evolvementfrom the former R-L Monitor System, sometimes referred to as MS/8. A secondary purpose is to clarify the exact nature of just what the P?S Monitor system is, namely an operating system for the PDP-8 series of computers. This will, hopefully, then eliminate all rumors of its possible alternative purposes, some of which have come forth from people who should have not prejudged it merely by its name and association with certain people, etc.

1.2 Disclaimer

The description of the historical events revealed in this document are as accurate as the author can recall. Any actual departure from the truth is purely without malice and unintentional. However, various individuals have maligned the Monitor system and/or its implementors/supporters, causing misinterpretations of their intents, or other such detrimental conditions. These people will be named and their actions will be described as accurately as possible, without regard for possible damage to their reputations, etc.

1.3 Style

Various portions of this document definitely have a light-hearted style. This is the usual mode of interaction between members of the P?S, and is necessary for proper interpretation by readers of the document associated with the organization for proper confirmation of the facts as presented. Some of the information was obtained by members of the P?S directly and then related to the author (also a member); no editorializing will be done on this second-hand (or sometimes third-hand) information.

1.4 Definitions

Various names, phrases, and technical terms are presented in this document. The reader is assumed to have some familiarity with Digital Equipment Corporation (DEC) *[purchased by*

Compaq Computers for $9.6 billion in January, 1998; purchased by Hewlett-Packard for $19 billion in April, 2002], and its (and others) hardware and software for its various products, most notably the PDP-8 and its descendants. References to these items will be minimally defined; the rest will be defined in sufficient detail to explain their relevance. The reader is referred to the members of the P?S for further information.

1.4.1 Bic

Bic® is a trademark of the Waterman-Bic pen company. It is used as an adjective for anything that "works the first time and every time." *[The father of Charles J. Lasner owned a store in Middle Village, Queens, New York, NY, which was one of the first retailers ever to sell Bic pens, while Mr. Lasner was still in his youth.]*

1.4.2 BIN

BIN is the format used by binary programs for the PDP-8 paper-tape operating system. There is a program, called the "BIN Loader", which is used to load these binary files into core memory. The BIN Loader, is itself, loaded into core by another program called the "RIM (Read In Mode) Loader", or a refinement of the RIM Loader called the SLIM loader. The RIM Loader, or SLIM Loader, is placed into core using the console toggle switches and light bulbs.

1.4.3 Brooklyn Poly

Brooklyn Poly is a common name for the Polytechnic Institute of Brooklyn (PIB), where most of the members of the P?S met each other. It probably is known currently by another name, due to various mergers of institutions (probably PUNY or PINY). Poly owned a 4K PDP-8 with TU55 (one) and TC01, which was the original hardware used to construct the original Monitor System. *[A 4K PDP-8 is a minicomputer built by DEC, circa 1965, prior to the invention of integrated circuits, from circuits of discrete transistor, diodes, resistors and capacitors. The original model came with 4,096 12-bit words of memory, one general porpose register, eight hardware instructions, and a Teletype model ASR33 for input and output. Programming was done by manually setting bits in memory using toggle switches and light bulbs on the console.]*

1.4.4 Burness

Jack Burness is a programmer extraordinaire, and sometime consultant/employee of DEC. He was responsible for the hardware arrangement between Brooklyn Poly and DEC whereby Poly had use of a PDP-12 for nearly a whole year (on loan), primarily for the benefit of the P?S. He was responsible for many PDP-12 based projects, including a vast rewrite of the LAP6 *[Laboratory instrument computer Assembler Program number 6]* programming system for reasonable use on a PDP-12. At one point, his works (along with others) were considered

important enough to warrant a separate programming group. Jack Burness is also the author of the legendary PDP-11 GT-40 "Moonlander" program. A complete list of his many accomplishments is beyond the scope of this document.

1.4.5 DEC

Digital Equipment Corporation (DEC) is a multinational corporation, headquartered in Maynard, Massachusetts *[see note above about recent acquisition by Compaq, then Hewlett Packard]*. It is responsible for various hardware and software (also trademarks) appearing in this document. Various members of the P?S (and others) are now, or have been, employees of DEC. Other than references to P?S, etc., all other items are probably copyrights or trademarks (or both) of DEC. This document does not attempt to usurp this, but yet must mention them to explain the relationship between the various persons and items, etc.

