The PDS Control System:
Software structure and User Interface

Paolo Marcucci

Pubblicazione Osservatorio Astronomico di Trieste no. 1343, 1990

Table of Contents

Introduction . . . 3
Requirements . . . 3
Performance . . . 3
Usability . . . 4
Reliability . . . 4
Methods of implementation . . . 4
Hardware . . . 4
Programming environment . . . 4
Programming language . . . 5
Layers . . . 5
User interface . . . 8
Dialog boxes . . . 8
Function keys . . . 9
Menus . . . 10
Function keys . . . 10
Help . . . 11
Function keys . . . 11
Application . . . 12
Structure of application . . . 12
Session, log and batch files . . . 13
Working sets . . . 14
Starting the application . . . 16
Format . . . 17
Scan . . . 18
Interactive . . . 20
Batch . . . 22
Batch file editing . . . 24
Utilities . . . 26
File management . . . 26
Color settings . . . 27
Off-line utilities . . . 28
Data structure . . . 29
DMA usage . . . 29
Shared Data Base . . . 30
Mass storage . . . 30
Image file format . . . 31
Remote storage . . . 31
Software interfaces . . . 32
Stage movement . . . 32
Lock control . . . 32

1. Introduction

The PDS Control Program is the software part of a complex system designed in order to control all the functionalities of a Perkin-Elmer PDS 1010A microdensitometer.

This instrument is currently used at the Trieste Observatory for the digitization of astronomical photographic plates. It can be used as well for every kind of work that needs digitization or scanning, as computer-based storage of images in the medical field.

The software that actually controls the instrument was provided by Perkin Elmer along with the PDS machine. It has some bottlenecks, as the storage of the scanned data in records of 3919 points and the unfriendly user interface, based on a teletype.

To be more specific, there were 5 main defects in the software:

- unfriendly user interface;
- very little data buffer;
- impossibility to perform a data check prior to storage;
- unvailability of a macro-like control language;
- no error checking on the tape subsystem.

The need for a better software arised quickly, but also the hardware had to be adjusted. The modifications to the instrument hardware took place in the area of plate runout error recovering, fine motor control and the replacement of the main control computer.

2. Requirements

The basic requirements of the system have been defined either by the designers and by the users.

2.1. Performance

The system should perform scan operations at the maximum allowed speed, e.g. 255 arbitrary units (vs. 10/20 in the old system), corresponding to 1720 microns/sec, should use a greater buffer to store the scan data, up to 32767 points per scan (vs. 3919 in the old system), and a faster mass memory storage device (16 ms acces time, 40 Mbytes capacity hard disk vs. 1200 bps tape).

2.2. Usability

The system should use an up to date user interface, based on pop-up menus and dialog boxes (vs. TTY cryptic input/output in the old system). An help system should be at the user's hand in every moment, context sensitive and hypertext based (there wasn't an help feature in the old system). All operations should be full controlled by the user.

2.3. Reliability

All operations should be error proof (in the human limits), and an error in the disk write procedure (for example) should not halt the system.

All of these requirements have been reached.

3. Methods of implementation

3.1. Hardware

The new control program is implemented on the new control hardware composed by a personal computer, an Olivetti M240 with a 40 megabyte hard disk, and a series of interface cards.

The choice of a personal computer was made taking in account the ease of maintenance, the cheap expandibility, the operating system solidity and the choice between various languages and software tools.

The Olivetti M240, an XT compatible machine, may seem a bit low in performance, but the tests showed that the overall speed of the system is quite acceptable. The use of a more powerful AT compatible (with 80286, 80386 or 80486 CPU) computer is not allowed by software conflicts between the peculiar AT hardware and the interface cards.

3.2. Programming environment

A personal computer doesn't support many operating systems. The typical choice is between MS-DOS and XENIX, but only MS-DOS is a reasonable choice in this case. XENIX provides UNIX compatibility, native network support and a lot of compilers and tools, but those features overload the 10 Mhz 8086 CPU in a way that the basic performance requirement cannot be reached.

MS-DOS doesn't provide native network capabilities, but these can be added via the use of third party software. In our case, the network software is the NEXOS local area network operating system plus the TCP/IP extensions by FTP software.

3.3. Programming language

The MS-DOS operating system supports literally thousands of languages and developement tools, but only a few are suited for professional use. Our developement environment have to support code reutilization, information hiding, highly structured programs and superior performance in term of speed and code compactness. These strict requirements thighten the arena to a very few languages. C and Pascal are two of them.

C is becoming a standard programming language for every kind of computing machine. Good MS-DOS implementations of it are provided by Microsoft and Borland, the first with his C 5.0 compiler and the second with the Turbo C 2.0 developement system.

Pascal is a very good programming language that have only one serious implementation in the MS-DOS environment: the Borland's Turbo Pascal 5.5. This implementation is one of the first OOP (Object Oriented Programming) languages available on personal computers, and OOP is what the PDS control program needs.

This programming paradigm highlights on our requirement characteristics: information hiding is perfectly accomplished and code reutilization is obtained building object classes whose properties and data are inherited by derived subclasses.

Our choice, now it's clear, was for the Turbo Pascal 5.5.

4. Layers

The control program consist in a single huge executable file, composed by a main routine and several libraries and modules that interact among them via a common data area (see figure 1).

It is logically divided in four layers:

- the user interface (UIF) layer, that is what the user really see of the program and of its interaction mechanisms;
- the application layer, that is the main body of the program;
- the data layer, that is where all the data are stored;
- the software interface layer, that is the layer directly in communication with the hardware.

Two other layers exist: the hardware interface layer and the instrument itself. Those layers are completely based on hardware, so in this paper we will cover only the first four layers.

