PDP-1 COMPUTER
ELECTRICAL ENGINEERING DEPARTMENT
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
CAMBRIDGE, MASSACHUSETTS 02139

PDP-49

INTRODUCTION TO THE PDP-1 TIME-SHARING SYSTEM

September 22, 1971

Introduction to the PDP-1 Time-sharing System

Introduction

This is written for the purpose of giving the beginning PDP-1 user a brief overview of the time-sharing system and enough information, condensed into one short memo, to enable him to sit at a console for the first time and, using a simple subset of the available commands, be introduced to some of the system software features. However, do not be misled by the simplicity of this memo. The PDP-1 time-sharing system has many powerful and sophisticated features.

General Information

The present time-sharing system has five consoles, three of which are permanently located in room 26-260 and are generally available. The user may have up to six fields (4096 18 bit words each) of virtual core memory. The backup storage is a drum, segmented into fields each of which has the capacity to store 4096 18 bit words. There are four microtape (DECtape) transports for storing programs on microtape. Each microtape holds ~~512~~ 578 blocks of 256 words.

Peripheral equipment includes a scope display (1024x1024 points), light pen, high speed paper tape reader and punch, 30 inch Calcomp plotter, 8 bit A/D and D/A converter, provision for music output (stereo), and sets of knobs, buttons and switches for interactive programs. Off-line Flexowriters are available for listing programs that have been punched out on paper tape.

The current list of (extended) documented languages is Certainly (assembler), Fortran IV, Lisp, TR3AC (a version of TRAC), PAL, and a music compiler. Others are in development.

The system utility programs include Invisible Debugger (ID), Expensive Typewriter (ET - text editor) and the File System which resides on microtape.

The Instruction Manual is available in several parts. Part 1, Basic Instructions, and Part 2, Address Calculation, are essential for machine language programming. Other parts explain the use of I/O devices and the more sophisticated system features.

Console Control Panel

Each typewriter has a control panel associated with it that includes the following switches: Console On (press down, locking), Call Button (press down, momentary), Display Lever (press down, momentary), Stop Print (normally in the up position), and six Sense Switches. Since some of the utility programs provide options depending on the position of the Sense Switches, it is safest to start with all of them in the off (down) position. In addition the panel has two indicator lights. The Run light indicates when the console is running, i.e., the program is executing instructions. The Type light indicates when it is permissible to type in characters, i.e., the input buffer is not full and the typewriter is not hanging up on a type-out.

Console Turn On (Log-in Procedure)

Users are expected to first sign the PDP-1 log, giving their names and the time of day.

Find an unused console. A console is unused if its Console On switch is off. To turn it on, press down the Console On switch on the Console Control Panel. The typewriter should type out a down shift, carriage return and the console Type light should come on. If the above does not happen, check the typewriter ac power switch (under right hand side), check the Stop Print switch (normally up). Try typing a carriage return on the typewrit The same lack of response may also be evidence of a lack of system resources available for the console (i.e. the system is overloaded).

At this time the system has assigned to the console an ID (explained later) and a core memory field where the user's program is executed.

The utility program Invisible Debugger (ID) currently serves as the monitor program so at this point ID is in control and is "listening" for input commands from the typewriter.

Whenever it is desired to interrupt a running program, just press the Call Button. The program will be stopped and ID brought into control.

To log-out, turn the Console on switch off (up) and sign off in the log, indicating the time. Remember that when you log out, everything in the system belonging to your console is flushed and may not be readily recovered.

Expensive Typewriter (ET), see memo PDP-22

ET is the text editor which enables one to initially type in and easily modify a source program. ET is called by typing an upper case E to ID, (also entered automatically in some of the File System operations).

ET has two modes. In control mode all characters typed in are interpreted as commands to ET. In text mode all characters typed in are stored in ET's text buffer. In control mode the typewriter prints in red, and in text mode it prints in black.

Some of the commonly used commands are explained here. Complete information may be found in memo PDP-22.