1.4.6 Dectape

Dectape is the DEC name for an unusual mass storage device, defined as a serial disk on tape. Totally block replacable, Dectape ranks among the most reliable devices for data storage long-term. Due to its serial nature, it can have access characteristics that are severely downgraded by bad software handling. The P?S Monitor System, when running on Dectape, attempts to handle this problem in an efficient manner by having maximum buffering, where possible, and clever placement of blocks, allowing meaningful development even if running on a single Dectape drive. Dectape also has been implemented with a more feeble controller; the TD/8E. This device uses the same format tapes, but requires much larger software effort and bus loading to accomplish what the TC01 does with ease. P?S/8 supports the TD/8E hardware with efficiency similar to the TC01. Also supported by P?S/8 is the Linc-8 and PDP-12 Linctape, which is a hardware cousin of Dectape, with a similar, though incompatible, format. Utilities exist to interchange information between tapes of different formats, which run from P?S/8 on either a Linc-8 or PDP-12. (A possible hardware project may be accomplished to allow the TD/8E to do this also.)

1.4.7 EAE

The Extended Arithmetic Element (EAE) was a rare, high-cost, option from DEC for the PDP-8 computer that added hardware instructions for integer multiplication and division. In later models of the PDP-8 (PDP-8e and later), the EAE was included as a standard part of the base CPU. This is analogous to the 80x87 floating point processor for Intel's 80x86 series of microprocessors, which was an extra cost add-on until the 80586 (Pentium).

1.4.8 Focal

Focal [FOrmula CALculator] is a high-level language implemented on all major DEC machines. It is most notably the target much effort by hoardes of people to modify it for purposes ranging from inane games to sophisticated custom applications. P?S/8 Focal represents an elegantimplementation of a totally modifiable Focal with access to most P?S system resources, either built-in or trivially available through straight-forward "patching", according to well-established (and documented) guidelines.

1.4.9 Friends of the Boston Massacre

The Friends of the Boston Massacre (FBM) is a New England based organization not necessarily affiliated with the I?S.

1.4.10 GE (235) BASIC

GE's original time-sharing BASIC *[Beginner's All-purpose Symbolic Instruction Code]* was the original inspiration for the R-L Monitor System. It is very "user-friendly" and lends itself to use by beginners, as well as advanced programmers.

1.4.11 George

George Thissell was an employee of DEC circa 1971. He was most notably a hardware group manager, of possibly reasonable prior reputation (in hardware situations). In a move typical of ludicrous managerial decisions, George was made head of the group responsible for PDP-8 software development. When P?S members offered P?S/8 to DEC as a possible alternative operating system, he rejected the offer, sight unseen, because in his opinion, a 4K operating system would "discourage memory sales". Thus, P?S/8 was never properly evaluated by any DEC personnel is a position to do anything about such an arrangement.

It should be mentioned here that this offer was made without monetary considerations, rather as a charitable act from Brooklyn Poly. DEC had already dealt with members of the P?S and others in such arrangements before. Most notably, through the efforts of Jack Burness and others in DEC and Brooklyn Poly, a PDP-12 system was made available to Brooklyn Poly for a year, primarily for members of the P?S, but this machine actually served some concrete educational needs of Brooklyn Poly. Some students even went on to become DEC employees, primarilly due to their encounter with this P?S-inspired machine. It is often the case that DEC benefits (beyond the obvious tax advantages) more than anyone else in these "educational" dealings, so it was in the company's best internest to look harder at P?S/8. It is most notably poor business practice that they chose not to. This is further borne out by more recent dealings between DEC and small vendors of software for many DEC systems. The main point is that DEC personnel have been continuously aware of the existence of P?S/8 since at least 1971, but failed to act on this information, in spite of several proposals by P?S members, both inside and outside DEC.

In 1980, an unnamed manager decided against P?S/8 merely because it was "an additional system" that they didn't want to "support additional systems". With decisions like this coming from DEC, it is no wonder that Data General is the second largest mini-computer company in the U.S. (and gaining).