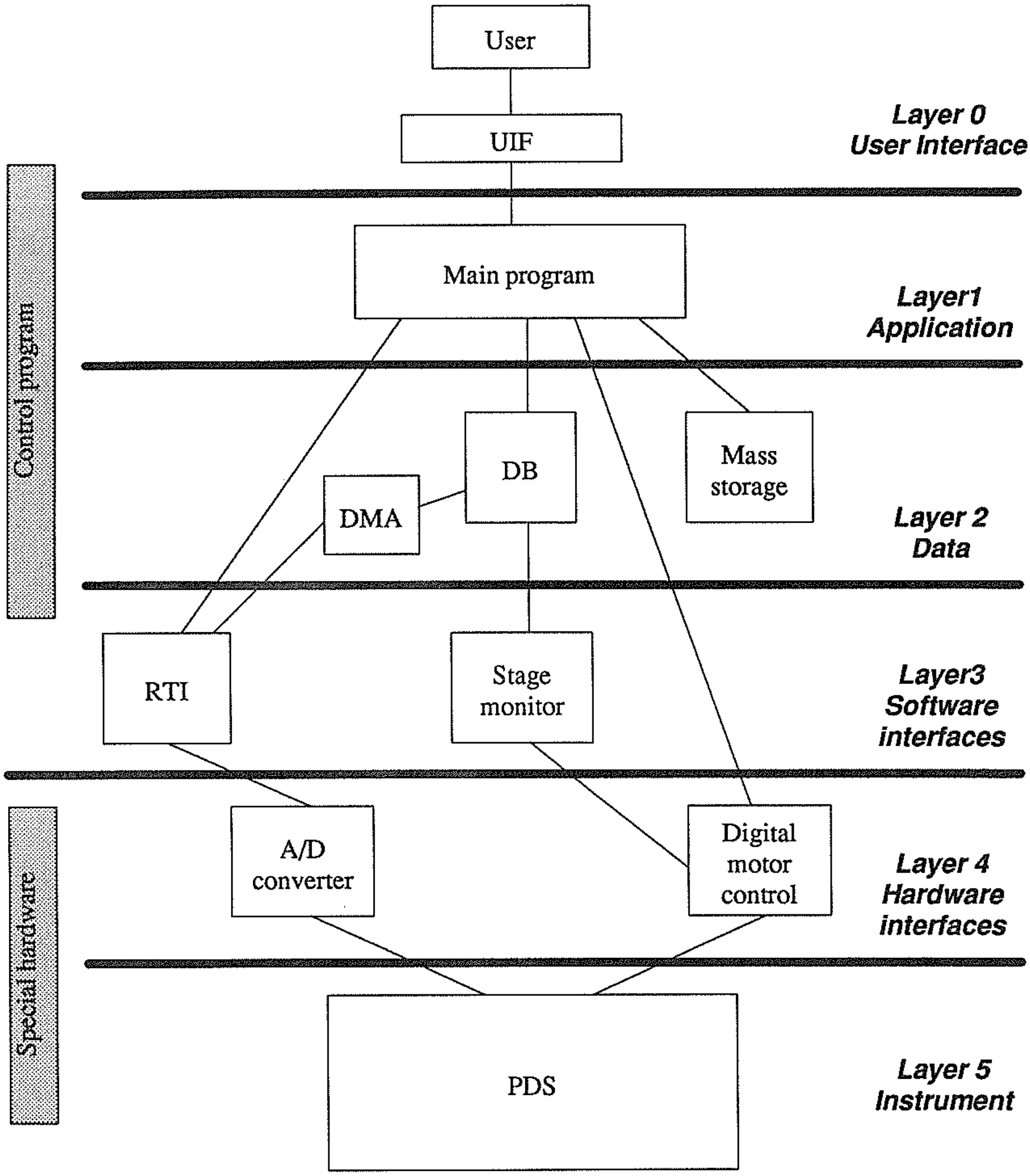

Fig. 1 - Overall structure of the PDS control system

5. User interface

The user interface is built around a set of objects, like menus or dialog boxes, that preserve common characteristic and features. This means, for example, that the F1 function key will bring up an help screen in every situation.

All these objects are linked toghether via a "function tree" (figure 2), where the root is the main menu, and the branches are submenus and dialog boxes.

The user can roam through the function tree freely, using a litte set of functional keys that keep their means throughout the whole system.

5.1. Dialog boxes

These objects appear to the user as a shadowed box that provides an interaction with the program.

There are two main type of dialog boxes:

- MASK - a window containing a series of input/output fields that can be freely edited;
- COMMUNICATION - a window presenting a request to which a user's reply is needed.

These two types can be distinguished at a glance by the color in which they are displayed on screen: gray background for the first type, red background for the second one.

As an example, figure 3 shows the dialog box of the working set, that lets the user to modify all the parameters composing a set of working parameters, presenting the physical value of the speed in real time and giving the user a full featured form editing.

The dialog box for choosing the kind of operation to insert in a batch file, instead, simply asks the

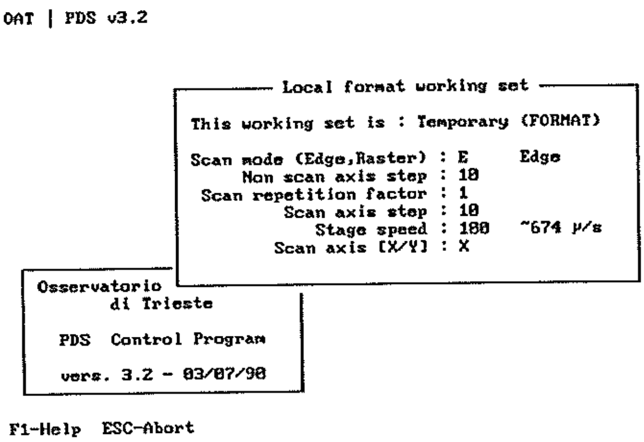

Fig. 3 : Temporary working set dialog box

user for a keypress, V or O depending on the operation to insert (figure 4).

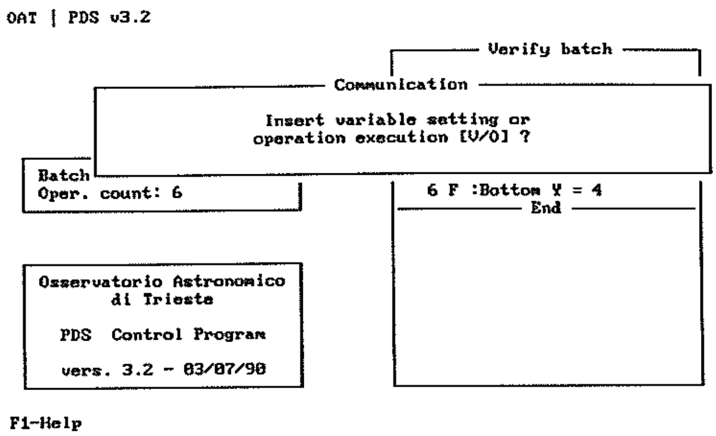

Fig. 4 : Operation insertion dialog box

5.1.1 Function keys

Function keys provide full control over the management of a dialog box. The keys accepted in every context are the up and down arrow keys, the Enter key, the Esc key and the F1 key. In addition to these, also left and right arrow keys, Backspace and Home keys are allowed for dialog box editing.

KEY	MASK	COMMUNICATION
Up arrow	Previous field	unused
Down arrow	Next field	unused
Left arrow	Enter edit mode & move cursor	unused
Right arrow	Move cursor in edit mode	unused
Enter	Next field or exit mask	unused
Esc	Abort mask reading	unused
Home	Accept all fields	unused
Backspace	Correct field	unused
F1	Help	Help

Table 1 : Function keys usage in dialog boxes

5.2. Menus

Menus are a fast way to choose one item from a list of options. A typical menu is shown in figure 5. Here a list of all the variables that can be used in batch programming is presented and the user can choose among them the one to use. This is done via a set of function keys described in table 2.

Menus can be "multi-level". This means that a menu selection doesn't perform an action but presents another menu of a lower level. Navigation throughout the menu system is done with the Enter and Esc keys, respectively to access an to leave a menu.

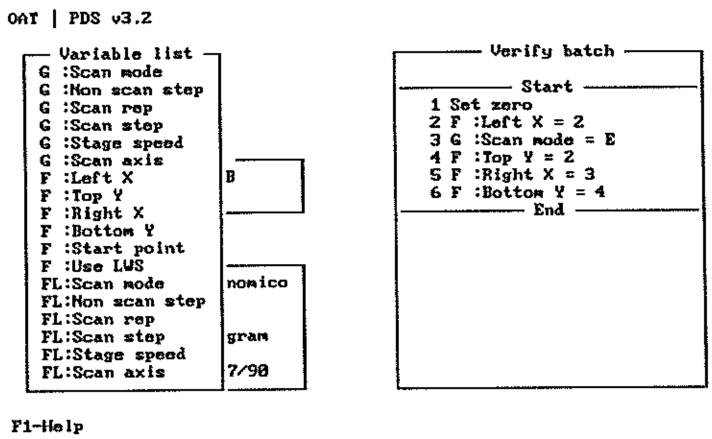

Fig. 5 : The menu for the selection of a variable

5.2.1 Function keys

KEY	MENU
Up arrow	Previous item
Down arrow	Next item
Left arrow	unused
Right arrow	unused
Enter	Select item
Esc	Abort menu selection
Home	Accept all fields
Backspace	unused
F1	Help

Table 2 : Function keys usage in menus

5.3. Help

In the older version of the PDS software, there wasn't an help facility, so that only skilled users was able to run the program. In the current version this point is particularly stressed. An help window is always available, giving information on the operation carried out by the user.

The help system is context sensitive and hypertext based. This means that the help text proposed is relative to the operation the user is performing (context sensitive) and the navigation throughout the help screens doesn't follow any fixed sequence, but the user can view a particular help screen using a method of "linked concepts" (hypertext based). For example: if the user requests help about the MOVE STAGE operation, an help screen is presented on the video terminal. Then, the user can see other help screens related (operations index, move horizontal, move vertical) simply by moving the cursor on the desired topic and pressing Enter. The help tree is quite complex and can be edited off-line to add new details and suggestions or to translate the screens in another language.

