

St. Clement Okhridsky University of Sofia

SOFTWARE RESEARCH & DEVELOPMENT LABORATORY

PYLDIN 601A
PROGRAMMERS GUIDE

Sofia 1990

PYLDIN 601A
PROGRAMMERS GUIDE

UniDOS, UniBIOS, UniPASCAL and **UniBASIC** are trademarks of SOFTWARE RESEARCH & DEVELOPMENT LABORATORY.

PYLDIN 601A is a trademark of Abacus Co.

EduLAN is a trademark of ComseD.

Copyright ©1989-1990 SOFTWARE RESEARCH & DEVELOPMENT LABORATORY

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic mechanical, photocopying, recording or otherwise, without the prior written permission of SOFTWARE RESEARCH & DEVELOPMENT LABORATORY. Printed in Bulgaria.

SOFTWARE RESEARCH & DEVELOPMENT LABORATORY

5, A. Ivanov Blvd.

1126 Sofia, Bulgaria

phone: (359 2) 70 61 58, 62 77 54

fax: (359 2) 70 62 48

Contents

1	Getting acquainted	7
1.1	Specifications	7
2	UniDOS	9
2.1	UniDOS files	9
2.2	UniDOS commands	9
2.3	UniDOS error messages	16
3	The Text Editor – UniED	19
4	UniBIOS calls	21
5	Assembler Instructions	23

Preface

The realization of the system software for the Pyldin microcomputer is a result of the hard work and the experience of the authors and the whole group participating in the Pyldin project. The aim was a creation of a comfortable and functional software environment in a respect with the computers of this class.

The style of the system software was intentionally directed to what is known as PC-DOS (and run on IBM-PC machines and compatibles). The benefits of such approach is the considerable reduce of time necessary to start over (in case of such experience). This approach was observed also in many technical decisions. One of the great advantages is the physical and logical 100 % compatibility of floppy disks of Pyldin and IBM-PC compatible machines. Thus you can read and write diskettes with Pyldin and/or with other machine. So you can say that Pyldin is *file-compatible* with an IBM-PC.

This book don't teaches you programming, it is a reference manual. A prior experience with computers is supposed. The book itself consists of several chapters - acquaintance with the computer, description of UniDOS commands, the UniBIOS calls, the Text Editor, the microprocessor instructions and the assembler directives.

The UniDOS commands are very similar to PC-DOS ones, with very exceptions they may be considered a subset of PC-DOS. The Text editor is similar in concept to editors like Word and Word-Perfect or the PE2. Two other products – UniPASCAL and UniBASIC are also similar to the coresponding products found for IBM-PCs.

We hope, that you will enjoy working with our software.

Authors

Chapter 1

Getting acquainted

1.1 Specifications

Technical specifications

- Microprocessor: CM601 (Motorola MC6800 equivalent) 1 MHz clock frequency
- Dynamic memory (DRAM): 64 Kbytes
- Read-Only Memory (ROM): 4 Kbytes expandable to 68 Kbytes

Chapter 2

UniDOS

2.1 UniDOS files

2.2 UniDOS commands

The following convention is used in describing UniDOS commands:

- The TYPEWRITER face is used for items entered via keyboard or displayed.
- Optional (not required) arguments are placed in square brackets [like this].
- Concrete file name or device name which must be supplied by the user are placed in angle brackets < like this>.
- Space characters (blanks) which are part of the command are shown with an underscore character like `like_this`.
- All other items must be entered exactly as shown, lowercase and uppercase are not distinguished.
- Wildcard characters * replacing any amount of symbols till the end of the field and ? replacing one character may be used in filename specifications in most of the commands.
- Typical examples of commands are provided and the type of the command (internal or external) is given.

Redirection of input and output

Input and output of commands and most of the programs may be redirected from/to file or character device. The < symbol is used for redirecting the input of a command (instead of the keyboard) and > or >> symbols are used for redirecting of the output. The >> is used for explicit appending to file (the file may be created if does not exist), otherwise the file is always created (old contents are lost).

Syntax: <command> <redirection symbol> <filename or device>

Example: `_DIR>_PRN:`

CD – Change Directory

This command is used for changing current directory. If no parameters are specified, the current directory is displayed. The symbol \ is used for the root directory and for separator between subdirectories. The symbol .. is used for the parent directory.