1.4.12 Green

John C. Green is a member of the I?S who worked for DEC and Data General, who notably hated paper-tape and Focal.

1.4.13 Hacker

A hacker (non-derogatory usage) is a generally suave programmer who does an excellent job on something because that is his specialty. Hackers do do what they do because they want to, not because they have to. This generally leads to a better product than paying an un-inspired employee to "hack out" a piece of work (this is the derogatory use of the word) on an artificial time schedule. This is probably due to the artistic nature of programming and other endeavors in general, proving that you can't pay an artist to work, and that you must believe a (program) writer is working even when staring at the ceiling (or playing Adventure). The reader is referred to the MIT file, "HACKER.JARGON", available on many PDP-10 systems, for further understanding of hacker-related definitions. *[Being, myself, one of the world's most formidable "hackers", and one of only four people in the history of the known universe to single-handedly write an entire operating system with all utilities and applications, I will say authoritatively, that really good programming begins to occur when it becomes brainstem activity. Programming a computer is first learned as "left brain" (mathematical) activity. In some individuals, within a few weeks, it becomes "right brain" (language, poetry, music, art) activity. After a year or two as "right brain" activity, it becomes conditioned reflex, then finally, brainstem activity (with heart beat, breathing, etc.) When this final transition is complete, the individual is able to read and write binary machine code much faster than most people can speak their natural language, and the code is said to come via "direct inspiration from God", so debugging is no longer required. Thus, by brainstem activity, we write "Bic" programs in any arbitrary language at speeds of more than 5,000 lines per day, fully tested and documented. The typical rate for writing flawless, fully documented and tested code using left brain activity is 1 - 2 lines per man-day. When programming becomes brainstem activity, the programmer can consciously be doing other things like reading a book, watching television or working complex calculus problems while writing flawless computer code as a "background job".]*

1.4.14 I?S

The International Question Society is an improper superset organization of the P?S. Its membership usually consists of those individuals who have strayed from the from the environs

of the P?S, or who never were bona fide P?S members (though they might consider it otherwise). The usual reason for membership in the I?S is because one's interests or geography prevent mainstream P?S participation.

1.4.15 Murphy's Law

Professor Edsel Murphy of the School of Hard Knocks was the first person to put forth the proposition that, "if something can go wrong, it will." Corollaries of Professor Murphy's Law exist in all scientific endeavors, most notably the Law of Selective Gravitation, which states that an object moved accidentally will fall where it will do the most damage. *[Another corollary that requires mention in this context is the Law of Mean Objects, which states that, "certain objects are just mean." Traffic signals can sense emergency and turn red because they are mean. Computer hardware can sense an approaching deadline and chooses those times to fail because it is mean. Coins, when randomly dropped through a hole in the pocket or accidental mishap, will take the shortest path to the most inaccessible place and their speed will be directly proportional to their value, because they are mean. Have you ever noticed that, in cinema, an otherwise perfectly functioning motor vehicle will always refuse to start if imminent and mortal danger is approaching? ...another mean object]*

1.4.16 OS/8

OS/8 is an operating system, made by DEC, for the PDP-8. It requires, as a minimum configuration, 8K (8,192) (12-bit) words of memory and a disk or Dectape drive; though it runs a lot better with 12K of core and two disk or Dectape drives. P?S/8 does, essentially, everything that OS/8 does, but does it in 4K of core with one Dectape drive, and quite a bit faster. OS/8 has a command line interface with commands which are virtually identical to those of Windows, OS/2, MS/PC/DR-DOS, CP/M and RT-11. This is not an accident; each of these systems was used to edit, assemble and compile the source code for the generation which followed. The only one that was "assembled" by hand on scraps of paper, and placed in memory by toggle switches was the R-L Monitor. While Richard F. Lary worked as an employee of DEC, he expaned and developed the R-L Monitor to become OS/8, which is the ultimate ancestor of all operating systems used by personal computers today.

1.4.17 PAL II

PAL II (PDP-8 Assembly Language, 2nd attempt) is an early assembly language for the PDP-8,

1.4.18 PAL III

PAL III (PDP-8 Assembly Language 3rd attempt) is the assembly language of the paper-tape

operating system for the PDP-8, and a derivative of PAL II.

