The Professional Adventure Writing System

Differences between the Spectrum 48K / 128K versions
& the Amstrad CP/M version

(Compiled by Gareth Pitchford of 8bitAG.com / Version 1.3)

Background

The Professional Adventure Writing System (often referred to as the PAW or PAWs) is an 8-bit text adventure creation toolkit developed by Gilsoft and published in the United Kingdom in 1986.

The PAWs was the successor to Gilsoft's pioneering multi-platform creation tool The Quill. The PAWs itself was succeeded by the development systems SWAN and DAAD, both of which were produced for commercial clients.

This document outlines some of the key differences between the version of the PAWs created for the ZX Spectrum and the one which was produced for the Amstrad CP/M (and later for PC DOS). It should be useful for authors creating text adventures targeting both platforms, or those porting existing games between platforms.

Note: the CP/M version's manual references the fact that it would also run on the BBC Micro with a Z80 co-processor as well as the Amstrad PCW 8512. In this document we'll be comparing the Spectrum version to a disk-based Amstrad CPC464 or CPC6128 running CP/M PAWs.

The process of adventure writing

The ZX Spectrum version is a complete self-contained authoring environment. A menu-driven system allows you to write and edit the PAWs database directly and almost instantly test your code by launching a playable version of your game. If you have a 128K machine or disk-based system you can take advantage of the extra memory available, both to create larger adventures, and also to reduce the amount of loading required when swapping in and out sections of the editor program.

The ZX Spectrum PAWs program allows you to save a copy of your database at any point or save a full, standalone version of your adventure. Spectrum PAWs, and any standalone adventures, can run from tape or from disk.

The Amstrad version of the PAWs is disk-only. It runs under CP/M which can be loaded from the PAWs disk using the command |CPM on an Amstrad CPC machine. The PAWs, and adventures produced with the CP/M PAWs, also run on Amstrad PCW machines as well as other CP/M compatible machines, such as a BBC Micro with a Z80 co-processor.

A .SCE "source" file, consisting of a collection of tables of data in ASCII format, must be prepared using a suitable text editor. This file is then converted into a PAWs database (.PDB file) using the CP/M program PAWCOMP. Compiling takes several minutes to complete. A separate CP/M program, PAWINT allows the compiled database to be played, and also provides the user with an option to create a standalone version of their adventure.

Although you can create your .SCE file on an Amstrad (or emulated Amstrad) using a text editor, as in the 1980s, these days authors will probably wish to use a PC-based program or a third-party authoring aid such as inPAWS.

The Screen & Colour

The default ZX Spectrum screen layout is 32-columns wide and users can create text in all Spectrum colours as well as use attributes such as brightness and flash. Colours can be changed using CondActs or through control codes embedded in the text.

The standard screen on the Amstrad (CPC) CP/M version of the PAWs is 80 columns wide. By default, the text & background colour are tied to the system disk from which CP/M is launched. (But, see: EXTERNS)

The Spectrum version of the PAWs gives the user the option to move the input line to the lower part of the screen. The Amstrad CP/M version does not.

Fonts & UDGS

The Spectrum PAWs allows the use of custom fonts. Multiple fonts can be incorporated into a game and switched between using the CondAct CHARSET or embedded control codes. The Spectrum PAWs comes with a selection of pre-made fonts and users can create or tweak their own fonts using the built-in character editor.

Spectrum users can include up to nineteen 8x8 pixel user defined graphics (UDGs). Spectrum users can also alter both these and the 16 predefined shades using the built-in character editor.

The font in the Amstrad version of the PAWs is the standard CP/M one. (But, see: EXTERNS) The Amstrad version does not support User Defined Graphics (UDGs).

Graphics

The Spectrum version of the PAWs includes a full graphics editor that allows images to be drawn from a set of instructions based around movement, lines and shading.

The original Amstrad CP/M version did not include a graphics editor or native support for graphics, but (as in the Spectrum version) externally sourced graphics could be integrated into adventures by using a user's own machine code routines. (See: EXTERNS)

Gilsoft later (in 1990) made an update available for the CP/M version of PAWs that added support for disc-based graphics in CPC/PCW games, without the user needing to provide their own machine code routines.