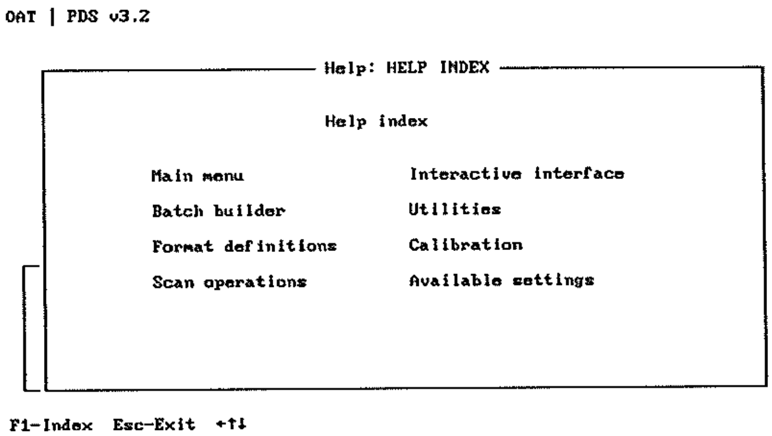

Fig. 6 : The help system main index

5.3.1 Function keys

KEY	HELP
Up arrow	Previous topic
Down arrow	Next topic
Left arrow	Previuos topic
Right arrow	Next topic
Enter	Select topic
Esc	Exit help system
Alt+F1	Back to previous screen
F1	Goto main index

Table 3 : Function keys usage in help system

6. Application

The application layer is the body of the PDS control program; it is not visible to the user and the only way to interact with it is via the user interface.

This layer is devoted to:

- the management of direct commands given by the user;
- the management of batch files;
- the logging of the work sessions;

As the user interface, this layer is written in Turbo Pascal 5.5, and it is part of the main program file.

6.1. Structure of application

The structure of this layer is represented in figure 7.

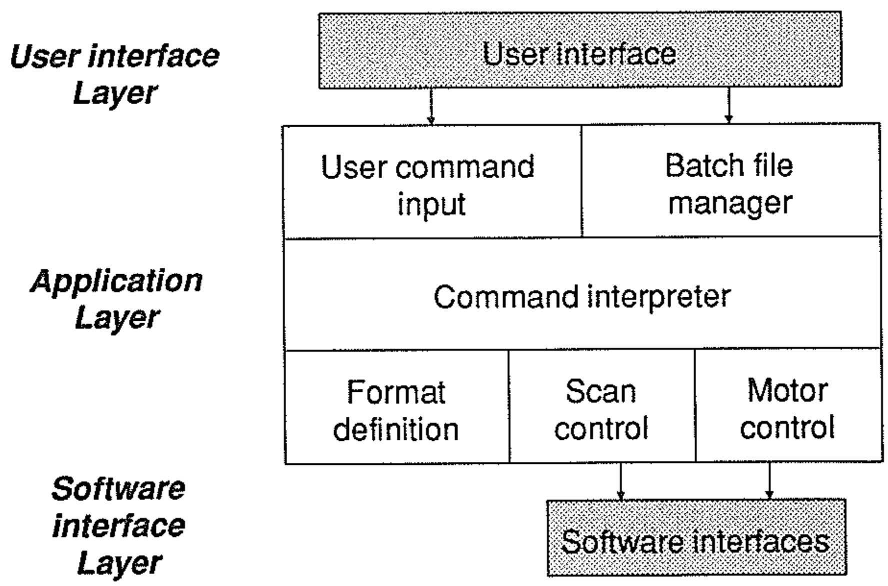

Fig. 7 - Application layer structure

Here we can see that the input to the layer comes either from direct user commands or from a batch file created using the BATCH menu tools.

6.2. Session, log and batch files

When the user log in, the program checks for the existance of a session file. If it is present, this means that the program was not correctly exited the last time the user runs it.

This procedure may seem unuseful until we think of a large scan procedure that collect several scans on a large plate. This scan procedure can take many hours to complete, so the user can execute it and leave the PDS do the work unattended. If during this time, late at night, the power fails, the system doesn't shut down correctly and the session file is not erased. In this file are stored all the operations completed until that moment, in a way that the program can restart the scan procedure from the last command executed.

The session file records all the user direct commands and the batch manager commands, and, when the user wants to exit from the PDS control program, asks if the user wants to save it in a log file or cancel it.

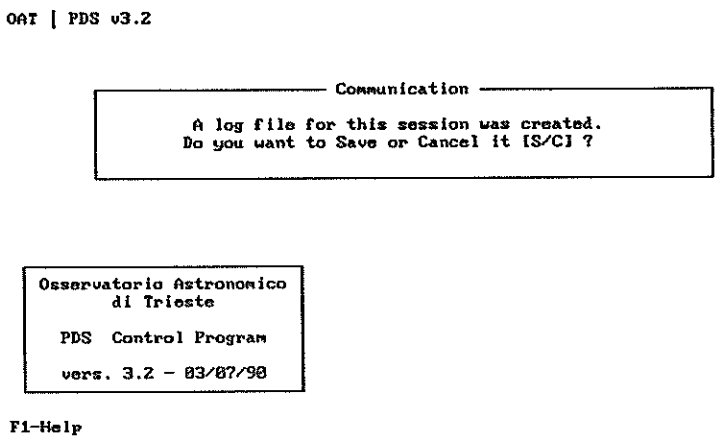

Fig. 8 : Saving of a log file

Log files are copies of previous session files and can be examined either within the program (via the FILE MANAGER in the UTILITIES menu) or off-line by the SHOWLOG program. This is very useful for documentation purposes.

Batch files are not exactly the same thing. As the session and log files, they record all the operations executed by the program, but in a way that can be interactively post-edited by the user. A batch file can start at any time of the session, and end before the completion of it. These files works well for the replay of frequently used sets of operations (something like subroutines).

A complete description of all the capabilities of batch programming can be found later in section 6.7.

6.3. Working sets

A working set is the group of parameters that the user can define in order to perform a SCAN operation. These parameters include scan speed, scan axis, start and stop points, etc...

Four working sets are available to the user:

- DEFAULT - the default working set, preprogrammed at the start of the session. Its values are hard-wired in the program code.

- GLOBAL - the global woking set, it replaces at any time the DEFAULT and it used by the program whenever a particular working set in not defined.

- FORMAT - the local format working set, replaces the GLOBAL and is used during the FORMAT operations.

- SCAN - the local scan working set, replaces the GLOBAL during the SCAN operations.

The definition of the GLOBAL working set is achieved via the FORMAT/GLOBAL WORKING SET menu entry.

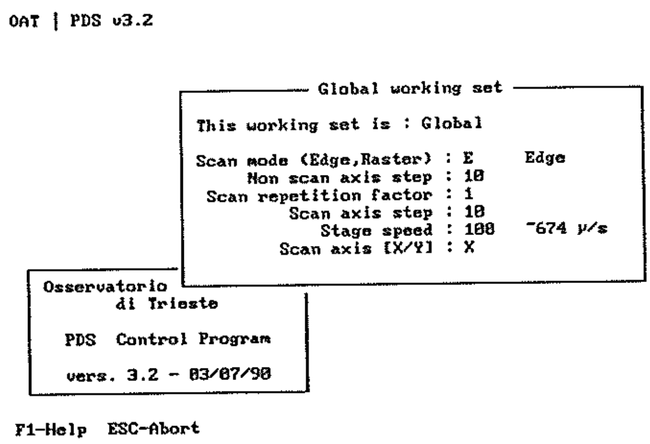

Fig. 9 : Global working set definition

Working sets can be edited via a dialog box presenting six I/O fields. Cursor movements inside this dialog box are described in section 5.1, while the contents of the fields are described below:

- SCAN MODE - it defines the mode by which the stage will scan the image. It can assume two values: E or R (edge or raster). Edge mode moves the stage to the start position, starts the scan line operation and then, without collecting data, returns to the start point; then moves the stage to the next start position and continue scanning. Raster mode, instead, performs the same operations of the Edge mode but, at the end of the scan line operation, moves to the next start point collecting data (see figure 10 for a visual representation).