1.4.19 PAL-D

PAL-D is the assembly language of the PDP-8 Disk Monitor operating system, and a derivative of PAL III.

1.4.20 PAL8

PAL8 is the assembly language of the OS/8 operating system, a successor to PAL II, PAL III and PAL-D.

1.4.21 PAL10

PAL10 is a cross-assembler, which produces binary code for the PDP-8 on a PDP-10.

1.4.22 P?S PAL

P?S PAL, written by Charles Lanser, is the latest and most sophisticated assembly language for the PDP-8.

1.4.23 PDP-8

The PDP-8 *[(Programmable Data Processor number 8)]* is a remarkable computer made originally by DEC. It is currently manufactured by many manufacturers under various names with varying configurations, all the way from hand-held devices to two ton cabinet-based systems. All of these systems share a common architecture, making it unique among small/medium computer systems to have evolved from a soldered back-plane, germanium transistor machine (PDP-5), circa 1963 to a microprocessor chip version (Harris 6120), circa 1980, with all levels of hardware in between commonly available. Good software exists to support any viable configuration of this machine, making it competitive and often superior to other more "hyped" machines. As a point of interest, this document was prepared on a PDP-8/E resource sharing, multi-tasking, news document management system, for up to 40 (perhaps more) users, an example of an efficient, medium scale specialized application of PDP-8 hardware/software. *[Update: The Intel/AMD x86 CPU architecture, which is used by nearly all computers in current manufacture, is a single accumulator, RISC (Reduced Instruction Set Computer) architecture, very similar to that of the PDP-8, but based on an eight bit byte rather than a twelve bit byte. CISC (Complex Instruction Set Computer) architecture, which originated with the PDP-11, is now as extinct as the dinosaurs.]*

1.4.24 PDP-12

The PDP-12 is a PDP-8/I with LINC hardware built in. It also has many internal peripheral devices, most notably the VR-12 scope and dual Linctape drives. PDP-12 systems are often configured with many additional peripheral devices, such as disks, lineprinters, etc., as it was perceived as a "big machine", the top of the line. With an impressive manual console, designed for programmers, the PDP-12 is generally perceived as the crowning achievement of the PDP-8 world (although not necessarily the most advanced, architecturally speaking.) It was originally given its own grouping, but was later joined into the PDP-8 Software Group.

1.4.25 PDP-12 Systems Programming Group

The PDP-12 Systems Programming Group was a short-lived, separate group from the PDP-8 group, formed to develop distinct PDP-12 programming. Eventually, it was realized that the best way to deal with the PDP-12 was to treat it as a PDP-8 to benefit from the mainstream of software effort, not to write a PDP-12 only system. After the regrouping, the PDP-8 group handled all PDP-12 programming support.

1.4.26 P?S

The Poly Question Society (P?S) (or Polytechnic Question Society as it is sometimes erroneously known) is an organization of questioners, many of whom are also programmers. In fact, members of the P?S are responsible for a significant percentage of the software systems available for many DEC systems upon which the general users of such systems depend on a daily basis. In some cases, members of the P?S acted as independent contractors or employees of such companies as DEC, etc., but in any case, membership in the P?S usually indicates one's high-quality software competence.

1.4.27 Richard Lary

Richard Lary is a programmer extraordinaire, member of the P?S, WCFMPG, employee of DEC, etc., who wrote the first R-L monitor (aka MS/8) and deposited it into memory by hand, etc. Richard Lary's accomplishments are described, in part, elsewhere in this document; a complete description of his achievements are beyond the scope of this document.

1.4.28 Mouth

Mouth is an affectionate alter-ego tp Richard Lary.

1.4.29 RIM

RIM (Read In Mode) is a PDP-8 program of sixteen instructions, which must be toggled into

core memory by hand, using the console switches and lights, before doing anything else with the paper-tape operating system. If the core is without form and void, then a human must toggle in the RIM Loader or SLIM Loader, which then reads in the BIN Loader, which then reads in the program you want to execute. The RIM Loader is an example of a "bootstrap" program. All modern operating systems are started by a similar procedure, except that their equivalents of the RIM Loader are stored permanently in a Read-Only Memory (ROM) called the BIOS (Basic Input/Oputput System). With core memory, a ROM is not needed because core memory is an array of tiny magnets, which does not "forget" its contents when the power is turned off.