Syntax: CD_ [drive:] [pathname]

Example: CD_ \MYFILES

Type: internal

CLS – CLear Screen

This command is used for clearing the screen. The videomode is not changed.

Syntax: CLS

Example: CLS

Type: internal

COPY – copy file(s)

This command is used for copying file(s). Wildcards * and ? are accepted. Omitted parameters are substituted with default drive and directory and the filename(s) are the same. Existing files are overwritten if the target drive/directory differs from the source. The character devices CON:, PRN:, AUX: and NUL: can be used as source (if possible) or target.

Syntax: COPY_ <source specification> [<destination specification>]

Example: COPY_ MYFILE.TXT_ B:

Type: internal

Error messages:

FILE CANNOT BE COPIED ONTO ITSELF – source and target specifications are identical.

PATH NOT FOUND – source specification is not valid.

DATE – set current date

This command is used for setting the current date. If no parameters are given the old date is displayed and the user is asked for the new date. If no hardware clock is present the date is set to January 1-st, 1989. The format of the date specification is dd-*mmm*-[yy]yy, so dates after 01-JAN-[19]80 and before 31-DEC-2099 may be entered.

Syntax: DATE_ [<date specification>]

Example: _DATE_ 03-JUN-90

Type: internal

DIR – DIRectory

This command is used for displaying the contents of the specified directory. If no directory is specified contents of the current directory are displayed. Wildcards * and ? are accepted. The output may be redirected to file or printer.

Syntax: DIR_□[<drive:>][<path>]

Example: DIR_□A:*.TXT

Type: internal

DISKCOPY – DISKette COPYing

This command is used for copying of diskettes. Only standard 180, 360 or 720 KB diskettes may be copied. Target diskette may be formatted if necessary. The process of copying may be quite tedious on one-floppy system.

Syntax: DISKCOPY_□<source_drive>_□<destination_drive>

Example: DISKCOPY_□A:_□B:

Type: external

DEL – DELeTe files

This command is used for erasing disk files. Hidden, system or read-only files are not erased. The number of erased files is displayed.

Syntax: DEL_□[<drive:>][<path>]<filename>

Example: DEL_□*.BAK

Type: internal

ECHO

This command is intended for use in JOB-files. Its purpose is to echo a message on screen or to suppress output during processing of a JOB file.

Syntax: ECHO_□[ON or OFF] or [<any combination of printable characters>]

Example: ECHO_□OFF

Type: internal

FORMAT – FORMAT diskette

This command is used for initializing diskettes. Physical format is performed and new FAT (File Allocation Table) and directory are created. The following formats are supported:

- 360 KB DS/DD (standard)
- 180 KB SS/DD
- 720 KB DS/QD – this format is possible if the floppy disk drive is 80-track model

If the number of tracks is not specified, the default is 40 tracks, except the case of 80-track FDD, which has to be declared with the corresponding MODE command. If the number of heads is not specified it is assumed that the FDD is double-sided.

Syntax: FORMAT_<drive:> [/<number_of_tracks>] [/<number_of_heads>]

Example: FORMAT_A:

Type: external

HANDLES

This command is used for specifying the maximum possible amount of open files in one and the same moment. The default is 15 (after bootstrap) and may be increased up to 32. Each new handle uses about 530 bytes of memory, thus reducing the amount of memory available for programs. The number of handles can be increased only, decreasing is ignored. It is recommended to include this command in the AUTOEXEC.JOB file.

Syntax: HANDLES_<number_of_handles>

Example: HANDLES_24

Type: internal

MD – Make Directory

This command is used for creating new subdirectories. If the full path is not provided the directory is created as a subdirectory of the current one.

Syntax: MD_ [<drive:>] [<path>] <directory>

Example: MD_TMP

Type: internal

MODE – select MODE of operation

This command is used to select mode of operation of specific devices supported by UniDOS. The current release supports display, auxiliary channel (RS-232) and floppy disk drives.

Display mode

User can select between 80-column and 40-column text modes or to return to text mode from graphics. In case of 40-column text mode default attribute can be specified. The attribute is a positive number between 0 and 255 or \$FF in hexadecimal notation which is formed by the background color (0..15) added with foreground color (0..7) multiplied by 16 and finally added with 128 for blinking if desired. If no attribute is given black and white mode is selected.