- NON SCAN AXIS STEP - the distance, in microns, between two scan lines.
- SCAN REPETITION FACTOR - currently not implemented.
- SCAN AXIS STEP - the distance, in microns, between two acquisitions.
- STAGE SPEED - the speed, in arbitrary units, of the stage during the scan operation. This speed is used only during data collecting, while the speed of stage movement between acquisitions is automatically determined by the program. The range allowed is from 1 to 255, that means real speeds from ~6 microns/sec to ~1718 microns/sec.
- SCAN AXIS - is the axis along which the scan line operations are performed. Allowed values are X or Y.

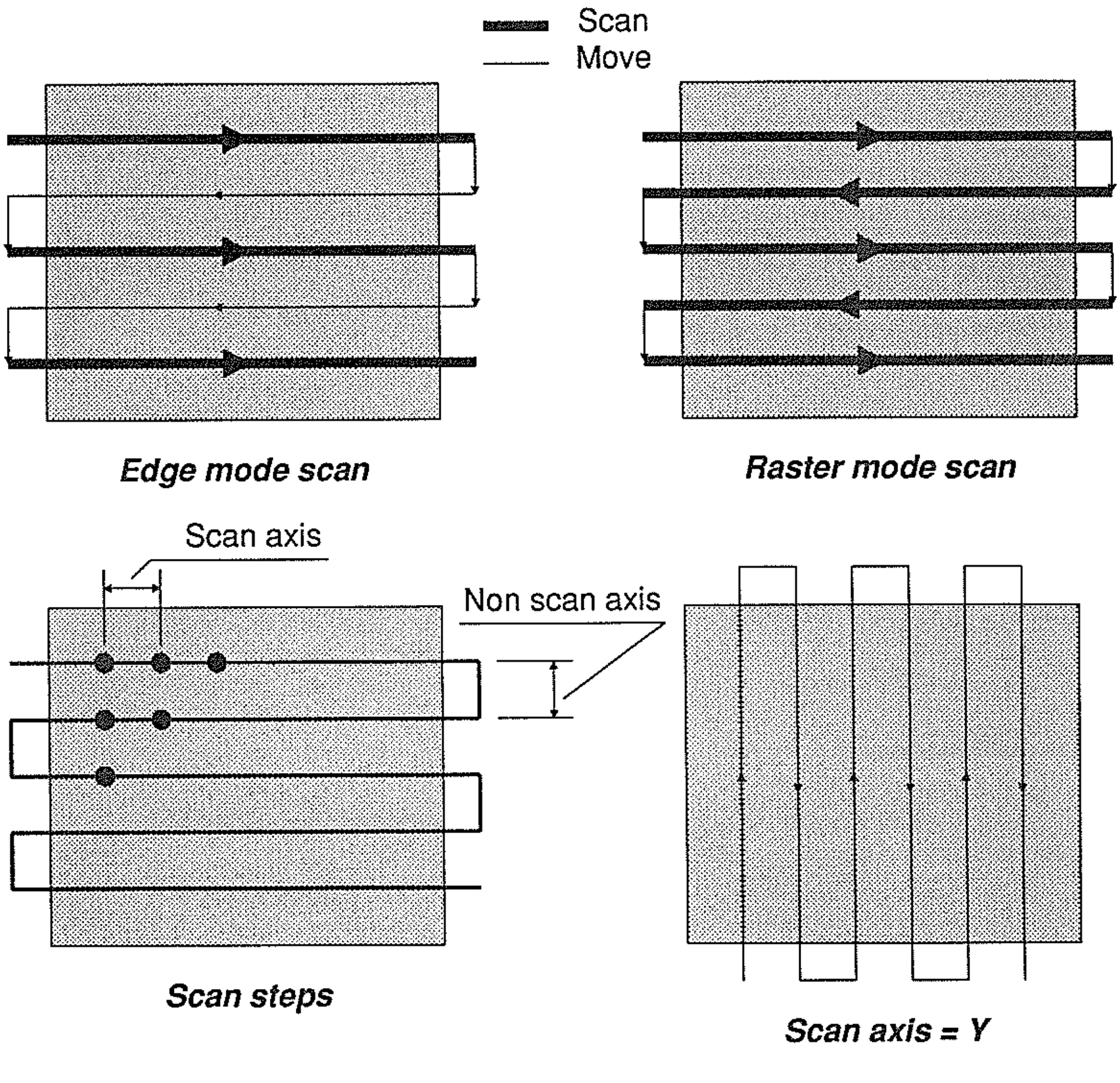

Fig. 10 - Working set items

6.4. Starting the application

The PDS control program is a DOS application so it can be started by the command line via a batch file called PDS. This DOS batch file (not to be confused with the internal PDS batch files) starts two drivers for the DMA management and the A/D converter board and then launch the PDS.EXE program.

The first thing to do for the user is to log into the system. This is accomplished via the first dialog box presented (see figure 11)

```
OAT | PDS v3.2

                                   ------ User identification ------
                                   USERNAME: PAOLO
 Osservatorio Astronomico
       di Trieste                  PASSWORD:

   PDS  Control Program

  vers. 3.2 - 03/07/98
```

Fig. 11 : Login dialog box

After a successful login, the main menu appears (see figure 12). From here on the user can reach all the functionalities of the PDS control program.

```
OAT | PDS v3.2

---- Main Menu ----
Reset system
Batch file manager
Format options
Scan operations
Interactive
Utilities
Quit program

 Osservatorio Astronomico
       di Trieste

   PDS  Control Program

  vers. 3.2 - 03/07/98

F1-Help  ↑↓-Move  ↵-Select
```

Fig. 12 : The PDS control program main menu

6.5. Format

The FORMAT/OPTIONS entry in the main menu lets the user to define the limits of the scan area and the scan starting point (see figure 13). Also the local format working set can be activated via this dialog box. The definition of the local working set is given with a following dialog box (see figure 14).

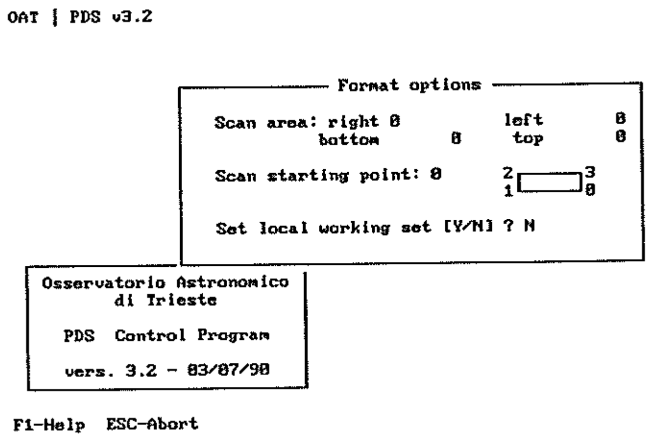

Fig. 13 : Format options

The items that can be edited are:

- SCAN AREA - a series of four values that indicate the limits of the scan area, in microns.
- SCAN STARTING POINT - the point from where the acquisition process will start.

If the SET LOCAL WORKING SET item is set to Y then a local working set will be used during the format definition.

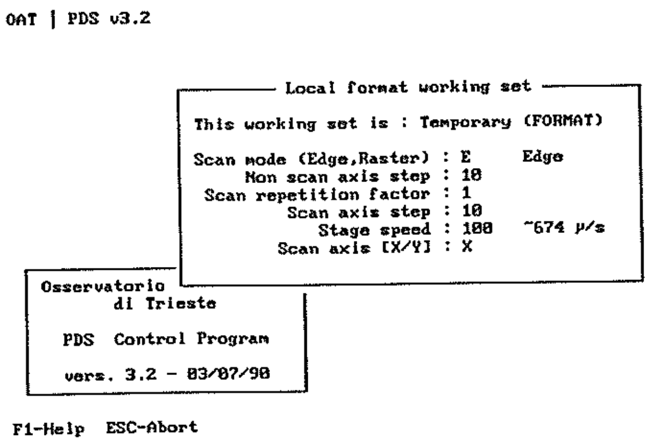

Fig. 14 : Format local working set definition

The items editable in the local format working set dialog box are essentialy the same seen in the global working set items description.