1.4.30 Lineprinter

A lineprinter is a device used to acquire hard-copy listings of a file. It is, hopefully, faster than a teletype. Lineprinters are more expensive than teletypes, and are often deleted from systems by managers who don't understand their purpose. The P?S System supports a lineprinter, but doesn't absolutely require one. Realistic programming development of anything but the tiniest program virtually demands a lineprinter. *[Note: When Mr. Lasner wrote this, in 1975, video terminals had not yet been invented. Personal computers, as we know them today, had not yet been invented. Portable, pocket calculators were still a novelty. The only way to see a program was to print it on a lineprinter or a teletype.]*

1.4.31 Mario DeNobili

Mario DeNobili (sometimes misspelled as Mario Denobili or Alonzo Dorsali) is the emperor of the P?S. Without his great presence and guidance, this document would have been finished three years sooner.

1.4.32 SLIM

The SLIM Loader is a refinement of the RIM Loader, which is easier to use because it has only ten instructions which must be toggled into core using the console switches and lights.

1.4.33 TC01, TC08 and TU55, TU56

These are the hardware components of Dectape (controllers and drives, respectively).

1.4.33 Teletype

Teletype is a trademark of the Teletype Corporation, and refers to the relatively slow terminal devices used in the interactive software environments successfully applied to DEC and other manufacturer's computer systems. All such systems support the actual device (model 33

teletype), but the use of new devices that obey the same protocol is more desirable. Early P?S/8 configurations had to put up with teletypes and the software includes features to limit unnecessary printout in installations lacking a lineprinter.

1.4.34 WCFMPG

The William Claude Fields Memorial Programming Group (WCFMPG) is an affiliated organ of the I?S. Members of the WCFMPG are responsible for many memorable programming events.

1.4.35 Various Other Mass Storage Devices

Mentioned throughout this document are other Mass Storage Devices, such as DF32, etc. These are DEC-supported PDP-8 peripherals. Some of them have only been read about (outside DEC itself) due to their extreme cost. Others are quite common due to their relative inexpense.

1.4.36 Additional Definitions

The reader is referred to the appropriate P?S document for additional definitions. The usual P?S security clearances will apply (send for P?S-01-001-RIF-00 Request for Information Form revised, 1980 to Mario Denobili, P?S at the usual address).

1.5 Ancient History (B. L.) (Before Lary) at Brooklyn Poly

In the beginning, there was a PDP-8, a teletype, and a TC01/TU55 Dectape and control unit, and the software was void and without form... Also present were Richard Lary ("my good friend", mario Denobili, 1968), and PAL III, a DEC assembler program. While PAL III theoretically was useful, it was essentially designed for use in a paper tape environment: The contents of memory was alternated between an editing program, and PAL III. The editor is used to update the source program on paper-tape by reading it into memory, and then punching it out onto a new paper-tape, until a point of modification is reached, at which point the user invokes editing commands to make changes, followed by punching out the modified portion, followed by the rest of the tape, otherwise unmodified. The contents of memory is then changed to the PAL III program by loading in the binary paper-tape of PAL III. At this time, the user reads in his source paper-tape several times to produce a binary tape and/or a listing on the teletype (or punch?). If possibly acceptable, the user then loads in the paper tape of his program (produced during the second pass of PAL III over his source paper tape), and then manually starts it to observe its action or inaction!). Assuming normal development cycling, the editor is then loaded into memory (it also is a binary paper-tape) to initiate further source editing to essentially repeat the above described process. *[Understand that, in those days,*

writing a working computer program was neither a job for the feeble minded nor the faint of heart. Programs had to be fully conceived, totally error free, in the mind of the programmer because there were no video terminals and printed listings could take many hours to produce, provided that the teletype didn't rattle itself to pieces in the process. Anyone unable to write a non-trivial "Bic" program; i.e., mentally conceptualize at least 12,000 bits with zero errors, was doomed to walk away in frustrration and failure.]