To utilise this add-on, games needed to have 2.5K of TPA space remaining. Images could take up 2 to 20 lines of screen space and were mode 1 (40 columns, 4 colours). A separate file ADDARTx.COM was used to add the graphics patch to a completed game together with a .MAP file, which included the data as to which image was associated with which location. The patch also added the ability to change the text colour for each location, on CPC, if desired. As part of the process of adding graphics to the game, users would need to change the number of columns used to 40, and also adjust the number of rows, taking into account the space required for the graphics, so the 'wordwrap' and 'more' features worked correctly.

Screen Modes & Advanced Screen Control

Spectrum authors have much greater control of how text appears on screen.

The Spectrum version of PAWs has five pre-set screen modes, with supplementary options. These modes govern screen scrolling, how pictures appear, whether the screen is cleared before a location is described and whether text or graphics are locked in place (with text scrolling underneath). Further details can be found on pages 22 and 23 of the Spectrum version of the Technical Manual.

The Spectrum PAWs has additional CondActs (LINE, GRAPHIC, PROTECT) to control aspects of the screen modes, such as the split-screen mode. There is also the command INPUT which allows the input prompt to be moved to the lower part of the screen.

Further Spectrum-only CondActs (PRINTAT, BACKAT and SAVEAT) allow authors to print text at specific positions on the screen and then return the active text area to a previous saved position.

The standard CP/M version technically has four screen modes but in reality there are only two options; if the screen is cleared before a location is described and if the prompt "More…" appears when the screen area fills.

Sound

The Spectrum version of the PAWs has a rather limited built in sound command in the form of the CondAct, BEEP. It's useful for sound effects but any music is best produced using external routines.

The Amstrad CP/M has a very basic sound CondAct, BELL, which simply triggers the keyboard bell.

Note producing sound in Spectrum adventures using BEEP also has the effect of creating a pause between CondActs. You may need to add extra pauses to compensate when transferring a game from the Spectrum to the Amstrad.

System Messages

The Amstrad reserves the use of the system messages 54 – 60 for disk-related messages. ZX Spectrum authors use system messages 54 – 60 for their own messages, although in practise sysmess 54 should be used for the filename prompt and 55-60 should be left blank to ensure compatibility with the CP/M version.

Response Table / Process 0

The Spectrum version uses the terminology "Response Table" while the Amstrad version refers to this table as Process 0.

Process 1 and 2

In the Spectrum version of the PAWs, words (VERBS and NOUNS) attached to the beginning of entries in Process 1 and Process 2 have no meaning. They can be used to comment on the function of each entry and will be completely ignored by the interpreter. There will be no attempt to match these words against the player's input.

In the Amstrad CP/M version of the PAWs, such words are not ignored so you should not use them as aide-memoires. The CP/M version of the PAWs will attempt to match any VERBs and NOUNs in every process table, not just the ones called from the response table (process 0). (There is an additional document on 8bitag.com/info which demonstrates & discusses this).

Entries in the response table (process 0) or processes called from the response table are always matched against the verb and noun in both versions of the PAWs.

(Note: Spectrum A16+ versions of the PAWs can be temporarily forced to behave like the CP/M, matching entries in process tables 1, 2 and their dependents, by using the entry LET 58 128 at the start of each table. See the section on version A16+ changes below for more details)

EXTERNs

In the Spectrum version, users can use the CondAct EXTERN to execute either their own machine code routines or small BASIC programs. Spectrum adventure authors often used this feature to activate graphic effects, play tunes, load images from disk or even, in some cases, play small arcade games.

In the Amstrad CP/M version, users are restricted to using machine code routines with EXTERN. Once again suitable routines can be utilised to load graphics or change the font and screen colours, allowing authors to get around some of the limitations of the CP/M version. Check out Doomsday Lost Echoes https://doomsdayproductions.org/ for an idea of just what is possible.

Text Adventure Size

The ZX Spectrum version of the PAWs offered support for 128K machines, allowing users to utilise additional pages of memory to store location and message text.

The CP/M version did not include such support.

Overlays

As well as using overlays to swap in and out sections of the PAWs development system as required, the Spectrum version (from version A14 onwards) had support for commercial and user overlays. This allowed companies such as Kelsoft (through their Phosis Tel Mega overlays) to add extra features & development tools to the main PAWs system. End users could also create their own machine code additions as long as they conformed to the standards in the Gilsoft overlays guide.