6.6. Scan

The SCAN entry in the main menu takes the user to a submenu (see figure 15). Here four options can be choosed: OPTIONS, START, VIEW IMAGE and PREFERENCES.

```
OAT | PDS v3.2

 ------ Main Menu ------
| Reset system          |
| Batch file manager    |
| Format options        |
| Scan operations     ---- Scan ----
| Interactive        | Options      |
| Utilities          | Start        |
| Quit program       | View image   |
 ------------------- | Preferences  |
                      --------------

 ------------------------
| Osservatorio Astronomico |
|       di Trieste         |
|                          |
|   PDS  Control Program   |
|                          |
|   vers. 3.2 - 03/07/90   |
 ------------------------

F1-Help  ↑↓-Move  ↵-Select
```

Fig. 15 : SCAN submenu

The first entry, OPTIONS, lets the user to define some values related to the scan operation (see figure 16). Here can be defined the starting point of a single line scan, the scan mode (from the beginning or the end of a scan line or using the format scan area values), the output device (screen or file) and the usage of a local working set. Currently only the F scan mode (using the format scan

```
OAT | PDS v3.2

                 ------------ Scan options ------------
                |        Starting point  X : 0          |
                |                        Y :       0    |
                |                                       |
                |        Scan mode [B/E/F] : F          |
                |                                       |
                |     Output [Screen/File] : F          |
                |                                       |
                | Set local working set [Y/N] ?  N      |
 ------------------------------------------------------
| Osservatorio |
|    di Trieste      |
|                    |
| PDS  Control Program |
|                      |
| vers. 3.2 - 03/07/90 |
 ----------------------

F1-Help  ESC-Abort
```

Fig. 16 : SCAN options settings

area values) is implemented, and the other two modes can be simulated via an adequate scan area definition. If a local working set is used, its definition follows the guidelines given in the WORKING SET section.

The second entry, START, effectively performs the scan operation following the described order:

- move the stage to the starting point (defined in FORMAT/OPTIONS)
- set the scan speed
- perform a single line scan, presenting the results on the screen (if requested) and saving the results to the mass storage device (if requested)
- move the stage to the next scan line starting point and loop to the previous operation until the end of the scan area is reached

The third entry, VIEW IMAGE, presents on the screen a 16 pseudo colors representation of an image found on the mass storage device. The method for selecting the image file is actually taken from the UTILITIES/FILE MANAGER submenu, so the overall "look and feel" of the application is mantained.

The fourth item, PREFERENCES, lets the user to set two flags (see figure 17). The first determines if the output graph of the scan resulting values have to be scaled or not, the second indicates if an output graph have to be plotted on the screen during the scan operation.

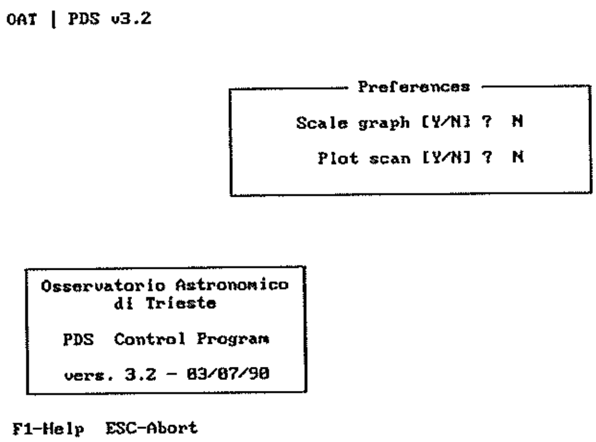

Fig. 17 : PREFERENCES dialog box

6.7. Interactive

Going on in our detailed description of the main menu entries, after the SCAN we find the INTERACTIVE entry. As for SCAN, no direct operations are performed at the selection of this item, instead a submenu is presented (see figure 18). Here the user can choose among five items: MOVE STAGE, READ VALUE, GOTO (0,0), SET (0,0) and FIND (0,0).

```
OAT | PDS v3.2

┌────── Main Menu ──────┐
│Reset system           │
│Batch file manager     │
│Format options         │
│Scan operations        │
│Interactive          ┌─ Interactive ─┐
│Utilities            │Move stage     │
│Quit program         │Read value     │
└─────────────────────│Goto (0,0)     │
                      │Set (0,0)      │
                      │Find (0,0)     │
                      └───────────────┘
┌──────────────────────────────┐
│ Osservatorio Astronomico     │
│        di Trieste            │
│                              │
│   PDS  Control Program       │
│                              │
│   vers. 3.2 - 03/07/98       │
└──────────────────────────────┘

F1-Help  ↑↓-Move  ↵-Select
```

Fig. 18 : INTERACTIVE submenu

The selection of the first item, MOVE STAGE, takes the user to a little dialog box (see figure 19) where can be specified the coordinates, in microns, of the point to be reached by the stage. When the user confirms the two values, the stage will suddenly move to the position selected, using an optimized speed.

```
OAT | PDS v3.2

                                  ┌───── Move stage ─────┐
                                  │ X position : 0       │
                                  │ Y position :        0│
                                  └──────────────────────┘

┌──────────────────────────────┐
│ Osservatorio Astronomico     │
│        di Trieste            │
│                              │
│   PDS  Control Program       │
│                              │
│   vers. 3.2 - 03/07/98       │
└──────────────────────────────┘

F1-Help  ESC-Abort
```

Fig. 19 : Move stage dialog box

The second item, READ VALUE, is currently not implemented and reserved for future use in batch programming (for example it will be used for automatic photometric calibration).

The third item, GOTO (0,0), moves the stage to the (0,0) position using an optimized speed.

The fourth entry, SET (0,0), sets the (0,0) position at the current position of the stage.

The fifth entry, FIND (0,0), starts a search operation of the absolute (0,0) position and, when found, sets the (0,0) position there. This operation uses the absolute zero hardware subsystem to define the position.

All the movements of the stage except for the scan line operation are performed using an optimized speed. This speed is calculated by the software in real time and is the maximum speed allowed in order to prevent excessive mechanical shocks. This speed is not constant during the motion but follows a curve of acceleration, constant speed, deceleration. The two phases of acceleration and deceleration are calculated using a set of predefined "ramps" (see figure 20).

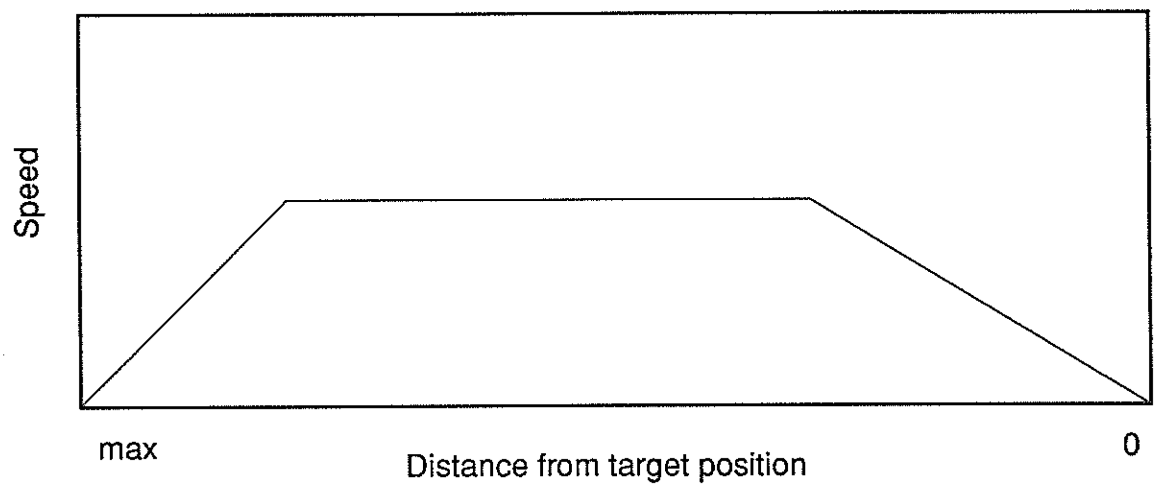

Fig. 20 : Optimized speed graph

The relations between speeds and ramps were calculated using an on-field test, but, when a major mechanical modification will be implemented on the PDS instrument, they can be updated using an external data file. This ASCII file, called PDS.RMP, contains seven lines of data each of the composed by five items separated by commas or blanks. The first item is the speed in arbitrary units (from 1 to 255), and the other items are the related ramps for the various directions of movement. In such a way, the programmer, using the data file, can tune very precisely the acceleration ramps while the predefined ones (see table 4) are unique for all directions.