This paper-tape process was seldom used at Brooklyn Poly, because it was incredibly slow. (N. B. the above method is intolerable on a high-speed reader/punch (300 cps input, 50 cps output), however Poly only had a teletype reader running at a mighty 10 cps in and out!) It was easier to assemble by hand! So Richard Lary, with his great programming skill and forthright fingers, toggled into core the original Keyboard Monitor System.

1.6 R-L Monitor System

The System quickly became known as the R-L Monitor System (or MS/8). It had known (and unknown) bugs, but still represented a great step forward over former systems (primarily DEC's Disk Monitor System, especially on Dectape) for general purpose program development. The R-L Monitor System was used for writing Poly Basic, a stand-alone BASIC system, meant to imitate the environment of typical BASIC time-sharing systems as seen by one user, etc. Other programs were written by Richard Lary and his associates: Poly Basic (original version), Poly Snobol (version 0.5), Poly Lisp, and String Language 8 (SL/8), a stand-alone system like Poly Basic, but with a string manipulation language as its base. The System also was used by the PDP-8 programming community now known by their membership in such organizations as: WCFMPG, P?S, I?S, Friends of the Boston Massacre, PDP-12 Systems Programming Group (defunct), etc. The limitations of the R-L Monitor System were well known by these individuals, who put up with them for a while (3.5 years) due to, in part, the laziness of these people as a group, who would rather put up with the problems and just concentrate on their own program, rather than do something to improve the situation.

1.6.1 Utility of the R-L Monitor System

The R-L Monitor System, as it was written then, more or less fulfilled the main requirements of the typical assembly language programmer:

1. Quick editing of a file for changes to the source in a convenient manner (insertion and deletion of text, etc.)

2. Quickly assembling the corrected source program.

3. Quickly loading the binary program into core and executing it.

4. Easily returning to the Monitor when the program didn't work because of logic errors in the program or successful termination of the program as written (non-Bic programs) (by Murphy's Law, this can never happen).

5. Easy restart of the Monitor by manual means, which includes the possibility of having to key in a few instructions to replace a wiped out bootstrap if the program last loaded "really" bombed.

1.6.2 Advantages of the R-L Monitor System

The R-L Monitor System was a very realistic system for its intended hardware/software/political environment. Among its advantages are:

1. It was designed to run off a single Dectape drive with manual intervention for any file transfer achieved through core-only copy, as the single Dectape was the only mass storage device available.

2. It was designed for problematic devices like Dectape, a fundamentally serial device which can take many times longger to transfer information than other mass storage devices (disks, drums) unless this serial factor is taken into account. The Monitor's tape routines can read several adjacent blocks of information which are sequential on the tape in one subroutine call operation. To attempt to read sequential blocks separately would require the tape to be turned around each time because of searching considerations. The system was designed to generally read in sequences of tape blocks instead of widely scattered individual blocks. The inability to read in more than one block at a time and the scattering of blocks during systems operations is what led to the downfall of what is known as: "the Disk Monitor System on Dectape" (retired by DEC) which was and still is the only DEC-supported program development system for 4K machines (yes, George, there are 4K machines out there in the real world!!).

3. It was designed to use the mass storage device itself for file input and output, rather than just program storage. There exists an additional system for 4K machines and a single Dectape known as the "Dectape Library System", which serves to preserve core images of portions of memory as it finds it (It doesn't directly participate in the events leading up to the existence of the desired core contents!). This system has acceptable multi-block input/output and uses the standard TC01 bootstrap convention for compatibility with most other systems. A desirable combination of system functions could include the use of both the R-L Monitor, and the DEC Library System together as the DEC Library System saves core images and nothing else, and the R-L Monitor System does everything else (several R-L sub-systems were actually created by transferring core images between the two systems with R-L utilities).

4. It includes many tape-motion saving devices such as placement of scratch files near the beginning of the tape to speed assembler execution speed, and a keyboard monitor command character (^R) to allow overlap of command typingg and tape rewinding in anticipation of the need for swapped in system blocks (for example: `.RUN PAL,(^R) FILE1,FILE2)`

5. It puts as much of the system as was practical into core at the same time to eliminate tape motion altogether during any reasonable monitor command (no swapping unless the tape was required to move anyway).

