X1S.ORG

User's Guide

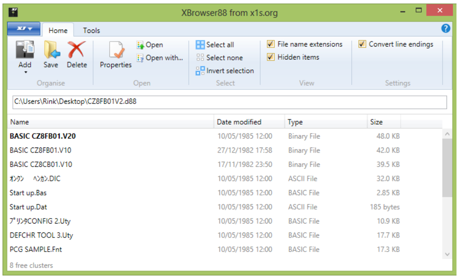
Contents Page

- Introducing XBrowser88 1
- Downloading and installing XBrowser88 2
- Working with D88 disk images 3
- Converting a disk image 11
- Appendices 13

Disk Utility Software

XBrowser88

上手に使って上手に節電



☑ File name extensions
☑ Hidden items
☑ Convert line endings

C:\Users\Rink\Desktop\CZ8FB01V2.d88

Name	Date modified	Type	Size
BASIC CZ8FB01.V20	10/05/1985 12:00	Binary File	48.0 KB
BASIC CZ8FB01.V10	27/12/1982 17:58	Binary File	42.0 KB
BASIC CZ8CB01.V10	17/11/1982 23:50	Binary File	39.5 KB
ｵﾝｸﾝ ﾍﾝｶﾝ.DIC	10/05/1985 12:00	ASCII File	32.0 KB
Start up.Bas	10/05/1985 12:00	BASIC File	2.85 KB
Start up.Dat	10/05/1985 12:00	ASCII File	185 bytes
ﾌﾟﾘﾝﾀCONFIG 2.Uty	10/05/1985 12:00	BASIC File	10.9 KB
DEFCHR TOOL 3.Uty	10/05/1985 12:00	BASIC File	17.7 KB
PCG SAMPLE.Fnt	10/05/1985 12:00	BASIC File	17.3 KB

8 free clusters

1 Introducing XBrowser88

XBrowser88 is an application for working with Sharp X1 floppy disk images on PCs. It is developed by X1s.org to help programmers, hardware hackers, and general users, get more from their X1 series machines.

Floppy disk images, such as the Pasopia D88 format used here, are a popular method of distributing software for old computers — they include all of the information from the original disk, along with information about how the contents were arranged. This allows emulators on modern computers to recreate the floppy disk perfectly.

XBrowser88 is intended for use with disk images of 5.25-inch 320KB disks that have been formatted by Sharp-HuBASIC CZ-8FB01. With it, you can open these disk images and examine the contents. You can save files from the disk to your PC, edit them, and then transfer them back into the disk image.

XBrowser88 also has features to manipulate files in ways that are difficult or inconvenient on the Sharp X1 such as choosing the file that is loaded by the initial program loader (IPL) when the system is started, or changing the load and execute addresses of programs.

For programmers, XBrowser88 is an essential part of your workflow. You can develop great Sharp X1 programs with modern text editors, assemblers, and graphics software, and then easily test these programs in one of the popular Sharp X1 emulators. In many cases, you can even use XBrowser88 while your emulator has the disk image mounted in one of its floppy drives. When your project is finished, the disk image can be transferred to a real floppy disk using an HxC Floppy Emulator, for testing or distribution on the actual hardware it was designed for.

For gamers and general users, XBrowser88 is a convenient and friendly environment for managing and backing up your files, and working with floppy disk images downloaded from the Internet.

The XBrowser88 User's Guide has been written to help you learn and use XBrowser88. In addition to this guide, you can find more help, or ask questions, at the X1s.org website at **www.x1s.org**.

2 Downloading and installing XBrowser88

XBrowser88 is easy to download and install. All you need to do is download the application file from the X1s.org website and save it somewhere on your system. Before you start, check that your PC meets the minimum system requirements.

System requirements

If you want to run XBrowser88, here's what you need:

- Windows 8 or Windows 7[1]
- Microsoft .NET Framework 4.5

Downloading XBrowser88

The most recent version of XBrowser88 can always be found at the X1s.org website, and can be downloaded for free.