Index	Speed	Ramp
0	1	0
1	10	100
2	15	200
3	40	700
4	80	3000
5	200	7000
6	255	15000

Table 4 : Predefined speeds and ramps

6.8. Batch

A typical work session is composed by several tasks. One can be the definition of a scan area or the setting of a working set parameter. Every time the user enters a session a different flow of operations will be performed but some "operational blocks" will be every time the same. The PDS control program lets the user to build and mantain batch files, i.e. data files that contains a recording of a series of operations to be performed all together. Batch file operations can be reached via the BATCH main menu entry (see figure 21).

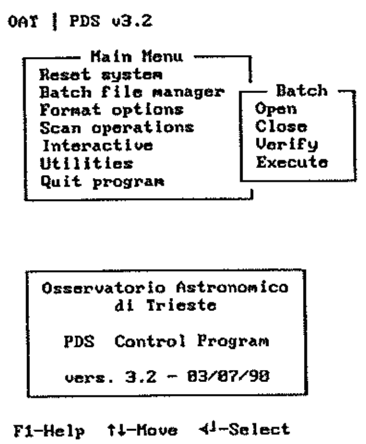

Fig. 21 : BATCH submenu

```
OAT | PDS v3.2

                         ──────── Open batch ────────
                         Enter file name:


     Osservatorio Astronomico
            di Trieste

       PDS  Control Program

       vers. 3.2 - 03/07/90

F1-Help  ESC-Abort
```

Fig. 22 : The OPEN BATCH dialog box

Batch files are created by giving them a name in a dialog box via the BATCH/OPEN menu entry (see figure 22) and closed using the BATCH/CLOSE menu entry. Between this two steps, all the operation execution and variable setting entered by the user at the console keyboard are recorded in the batch file. This can be later replayed using the BATCH/EXECUTE menu entry. This selection takes the user to a menu containing a list of the already present batch files.

The BATCH/VERIFY menu entry can be used either to check the contents of a batch file or to edit its contents; this is achieved presenting a menu containing a list of the already present batch files and, after the user has selected the right one, the operation flow of the batch file in a scrolling window on the right of the console screen and the list of allowed commands on the bottom of the screen (see figure 23).

```
OAT | PDS v3.2
                                     ──────── Verify batch ────────
                                     ────────── Start ──────────
                                       1 Set zero
                                       2 F :Left X = 2
                                       3 G :Scan mode = E
   ─────── Batch status ───────        4 F :Top Y = 2
   Batch name : B1.PDB                 5 F :Right X = 3
   Oper. count: 6                      6 F :Bottom Y = 4
                                     ─────────── End ───────────

     Osservatorio Astronomico
            di Trieste

       PDS  Control Program

       vers. 3.2 - 03/07/90

F1-Help  INS-Insert op.  DEL-Delete op.  ↵-Modify  ESC-Exit
```

Fig. 23 : The BATCH VERIFY screen layout

6.8.1 Batch file editing

The BATCH/VERIFY menu entry is the path to access the editing functions of the PDS control program. After this menu selection the layout of the screen changes as shown in figure 23. On the right we see a scrolling window with the contents of the batch file while on the left stays a small box with the number of operations in the file and its name.

A black bar indicates the current position inside the file and it can be moved with the up and down arrow keys. Some other keys are enabled during this phase. A quick reference table for the allowed ones is shown in table 5, while a discussion on the operations executed follows.

Key	Command
Up arrow	Move selection bar up
Down arrow	Move selection bar down
INS	Insert an operation at the curent position
DEL	Delete the operation at the current position
ENTER	Modify the operation at the current position
F1	Help
ESC	Exit BATCH VERIFY

Table 5 : Batch editing keys

INS - Insert

Pressing the INS key takes the user to a confirm dialog box (see figure 24). If the user press Y, then another dialog box is shown (see figure 25) where he is prompted for the type of operation to insert. The V value permits an insertion of a variable setting choosed from a menu (see figure 26), while the O value permits the insertion of an operation execution as start scan or set zero position (see figure 27). After the selection of the item to insert the scrolling window on the left is updated.

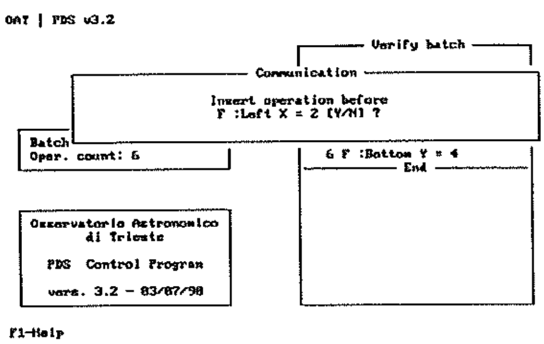

Fig. 24 : Confirm insertion ?

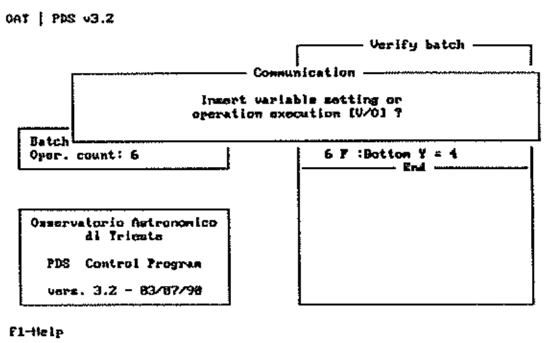

Fig. 25 : Type of operation to insert

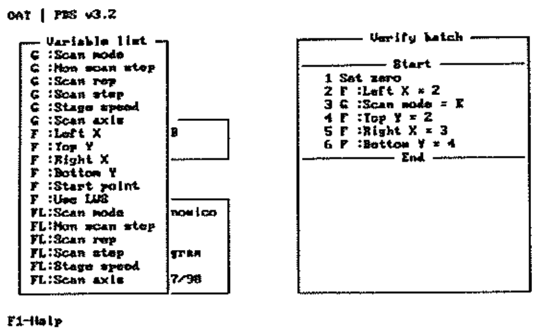

Fig. 26 : Variable setting list

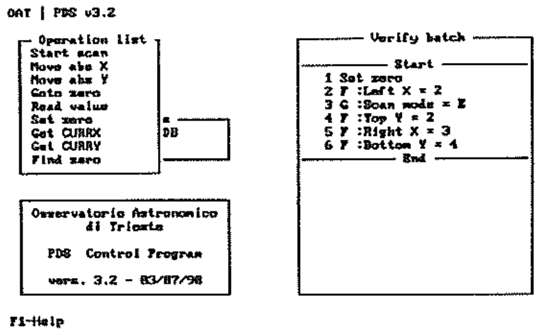

Fig. 27 : Operation execution list

DEL - Delete

Pressing the DEL key deletes the operation under the current position bar. Before the deletion is performed, a confirm dialog box is shown (see figure 28).

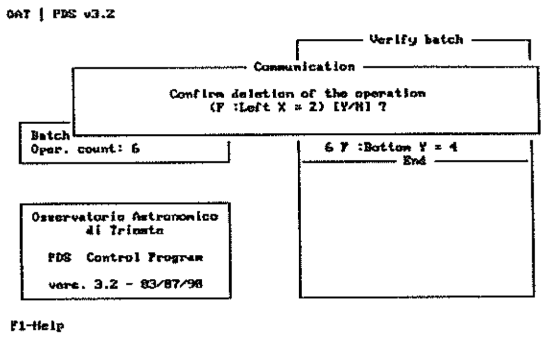

Fig. 28 : Confirm deletion ?

ENTER - Modify

Pressing the ENTER key lets the user to modify the operation under the current position bar. The only thing that can be modified is the value of a variable setting, nor the variable itself or an operation execution. If the user wants to modify them the only way to do it is to delete an reinsert the correct operation (see figure 29).