6. It has commands designed to be easy to learn for the inexperienced programmer, which are very straightforward (LOad file, SAve file, etc.), but were sophisticated enough to satisfy the needs of even the most jaded (Green, not necessarily J. C.) systems programmer without insulting him.

7. It has a file structure which was not assembly oriented to allow the use of high level languages (FOcal, BAsic, SNobol, etc.) as well as assembly language by all potential users of the system in a manner close to time-sharing systems such as GE BASIC, etc.

8. In connection with 7. above, the system used line numbers to full advantage instead of the standard DEC editor systems without line numbers considered by many to be (putting it mildly) less than ideal. (Editor's note: I typed up this portion of the document on a DEC editor originally. I had to retype this line because I couldn't locate it!)

9. The files are structured internally to appear to either have line numbers or not, depending on how you accessed the file. In addition, the use of line numbers provided a permanent record of the source as the latest string inserted can be verified from the teletype listing where it was typed. This saved much time over repeated listing of the source file as is required with other editing systems. In addition, the use of the line numbers didn't materially slow down the listing of the files, rather this made it easier to verify which section of a listing was desired to be listed (although of no consequence for a complete listing). For compatibility with paper-tape and other external systems, the System provides text-oriented commands (such as PUnch which outputs the current file surrounded by leader/trailer on the paper-tape punch) to input and output paper-tape versions of the source files without line numbers.

10. The System has a batch facility to self-execute without the user/operator typing in the commands as they are needed. Rather, the system operates from previously prepared command files (this process is adaptable to cards, paper tape, etc.) to allow the system to run itself for hours, if necessary. This freed the user from slaving over a hot teletype to allow other functions (sleep, food, work, etc.) to overlap with computer operation. In an environment lacking a lineprinter, the user now had a painless way of listing his programs for further use without actually waiting around to direct the system every few

minutes. (How many programmers in the real world (outside Maynard, Mass.) would like a listing of their program, but decided against it due to time considerations, shared machine usage, etc. At Brooklyn Poly, this situation prevailed due to the number of people waiting to use the machine, but a lineprinter was considered a luxury at this installation. Ask the people who have multi-colored listings from a teletype for further information: red = tuesday's corrections to monday's yellow changes to last friday's green editing notes, etc. There are available up to 50 distinct colors of colored pencils from at least 2 different manufacturers for those who are hard pressed).

1.6.3 Conclusions Regarding the R-L Monitor System

The R-L Monitor System attempted to fulfill the needs of many users for several years with emphasis on ease of programming (lack of system interference with programming thought and effort). (Editor's note: if this were the DEC paper-tape editor I might have to think - let's see, was that line six lines ago, or was it 7?, I don't remember, I'll have to list it to make sure!) The System recognized the editing and debugging phases of program development, not jjust its binary usage (once a program works it doesn't matter which system runs it, as long as it takes under an hour to load in the binary.). ("Paper tape is bad for you!" John C. Green, 1971)

1.7 Modern History (A. M.) (After Mouth)

In 1970 the P?S sent out a small, but mighty, band of hackers to fight the bugs and limitations inherent in the R-L monitor. During the fierce battles that ensued for the next 1.5 years, the P?S lost a few members to outside forces (marriage, the draft, DEC, science fiction writing, other schools, working, insanity, beachcombing, general bumming around, IBM, bubonic Plago [PLAGO - A translator-interpreter for a PL/I subset. "PLAGO/360 User's Manual", Poly Inst Broollyn.], etc.), but a few persisted and were able to overcome some of the problems. During this time, several important results were achieved:

1. The System was made to work on a DF32 disk which foresaw the feasibility of using other devices for the System Device such as PDP-12 Linctape, RF08 disk, RK08 disk pack, magnetic tape, and the, as of then unheard of, TD8/E simple Dectape.