Syntax: MODE_<number_of_columns>_ [<attribute>]

Example: MODE_40

Type: internal

Auxiliary channel mode

User can select baud rate and format of the serial interface (RS-232-C). The following baud rates are supported: 150, 300, 600, 1200, 2400, 4800, 9600, 19200. The eight possible formats (0 to 7) specify the number of data bits (7 or 8), parity control (none, odd or even) and the number of stop bits (1 or 2).

No	data bits	parity control	stop bits
0	7	even	2
1	7	odd	2
2	7	even	1
3	7	odd	1
4	8	none	2
5	8	none	1
6	8	even	1
7	8	odd	1

Syntax: MODE_<baudrate>_<format>

Example: MODE_1200_4

Type: internal

Floppy Disk Drive mode

In case of using of quadro density floppy disk drives (80-track) the user must specify explicitly the type of FDD. This is necessary for proper formatting of diskettes in such drives and reading/writing of 360 KB (double density) diskettes (40-track) in such drives. In this case the FDD is automatically driven in a double step mode. The corresponding command is recommended to be included in the AUTOEXEC.JOB file.

Syntax: MODE_<drive:>_<number_of_tracks>

Example: MODE_A:_80

Type: internal

MON – MONitor

This command is used for invoking system monitor program. The commands of the monitor program are described in the appendix. Returning to UniDOS is done by monitor (quit) command.

Syntax: MON

Type: internal

PAUSE

This command is intended for use in JOB-files. Its purpose is to provide pause, which is terminated by pressing a key on the keyboard. When executed the following message is displayed:

Press any key to continue...

User may be asked before this to change diskettes, etc.

Syntax: PAUSE

Type: internal

REM – REMark

This command is intended for improving documentation of JOB-files. Any characters on the line following REM are ignored during processing of the JOB-file.

Syntax: REM_ [<any text>]

Example: REM_This is my comment.

Type: internal

REN – REName files

This command is used for renaming files. Wildcards are accepted, but should be used carefully. The new name must be unique.

Syntax: REN_ [<drive:>] [<path>] <oldname>_ <newname>

Example: REN_ WORK.TXT_ LETTER.TXT

Type: internal

RD – Remove Directory

This command is used for removing subdirectories. The directory must be empty (all files, including system, hidden and subdirectories must be removed before that). The main directory can not be deleted.

Syntax: RD_ [<drive:>] [<path>] <directory_name>

Example: RD_ TEMP

Type: internal

TIME – set current time

This command is used for setting current time. If no clock card is present the time is set to 0:00:00 after power on. Time must be specified in 24 hour format i.e. 0:00:00 to 23:59:59 are valid entries.

Syntax: TIME_ [<time specification>]

Example: TIME

Type: internal

TYPE – display contents of a text file

This command is used for displaying contents of text files. Wildcards are NOT accepted. If the length of text lines exceeds the screen width the line is splitted at the corresponding position i.e. at the 40-th column. The process is terminated upon physical end of the file or coming upon EOF character – CTRL-Z. The rest of the file (till the physical end) is skipped.

Syntax: TYPE_ [<drive:>] [<path>] <filename>

Example: TYPE_ MYFILE.TXT

Type: internal

VER – VERsion of UniDOS

This command is used for displaying the version of UniDOS loaded. The logo of UniDOS is displayed.

Syntax: VER

Type: internal

2.3 UniDOS error messages

Access denied () .
 Address failure () .
 Attempt to remove the current directory ()
 AUX: framing ()
 AUX: overrun (,)
 AUX: parity (-)
 AUX: timeout ()
 Bad command or file name ()
 Cannot execute .CMD file (.CMD)
 - .CMD .
 Data failure () .
 Directory exists ()
 Directory not empty ()
 Disk full ()
 Disk write protected ()
 Failure in .PGM file (.PGM) .PGM
 .
 FAT failure (FAT- -)
 File already open ()
 File cannot be copied onto itself ()
 File creation failure ()
 File exists ()
 File opening failure ()
 File not found ()
 General failure ()
 .

Insufficient memory ()
Invalid date () .
Invalid drive ()
Invalid media type ()
Invalid name ()
Invalid number of parameters ()
Invalid parameter ()
Invalid sector ()
Invalid time ()
Not disk file (-) .
Not same device () .
Path not found () ; .
Printer general failure () .
Printer out of paper ()
Printer timeout ()
Root directory full ()
Too many open files ()
 , HANDLES.

Chapter 3

The Text Editor – UniED

Chapter 4

UniBIOS calls

Chapter 5

Assembler Instructions