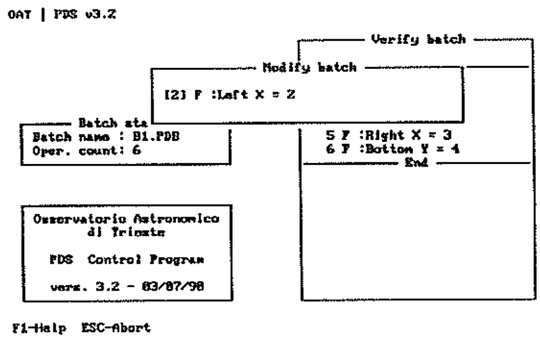

Fig. 29 : Modify batch file operation

F1 - Help

Pressing the F1 key (the standard HELP key) will cause an help screen to appear. This help screen is referred to the operation marked by the current position bar.

ESC - Exit

Pressing the ESC key will terminate the BATCH VERIFY session and returns the user to the main menu.

6.9. Utilities

The UTILITIES entry in the main menu presents to the user a submenu containing two items: FILE MANAGER or SET COLORS (see figure 30).

```
OAT | PDS v3.2

 ---- Main Menu ----
 Reset system
 Batch file manager
 Format options
 Scan operations
 Interactive
 Utilities          -- Utilities --
 Quit program        File manager
                     Set colors

 Osservatorio Astronomico
       di Trieste

   PDS  Control Program

   vers. 3.2 - 03/07/98

F1-Help  ↑↓-Move  ↵-Select
```

Fig. 30 : UTILITIES submenu

6.9.1 File management

This is a file management facility that lets the user to rename, erase or view images, batch files and log files. This module is build using a series of menus that asks the user the type of file to manipulate (figure 31), the operation to perform on that kind of file (figure 32) and, finally, the name of the file.

```
OAT | PDS v3.2

 - Files to manipulate -
 Images
 Batch files
 Log files

 Osservatorio Astronomico
       di Trieste

   PDS  Control Program

   vers. 3.2 - 03/07/98

F1-Help  ↑↓-Move  ↵-Select  ESC-Abort
```

Fig. 31 : Selecting the type of file

```
OAT | PDS v3.2

 ┌─ Operations ─┐
 │Rename        │
 │Erase         │
 │View          │
 └──────────────┘


 ┌──────────────────────────┐
 │ Osservatorio Astronomico │
 │        di Trieste        │
 │                          │
 │   PDS  Control Program   │
 │                          │
 │   vers. 3.2 - 03/07/98   │
 └──────────────────────────┘

F1-Help  ↑↓-Move  ↵-Select  ESC-Abort
```

Fig. 32 : Selecting the operation

The first two operations, rename and erase, perform the same task upon every kind of file selected, while the third operation, view, displays a 16 pseudo color representation of an image or a formatted listing of a batch or log file.

6.9.2 Color settings

This submenu entry lets the user to define a private set of colors that will be used in all subsequent work sessions. This color sets are private to the user so many users can have different color sets.

The creation of a color set is achieved first by selecting the visual object whose color has to be changed (see figure 33) and then the color that will be used from then on. After that, the user is prompted with a dialog box for the storage of the new color set on the mass memory. If the user enter the U value, this color set will be used during this work session but will not be saved and the next time the user will log in the default color set will be used.

```
OAT | PDS v3.2

┌──────────── Set colors ─────────────┐
│A. Positive                          │
│B. Negative                          │
│C. Highlighted                       │
│D. System errors                     │
│E. Menu selected shortcut            │
│F. Menu unselected shortcut          │
│G. Menu selected item                │
│H. Menu unselected item              │
│I. Menu inactive item                │
│J. Help border                       │
│K. Help text                         │
│L. Help active keys                  │
│M. Help items                        │
│N. Help selected items               │
│O. Error window                      │
│P. Error text                        │
│Quit                                 │
└─────────────────────────────────────┘

F1-Help  ↑↓-Move  ↵-Select
```

Fig. 33 : Selecting the visual object

6.9.3 Off-line utilities

Three off-line utilities are provided with the PDS control program:

- SHOWLOG - a batch or log file formatter
- SHOWIMG - an image viewer with limited image processing capabilities
- PDSUSER - a user management program

SHOWLOG

This program requests the complete name of the log or batch file to be inspected and then formats and displays the contents of the file on the screen or on the printer.

SHOWIMG

This program asks to the user the name of the image file and displays it on a VGA screen. Then the user can change the image color palette with the numeric keys (1-gray scale, 2-reverse gray scale, 3-rainbow, 4- reverse rainbow, 5-heat, 6-reverse heat, 7-jigsaw with period 2, 8- jigsaw with period 4, 9-jigsaw with period 8, 0-random). Another facility is the zoom/unzoom operation. First the user selects the area to be zoomed with the mouse and then, using the Z key or the right button, zooms the area. Another pressing of the Z key will furthermore zoom the area using a factor of two. The U key will unzoom the area.

PDSUSER

This program manages the user list. It has not to be available to the normal user and only the system manager can have access to it.

7. Data structure

The PDS control program is basically a sophisticated data collector. These data are stored in variuos places during the scan operation, first in the control computer internal memory than in the local mass storage devices and finally on the external mass storage media (see figure 34).

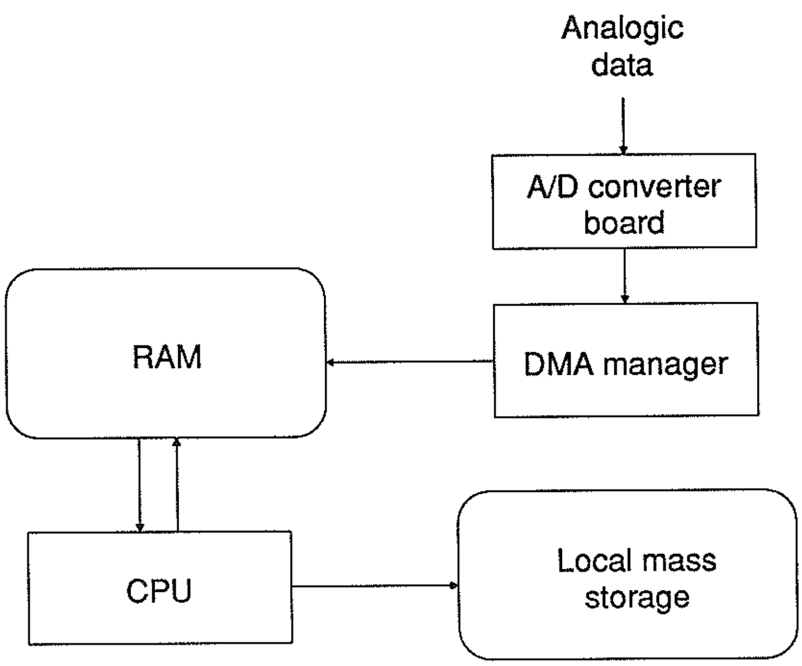

Fig. 34 - Data flow

7.1. DMA usage

The internal data storage device is the control computer RAM. A portion of it is assigned to the A/D converter board and to a DMA channel. This makes a tight connection between the incoming data and a well defined memory location. The use of a DMA-driven memory writing system has two major advantages:

- SPEED - the access to the internal memory is made in parallel with the CPU accesses in a way that the execution speed of the program is not limited by other memory accesses

- RELIABILITY - the access to the internal memory is completely managed by the DMA controller that works only when the system bus is free. Concurrent accesses to the memory are thus avoided.

7.2. Shared Data Base

A basic concept in the data structure of the PDS control program is the *Shared Data Base* (SDB). It is a memory resident data base that can be accessed either by the main control program or by the DMA controller.