Command	Action
a	(append) Text mode is entered and all succeeding typewriter input will be stored at the end of the current page of the text buffer.
backspace	In text mode - cancels the previously typed character. In control mode - types out the next line. To change from text mode back to control mode, type carriage return immediately followed by a backspace.
.	(period) A special character which may be used in place of the current line number.
w	(write) All of the current page is typed out in black.
nl	(line) Line n is typed out in red.
nd	(delete) Line n is deleted.
n,mc	(change) Lines n through m are replaced by the typed input following the c command.
ni	(insert) Typed text is inserted before line n.
k	(kill) The current page is deleted.
P	(punch) The entire buffer is punched out on paper tape.
b	(back) Control is transferred back to ID.

All of the above lower case commands require a carriage return to initiate the action. Typing a centerdot instead of a carriage return will cancel the command.

File System, see memo PDP-42

There are community microtapes where the user may file the source copy of his program. It is recommended demanded that the user's name be ~~included in~~ the first part of the file name. Please delete files when they are no longer needed so that the space may be reused.

The microtape reel is mounted, name (number) side out, on the left hand hub of the transport. Dress the tape over the heads and wind at least one and a half turns clockwise on to the right hand reel. Press down on the Auto/Run switch and hold down the Forward switch for several seconds to allow some tape to wind onto the right hand reel.

At this point, lift up on the Auto/Run switch to the Auto position. The Auto indicator light above the switch will come on. Tapes may be safely read, free of operator hazards, since the write permit has not yet been turned on.

The microtape transports are labeled numbers 0, 1, 2, 3. To enter the file system from ET or ID type nF, where n is the number 0, 1, 2, or 3. Wait for the tape to stop spinning and the typewriter to type out a carriage return.

You are now in the file system and it is listening for commands, which are handled in much the same manner as in ET with a carriage return necessary to start the action. Some of the commonly used commands are explained here. Complete information may be found in PDP-42.

Command Action

p,

(print) Prints out the directory of all the files stored on the tape.

e filename

(edit) Reads the text file titled "filename" into the user's text buffer and starts Expensive Typewriter.

l filename

(loadgo) Reads the binary file titled "filename" into the user's core memory field and starts executing it.

n filename

(nightmare) Reads in a source program titled "filename". It is passed to ET and then to the assembler (Certainly), assembled, and a copy of the object program put into the user's core. Control is returned to ID, where the program may be started by typing an upper case P.

To file information on microtape it is necessary to turn on write permit, i.e., lift up on the Write/Stop switch. The Write indicator light will come on. Then the text, or source copy, of your program may be filed with the command,

f filename

(file) Copies the contents of ET's text buffer onto tape under the title "filename". Wait for the typewriter to do a carriage return, then type b (back) to the file system to return control to ID and wait for the file system to complete, the tape to stop spinning, and a carriage return to be typed out. Do not use the call button to leave the file system.

WARNING: Never turn the console off or take off a microtape until you have left the file system.

To take down the tape, press down on the Write/Stop switch which will take the control out of Auto (Auto light off), press down on the Auto/Run switch and hold down the Reverse switch until the tape rewinds. Then press the Stop switch and stop the coasting tape by hand.

Invisible Debugger (ID), memo PDP-23

ID is a utility program to aid in the debugging of programs. ID oversees execution of all the user's programs. If the program executes any illegal instruction or if the console's Call Button is pressed, control is returned to ID. As implied by the title, Invisible Debugger, ID does not exist in the user's core memory field, and cannot be clobbered.

ID is most often used to examine and modify registers, to assign devices, and to start a program.

Some sample commands are,

Command — Action

adr/

The contents of location adr, which may be either an octal number or a symbol if it has been defined, are typed out. At this point the register adr is said to be open and new contents may be typed in and the register closed by typing a carriage return, thus modifying the register. In this manner a short program could be typed in, such as a short sequence to check the action of a set of instuctions.

A

The address of the user's saved AC register. For example, typing A/ will cause the contents of the AC to be printed.

adrG

(go) The user's program is started at location adr.

adrB

Places a "breakpoint" at location adr. If the user's program attempts to execute the instruction at that location, the program will be stopped and ID brought into control. At that point, the user may examine locations to determine if the program is behaving properly.

P

(proceed) If a program was stopped by hitting the Call Button, or if it reached a breakpoint, you may proceed from that point by typing an upper case P.

The above is a bare minimum of the commands and features available to the user in ID. Features include multiple breakpoints, various typeout and typein modes, search commands, and process control, all explained in the ID memo (PDP-23).