There are no installation processes to complete — the file you download is the file that you open when you want to use the application.

To download XBrowser88, follow these steps:

1. With your web browser, visit **www.x1s.org**.
2. In the 'Downloads' area, find XBrowser88.
3. Download the XBrowser88 application file and save it somewhere on your system.
4. Double-click the application file to start the program.

[1] Compatibility with earlier versions of Windows, or under emulation on other operating systems, cannot be guaranteed.

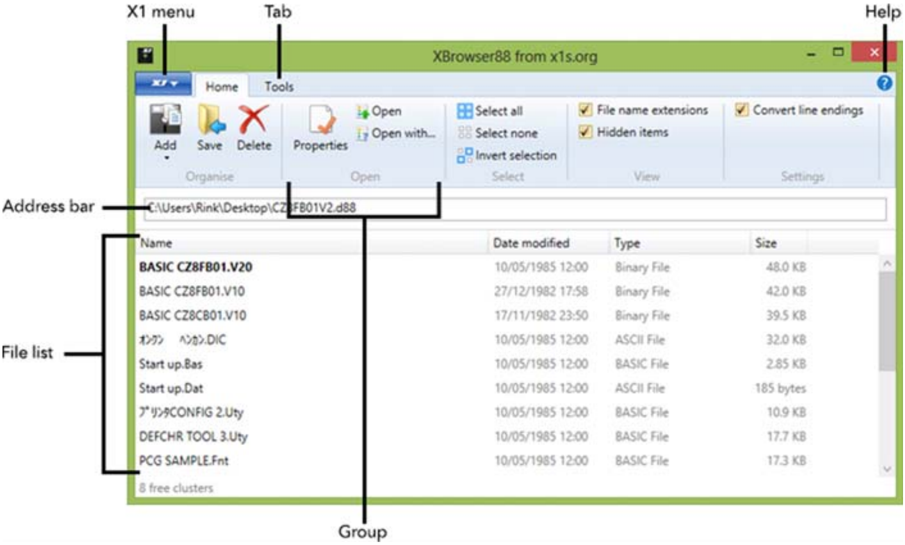
3 Working with D88 disk images

XBrowser88 is designed to work with Pasopia D88 floppy image format files, which commonly have the file extensions .d88 or .88d. Only images of 5.25-inch 320KB double-density floppy disks can be opened, and these disks should be formatted with Sharp-HuBASIC CZ-8FB01.

High-density disks and CZ-8FB02 features, such as directories, are not supported.

Use XBrowser88 to open floppy disk images and view a list of their contents. You can extract files from a disk image and save them to your Windows machine, add new files to the image, delete files, and change file properties. With XBrowser88, you can also set one file on the disk to be loaded by the X1's initial program loader (IPL) when the machine is booted.

Most functions are accessed from one window:



☑ File name extensions
☑ Hidden items
☑ Convert line endings

C:\Users\Rink\Desktop\CZ8FB01V2.d88

Name	Date modified	Type	Size
BASIC CZ8FB01.V20	10/05/1985 12:00	Binary File	48.0 KB
BASIC CZ8FB01.V10	27/12/1982 17:58	Binary File	42.0 KB
BASIC CZ8CB01.V10	17/11/1982 23:50	Binary File	39.5 KB
ｵﾝｸﾝ　ﾍﾝｶﾝ.DIC	10/05/1985 12:00	ASCII File	32.0 KB
Start up.Bas	10/05/1985 12:00	BASIC File	2.85 KB
Start up.Dat	10/05/1985 12:00	ASCII File	185 bytes
ﾌﾟﾘﾝﾀCONFIG 2.Uty	10/05/1985 12:00	BASIC File	10.9 KB
DEFCHR TOOL 3.Uty	10/05/1985 12:00	BASIC File	17.7 KB
PCG SAMPLE.Fnt	10/05/1985 12:00	BASIC File	17.3 KB

8 free clusters

Figure 1. XBrowser88 window

When you start the application, many options are unavailable until you open a floppy disk image.