Inside the Shared Data Base there are eight data objects:

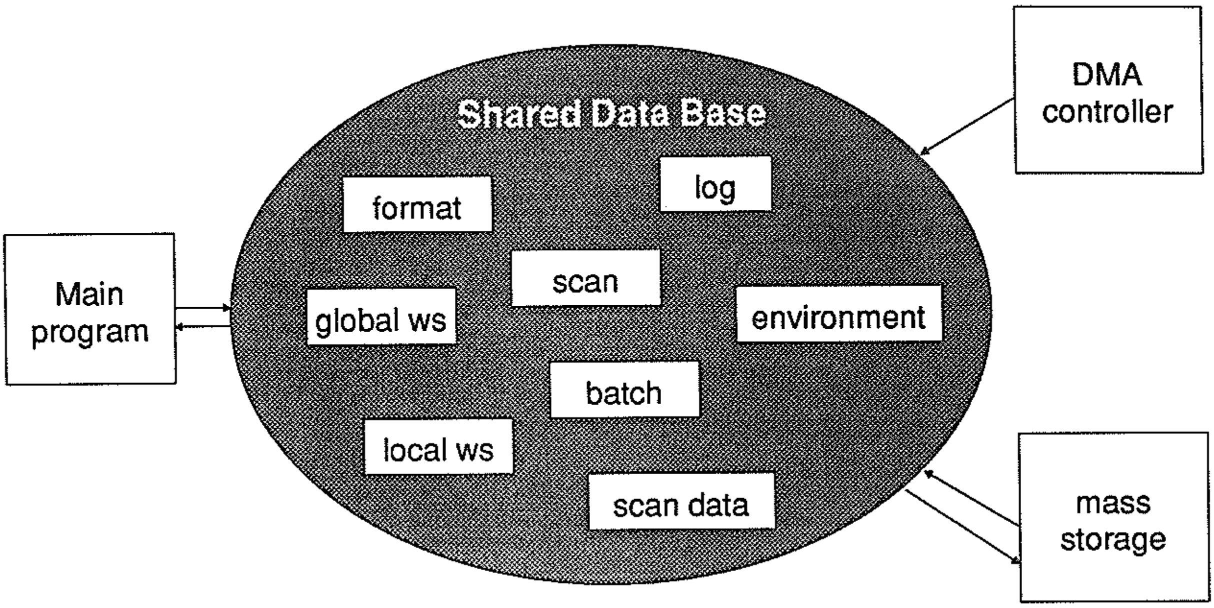

Fig. 35 - The Shared Data Base

These eight objects are those seen before in the discussion of the main menu entry. The SDB was built after some, for various aspects unsatisfactory, tests. The early ones used the local mass memory for a temporary storage, causing clear problems in speed, then large data blocks were transferred between the various modules of the main program, but the most simple and efficient solution was to put the shareable data in a common memory location, leaving only the CPU and the DMA controller to access the data concurrently.

7.3. Mass storage

After a scan line operation is performed , the problem to store a block of data of variable size arise. The most secure way to do this is to rely on a mass memory storage device like a fast hard disk. The internal scan data buffer is of limited size (about 60 kilobytes) and can store a single scan line; when a lot of lines are collected they can easily overflow the internal memory capability of the control computer (640 kilobytes).

After a successful scan line operation, the contents of the scan data buffer are immediately trasferred to the local mass storage. At the end of the complete scan operation, a header file is also written on the disk with all the informations related to the data collected.

7.3.1 Data file format

The data collected and stored on the local mass storage device have a common format. There are two files: the first, with the .IMG suffix, is the real data file; it is composed by a continuous stream of integers that represents the data collected (values can range from 0 to 4096 due to the internal 12 bit PDS A/D converter), the second, with the .HDR suffix, is an ASCII file with a set of keywords containing all the informations related to the .IMG file (format of data, time and date of scan, etc..).

7.3.2 Remote storage

After the completion of the scan process, the data file can be transferred to other systems for the data analysis. The systems that can use the PDS data are basically PCs, the VAX/VMS machines and Unix workstations. There are four methods to transfer the data files to this computers:

- PC LAN - a PC-dedicated network, based usually on the Ethernet network technology; this works only between linked PCs
- NFS - the Network File System, works between the control computer and VAX/VMS machines or between the control computer and Unix workstations
- DPCI - the Domain Personal Computer Interconnect, works between the control computer and Apollo/Domain workstations
- FTP - the File Transfer Protocol, connects all machines based on the TCP/IP protocol.

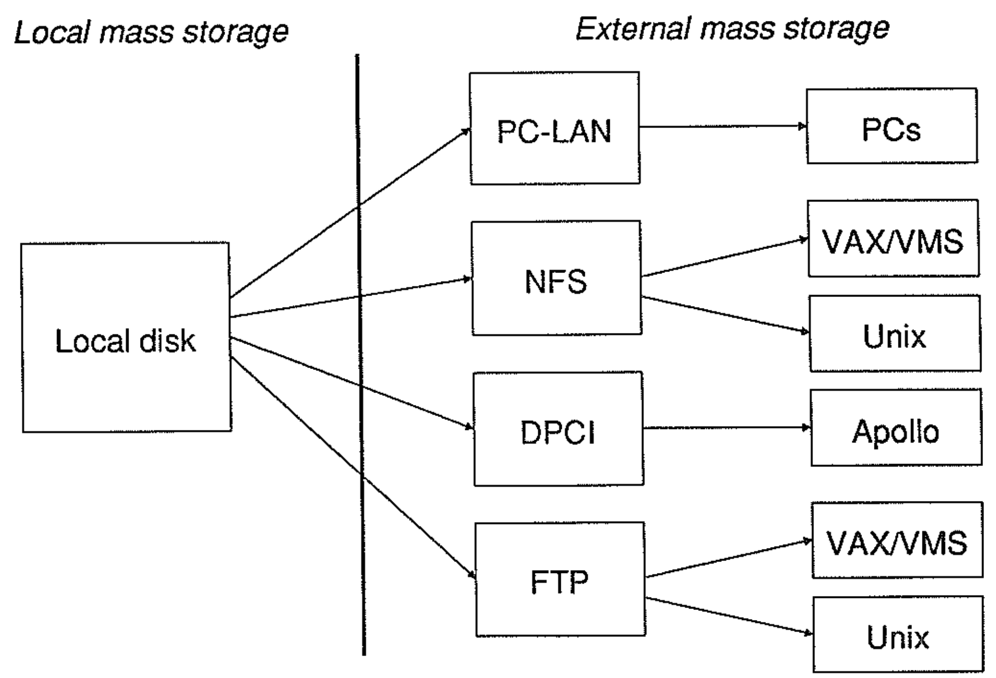

Fig. 36 - Methods of data transfer

8. Software interfaces

The PDS control program is written in Turbo Pascal 5.5, but for a set of functions which had to be coded using directly the assembly language for the sake of speed. In fact, the movement of the PDS stage can reach the speed of 1718 microns/sec that means a maximum sampling rate of 1718 Hz. The effective data sampling is done by the A/D converter board RTI-815 that stores the data via the DMA controller in the shared data base, but the control program has to check for the scan ending point and for the stability of the stage movement. This is done by two subsystems: Stage movement and Lock control.

8.1. Stage movement

The control of the stage movement is done by a subsystem that samples about eighteen time at second (the standard system clock) the counters that report the current stage position. These positions are converted in microns and then displayed on the top right of the screen every third of second. During the data acquisition process, the sampling of the counters is disconnected from the standard system clock and directly connected to the sampling trigger.

8.2. Lock control

During the scan line operation a very strict control is requested over fluctuations that the stage can make over the non scan axis. The lock control subsystem is activated during the data acquisition process and continuosly checks the deviations from the correct axis. If a deviation is detected, a correction routine takes the control of the stage and tries to restore the correct positions. From a series of test performed on the PDS, a very litte correction time was noted and the maximum deviation from the correct position, given a strong shock on the PDS machine, resulted in 4 microns, recovered in less than half a second.

Acknowledgments

The basis for this work were provided either by the system analysis on the old version of the PDS data acquisition system and by an extensive research on the current technological trends done by Dr. M. Pucillo. His are the ideas about the Shared Data Base and the scan line process. Useful discussions were made also with Dr P. Santin and Dr. G. Sedmak.

References

- 1010A Microdensitometer operation and maintenance manual with
  PDP-11 computer control.
  Perkin-Elmer Applied Optics Division, 1979