2. Several subsystems were made to run without EAE.

3. Several subsystems weree added to the System (ex: FOcal).

4. Multi-unit support for the System was added (SYs N).

5. Improvements to PAL III were made, such as binary output files.

6. Revolutionary tape routines were written:

    1. For the binary loader, a smaller tape search algorithm was discovered, which relied on two factors which were peculiar to the binary loader (not general purpose), namely:

        1. Lack of need to search for block zero and thus incur an end error.

        2. Standard format tapes have block numbers with 11 bits significence, leaving the sign bit for available for other tests.

        This allowed the binary loader to use the standard TC01 bootstrap convention for safety and compatibility. It also lifted the 5 binary file restriction to 13 (generally other sub-systems such as PAL III allow 15). As of 1975, this was raised to 17 for most system programs (BIN/I is restricted to 16 files).

    2. Support for extended memory became possible although it could not be implemented in PAL III. (internally, PAL III is tightly coded; it is nearly impossible to modify it as issued by DEC; bear in mind that the R-L subsystem known as PAl is an overlay to the DEC paper-tape!) The FIELD pseudo-op could not be modified and was considered worthless in-house for a 4K machine anyway. (This was before the P?S/8 PAL assembler.)

    3. The size of the "one page" Dectape handler subroutine used in many PDP-8 programs was reduced to the point that it was possible to consider a permanently resident handler in page 7600 with enough free words to pass all system information as before, in the "system loader", along with support for new parameters (switches), etc., as well as the support for multiple units and extended memory. This version was supplanted by a more efficient one which was then optimized by Richard Lary himself, assuring its reliability. This newer version was then optimized for use in page 7600 only, occupying 7600-7753 with several holes available at run-time for switches, etc., thus rendering obsolete the former system loader program. In all prior R-L subsystems, and internal handler was necessary for each system program. This also implied that custom versions of each program were needed to support alternate system devices, a crude restriction indeed. The R-L Monitor itself required a custom system loader to bring into memory the image of any system program (complete with its own internal handler) leaving behind in page 7600 a bootstrap to reload the system (preferably TC01 compatible if Dectape), along with all pased parameters as above. The permanently resident handler concept would eliminate all of this overhead! As of 1982, these routines have been optimized more than 15 times to create the current standard definition of system handler used by all P?S/8 systems in all cases. Regarding TC01/TC08 Dectape operation, these routines have a unique

feature, in that they remember the correct initial search directive when starting up a tape search. Well over 20 system device handlers have been written for all DEC standard devices as well as some non-DEC ones.

7. A new assembler (P?S PAL) was written to replace PAL III.

8. The DUp command was rewritten to allow for multiple tape drive support. It was later dropped in favor of a general purpose SYSCPY program, which supported extra buffer memory, as well as multiple drives. It also allows optional verfication of the copy.

9. The BIN sub-system was totally rewritten to have full compatibility with all loaders. The TC01 bootstrap (or any other device dependent equivalent) would now be co-resident with the equivalent of the DEC bin and rim loaders when the user program was in control of the machine. BIN contains utlities to deal with external paper-tape images of programs in either bin or rim format. Options exist to cause page 7600 to contain the system I/O request, but this is not forced upon the user.

10. The development of smarter and smarter manual bootstraps produced a tiny bootstrap for the TC01 system only three instructions [48 bits (six bytes)] long! (also requires one constant and one word to be in a large range with few unacceptable values.)

11. Other bootstraps were writteen for special purposees (non-overlaying, shortest starting at 7600, shortest in the highest address in the machine as small as the rim loader, etc.).

1.8 Recent Work on P?S/8

Rcently, a complete rewrite of the keyboard monitor occurred, which remained externally compatible in most respects with the original, but was internally coded more efficiently. Since the new version didn't have to evolve, it was more internally consistent and allowed variant features, a process still occuring today, as of 1982.

Many new system programs now exist; many more are planned in the near future. The P?S PAL assembler is now a proper superset of both PAL8 and PAL10 (Also DIAL10) (disreggars some nit-picking differences). The system MAP program is now a chain-able option of P?S PAL, etc. The reader is referred to more recent specifications of the P?S/8 System for more recent specifications of the P?S/8 System for more details.

Return to my home page.

Page last updated 29-Dec-2005