ZX Spectrum Vega Technical Specifications

ZX key map file (ZXK)

A ZX key map file contains configuration information which maps the Vega control buttons to the ZX Spectrum keys used by a game, along with other game information. There is one ZX key map file per game, each with a .zxk file extension.

On loading a game from SD card, the Vega looks for a file with exactly the same file name as the game file, but with a .zxk extension. If it finds one, it configures itself from the file contents, otherwise the default basic configuration is applied.

The ZX key map file is a simple text file that can easily be created with a text editor, and may contain several different types of configuration. Each configuration type occupies a single line in the file, and all follow the same structure (note that the square brackets below merely indicate that the enclosed value may be repeated, as necessary. They are not part of the key map file format.):

```
type: item [; item]
```

`type` is a single character identifier, and is always upper case.
`item` contains the configuration information for `type`, and is type specific. In some types contain a list of items, which are comma separated as follows: `item = entity [, entity]`

Comments may be include in the key map, and begin with the # character.

Configuration entity types

Entity description	Identifier
Game filename	F
Game title	T
ZX Spectrum model	M
Game pokes	P
Key to button map	K
Button action description	D
Episode	E

Configuration entity details

Entity name: Game file filename
Type identifier: F
Item content: The full filename of the game file, including file extension (`tap`, `z80`, `szx`). Paths are not allowed.
Restrictions: Maximum of 64 characters.

Entity name: Game title
Type identifier: T
Item content: The title of the game to be shown in the game menu.
Restrictions: Maximum length 32 characters. Only ASCII characters 32-127 are supported.

Entity name: Machine model
Type identifier: M
Item content: Which machine model the game should be loaded into.
Permitted values: `48` or `128`
Entity name: Pokes

Type identifier: P
Item content: A comma separated pair of numbers. The first is the memory address to be poked, with a value between 0 and 65535, the second the value to be poked between 0 and 255. multiple address, value pairs can be supplied by separating them with a semicolon: `16384,5 ; 49152,201`
Restrictions: Up to 12 pokes may be configured.

Entity name: Key mapping
Type identifier: K
Item content: Defines the key mapping for each Vega button. Multiple keys are expected to be configured, each separated with a semicolon. Keys must be configured in the correct order, as explained below.
Restrictions: A maximum of 22 keys may be configured, plus one start sequence.

Principles of key mapping

The key map defines a sequence of characters, one for each Vega button. Up to 11 buttons can be configured for the D-Pad and buttons Fire, Select, 1, 2, A, B and C. If more than 11 buttons are configured, then buttons 12 through to 22 are assigned to the secondary control set. See the Vega user manual for further details on primary and secondary control sets.

In addition to button-to-key mapping, the key map entity allows the configuration of a "game start key sequence". Consider the scenario where a game requires a sequence of keys to be pressed to start a game, for example key 3 to select keyboard controls, 1 to enable single player and key 0 to start the game. In the key map, such a start sequence would be configured as "`3 1 0`".

Key map content

Buttons are mapped to keys in a fixed order, each separated by a semicolon. Omitting a character, or inserting a blank, leaves that button unmapped.

First in the list is the start sequence. This is followed by up to 11 keys for the primary control set, and optionally, up to 11 keys of the secondary control set (here shown in italics), and always in the following order:

start sequence; up; down; left; right; fire; select; 1; 2; A; B; C*; up; down; left; right; fire; select; 1; 2; A; B; C*

On loading a game, if defined, the start sequence is immediately assigned to the select button [**S**]. Pressing select activates this sequence to start the game, and once complete, the sequence is disabled and the select button becomes assigned to it's configured key. The start sequence does not get re-assigned to the select button during the game session, as the Vega has no way of knowing that the player has returned to the game options menu. However, there is usually only a single key required to start a game which can be assigned to the select button, so in practice this scheme works well.

Abbreviations

Some ZX Spectrum keys are identified using abbreviations:

Key	Abbreviation
Enter	`EN`
Symbol Shift	`SS`
Caps Shift	`CS`
Space	`SP`

Example

The example below shows which Vega buttons we wish to map to which ZX Spectrum key:

Vega button	ZX Spectrum key
Up	Q
Down	A
Left	O
Right	P
Fire	Space
Select	Z
2	X
Start Sequence	3 1 0

As a key mapping configuration, this becomes:

```
K:3 1 0;Q;A;O;P;SP;Z;;X
```

Note that the mapping for the button [1] is blank, and that the list ends at button [2]. This leaves buttons [1], [A], [B] and [C] unconfigured, and so they will have no operation within the game. As there are less than 11 buttons configured, the game will have no secondary control set.

Entity name: Button action description
Type identifier: D
Item content: Gives the description of the action mapped to a Vega button, to be displayed when the game controls help is viewed by the game player. Multiple descriptions are expected to be configured, one for each mapped key, and each separated with a semicolon. Descriptions should match the keys mapped with the **K** (key mapping) configuration type.
Restrictions: A description may be no more that 22 characters in length. Note that because the display font of the Vega is proportional, descriptions containing many wide characters, such as the letter `M`, may become clipped when displayed. Descriptions should therefore be tuned for a good display fit.

The descriptions follow the same sequence as the key map entity **K**, beginning with the start sequence. As the start sequence is never displayed, it's description can be left blank, so it is typical to see the description list beginning with a semi-colon.

The button action descriptions for the example key map above, would be:

```
D:;Up;Down;Left;Right;Fire;Pickup;;Drop
```

Entity name: Episode
Type identifier: E
Item content: Defines that the game being configured is one in a series of episodes, or parts. The format is always a pair of numbers separated by a comma. The first number is the episode in the series, the second number is the total number of episodes in the series. Thus, a configuration of `E:2,3` says that the game is episode 2 of 3.
Restrictions: May only contain numbers.

Example key map file

The following example key map file is for Horace Goes Skiing, as included in the Vega games library but with the addition of an infinite cash cheat poke. The key map has the filename `Horace Goes Skiing.zxk`, matching that of the game filename, but with a `zxk` file extension. It configures eight action keys, repeating ENTER for Vega buttons [F], [1] and [2]. It does not have a specific mapping for a game start sequence:

```
T:Horace Goes Skiing
F:Horace Goes Skiing.tap
M:48
# Cheat poke for infinite cash
P:30027,0;30644,0;30762,0
K:;Q;Z;I;P;EN;S;EN;EN
D:;Up;Down;Left;Right;Enter;Start Game;Enter;Enter
```

The following example key map is for Skool Daze, as included in the Vega games library. This key map has the filename `Skool Daze.zxk`. This example is more complicated than that for Horace Goes Skiing, as it configures 19 actions, across the primary and secondary control sets, and provides a start sequence of Enter followed by N:

```
T:Skool Daze
F:Skool Daze.tap
M:48
K:EN N;Q;A;O;P;F;J;H;CS;S;W;EN;Q;A;O;P;F;;Y;N
D:;Up;Down;Left;Right;Fire;Jump;Hit;Hold to Run;Sit/Stand;Write;Enter;Up;Dow
n;Left;Right;Fire;;Yes;No
```