Spectrum Version A16+ changes

Spectrum versions of the PAWs, version A16 onwards, introduced various changes designed to support giving PSIs multiple commands.

The PARSE condact now remembers its position in the current logical sentence (the player's input) and no longer affects the command line empty/valid flag. The current verb and adverb are also no longer cleared when PARSE is used.

This may affect older Spectrum adventures that use the PARSE command when loaded into Version A16+. These will usually require the addition of the NEWTEXT condact after each occurrence of PARSE in order to behave as the author originally intended.

The CP/M version of the PAWs does not support multi-parse.

A related change, made to support multi-parse, is the use of the previously reserved **system flag 58**. This flag can be used to temporarily alter the behaviour of Spectrum PAWs when working through process tables. If flag 58 is set to 128 then Spectrum PAWs will strictly match any verb noun entries in process table 1, 2 and their dependents. It should be noted that PAWs will automatically subtract 128, setting the flag back to zero, whenever process 1 or 2 is next carried out. If you want a permanent effect, then you would need to use a LET 58 128 entry at the start of each of these tables.

I would strongly advise reading through Gilsoft's 'A Supplement for the Spectrum – Version A 16' if you wish to understand the changes made in versions A16 onwards fully. It includes a worked example of multi-parse given in that documentation.

Other Undocumented Differences

(This section will be added to if any additional differences of note come to light in the future)

LET 38 issues

For some reason the Spectrum version allows the user to directly manipulate the player location flag with LET, e.g. **LET 38 5,** but the Amstrad CP/M version does not. **GOTO 5** would have to be substituted for that previous example.

Objects starting as WORN

The CP/M version has stricter rules, than the Spectrum version, regarding how objects are defined in the adventure database.

In the Spectrum version you can define an object as being “initially at” WORN without that object being defined as wearable/removable in the item weight table.

In the CP/M version any object that starts as worn must also be given the wearable/removable (clothing) property.

Differences between the ZX Spectrum & Amstrad CP/M Professional Adventure Writing System

Easy Reference Tables

System Messages

	ZX Spectrum	*Amstrad CP/M*
System Message 34	Used for cursor	(spare)
System Message 54	Prompt for Filename (disk)	Prompt for Filename
System Message 55 - 60	(spare)	SAVE and LOAD messages

** by default, system messages 33 & 34 use UDGs in the Spectrum version of the PAWs*

Flags

	ZX Spectrum	*Amstrad CP/M*
Flag 29	Picture control flag	(spare)
Flag 39	Current top LINE of screen	(spare)
Flag 40	Screen Mode (range 0 to 4)	Screen Mode (range 0 to 3)
Flag 41	Line Split (set by PROTECT)	(spare)
Flag 49	Timeout Control (note differences[A])	Timeout Control (note differences[A])
Flag 58	*used in version A16+ (multi-parse)	(reserved for future expansion)
Flag 59	(reserved for future expansion)	(reserved for future expansion)

[A] in the Spectrum version, bit 3 and 4 of Flag 49 are used for the lower screen input option

CondActs Differences

	ZX Spectrum	*Amstrad CP/M*
MODE	**MODE** *0 - 4 0 - 3*	**MODE** *0 – 3*[B]
TIME	**TIME** *0 – 255*[C] *0 – 7*	**TIME** *0 – 255*[C] *0 – 7**

[B] The manual lists 0 – 255, but in practise there are only four screen options

[C] The Spectrum manual says 1.28 second intervals, the CP/M manual says 1 second intervals

** the CP/M technical manual says 255 but in reality the maximum value is 7*

ZX Spectrum-only CondActs

(mainly advanced screen layout & formatting commands)

LINE *0 – 20*	**GRAPHIC** *0 – 3*	**INPUT** *0 – 7*
PROTECT	**PICTURE**	**PAPER** *0 – 9*
INK *0 – 9*	**BORDER** *0 – 7*	**CHARSET** *0 – 255*
SAVEAT	**BACKAT**	**PRINTAT** *0 – 20 0 – 31*
BEEP *0 – 255 0 – 255*		

(plus UDGs, colour codes & character codes allowed in text)

Amstrad CP/M-only CondActs

BELL

*(from **http://8bitAG.com/info**)*