Certainly (assembler), see memo PDP-45

Certainly is used to assemble machine language programs, generally typed in using Expensive Typewriter (ET), into object (binary) programs which are run under control of ID.

Refer to the Instruction List memo PDP-35, at least part 1 of it, and following the format on page 1 of the Certainly memo, a program example may be typed in under control of ET.

Certainly is called by typing an upper case N to ET. The source program in ET's text buffer will be assembled and brought into the user's core memory field under control of ID, ready to be started with a P (proceed) command. ID will have been given a copy of the symbol table listing all symbols that the user may have defined. There will also be a copy of the binary program on the user's drum field 101 which may be loaded by ID.

If the source program fails to assemble correctly, Certainly will type out appropriate error comments. Type an E to ID, update the source program, type N to ET and it will start another assembly.

Likewise, if the object program fails to do the right thing, hit Call, you will be in ID, type E, you will be in ET and your text will still be there.

Fortran IV, see memo PDP-48

Programs are typed in using ET. The Fortran compiler resides on the microtape labeled Fortran. With that tape mounted on microtape transport n, typing nF to ET or ID will start the compilation of the text in ET's buffer, finishing options ! with the object program loaded in the user's core field, under control of ID, ready to start the program when a P (proceed) is typed.

If the program fails to run as expected, hitting Call will return control to ID where typing E will return control to ET and the facility to correct the source program.

Check the copy of the ~~FORTRAN~~ Relocatable Subroutine Library memo in the computer room for up-to-date information on available subroutines.

Lisp, see memo PDP-36

The Lisp program resides on microtape number 0. Mount the tape on microtape transport n, and type nF to ET or ID. When the tape stops spinning and the typewriter types out a carriage return, you will be in the file system. Type

```
l lisp pprint,15
```

to the file system and a binary copy of Lisp will be loaded into your core memory field and onto drum field 115. The program is automaticaly started and waits for the user to type in the number of core memory fields required. Type a carriage return unless you will need a lot of free storage. Lisp should then be in control mode, will type out a red shift and a minus sign and wait for control characters to be typed. The first character specifies input to be taken from typewriter (t) or Expensive Typewriter's text buffer (s). The second character specifies the output destination, usually the typewriter (t). The Lisp memo (PDP-36) explains the above control commands more fully, as well as giving a complete description of the Lisp language.

It is probably most convenient to type in the expressions using Expensive Typewriter. In this way the source text may be filed and later easily edited.

TR3AC

The TR3AC program resides on microtape number 0. Enter the file system and type

```
l tr3ac
```

and the TR3AC program will be started. The meta-character is single quote ('). A description of TRAC is available in the notebook in the machine room.

PAL (Pedagogic Algorithmic Language)

~~Check the notice on the bulletin board for the current status and operation of the program. Notes from course 6.231 describing the PAL language are available in the machine room.~~

PAL is a language developed by Arthur Evans for use in the subject 6.231 (no longer offered). PAL has many features in common with ALGOL. Its data types include integers, reals, strings, functions, (including recursive functions) truthvalues, tuples (arrays), and labels.

It has extensive error-checking facilities.

The PDP-1 implementation features infinite precision [redacted] integers and arbitrarily large precision reals.

The 6.231 course notes, describing PAL, are available in the machine room. Questions about PAL will be answered by qualified students.

10. Preparing a Tape for Use With the File System

These procedures, while not difficult, should nonetheless be performed only by users who are very sure of what they are doing.

The file system is stored as a binary program on public tape pdp-0, under the file name "filesys". This version is always kept up to date with the latest memos and system changes.

The normal use of this program will be to replace the file system on tapes that already have a directory and files on them. Load the file system program into core 0 and start it at location 0. It will ask for the tape drive number, and then allow the tape name to be changed. If no change is desired, type a carriage return when it asks for the new name. The file system will be replaced on the tape, and all files will be saved.

If the file system is to be put on a new tape that has never before been used with a file sytem, that tape should have its timing and mark tracks written and a tape test run before getting the file system. When the tape is properly marked, load the file system program into core 0 and start it at location 1. The tape drive must have been previously assigned, and the program will ask for verification of the drive number. The procedure from there is the same as for the entry at 0, except that a completely empty directory will be put on the tape. If the tape had previously contained files and a directory, they will be lost.