Opening a disk image

There are three ways to open a Pasopia D88 floppy disk image in XBrowser88:

- With XBrowser88 active, on the **X1** menu, point to **Open** and then click **Open X1 image**.
- Drop a file onto the XBrowser88 window.
- From Windows Explorer, right-click a D88 floppy disk image, click **Open with**, and use the Windows dialog box to find the XBrowser88 application file.

XBrowser88 can also open .2d disk images in the same ways. However, it cannot manipulate the contents of these disk images, so you will be prompted to convert the file.

Opening a disk image in a new XBrowser88 window

To open a floppy disk image in a new window and keep the current file open, you should:

- On the **X1** menu, point to **Open**, and then click **Open X1 image in new window**.

Understanding the file list

Once a floppy disk image has been opened, the address bar is changed to show the volume name[1] of the disk image, and its location on your PC.

The number of free clusters on the disk is shown in the lower-left corner of the window. This number gives an indication of the amount of free space that is available on the disk.

The contents of the disk image are displayed in the file list.

File names which appear in bold are marked as the current boot file. For more information about bootable files, see “Making a file bootable” on page 10.

File names that appear dimmed are marked as hidden. They do not appear in a file list in Sharp-HuBASIC. To hide these files from the list in XBrowser88:

- On the **Home** tab, in the **View** group, clear the **Hidden items** check box.

File extensions can also be hidden from the list in XBrowser88. To do this:

- On the **Home** tab, in the **View** group, clear the **File name extensions** check box.

[1] Not all disk images will have a volume name.

Selecting a file

Many of XBrowser88's features require that a file (or multiple files) is selected in the file list.

To	Do this
Select an individual file	Click the file name.
Select multiple files in sequence	Hold down Shift while clicking the file names.
Select multiple files out of sequence	Hold down Ctrl while clicking the file names.

There are also three options on the ribbon for selecting files, and cancelling the selection:

To	Do this
Select all files in the file list	On the **Home** tab, in the **Select** group, click **Select all**.
Deselect all files currently selected	On the **Home** tab, in the **Select** group, click **Select none**.
Deselect all files currently selected, and select all files currently not selected	On the **Home** tab, in the **Select** group, click **Invert selection**.

Extracting and opening a file

Files can be extracted from the disk image either individually, or by selecting a group of files. For more information about selecting a file in the file list, see "Selecting a file" on page 5.

To extract files and choose a location where to save them, you can:

- Right-click the selected file(s) and click **Save as**.
- On the **Home** tab, in the **Organise** group, click **Save**.

When opening a file, it is saved to a temporary location on your PC and opened with the default Windows application that is associated with that file extension. If no default application is found, you will be prompted to select the application to use.

To open a file with the default application:

- Right-click the selected file and click **Open**.
- On the **Home** tab, in the **Open** group, click **Open**.

To open a file and choose the application to use:

- Right-click the selected file and click **Open with**.
- On the **Home** tab, in the **Open** group, click **Open with**.

When extracting or opening an ASCII text file, end-of-line markers will be converted to Windows format if the **Convert line endings** setting is enabled. For more information about line endings, see "Adding and replacing a file" on page 6.

Please Note:
After opening a file and editing it, saving the file will not update the disk image. You should save the file to a convenient location, and then see "Adding and replacing a file" on page 6 for more information about updating the contents of a disk image.

Adding and replacing a file

Sharp-HuBASIC supports three different types (modes) of file:

File mode	Description
ASCII	An ASCII-encoded text file. This might be data or a BASIC program.
BASIC	A BASIC program which has been saved in a tokenised format, and is not human-readable.
Binary	An executable file, or data that is not suitable for editing in HuBASIC.

Windows determines file types based on file extensions, but Sharp-HuBASIC maintains a separate 'file mode' attribute for each file. The file extension is usually irrelevant.

There are several ways of adding or replacing a file in a disk image. The first of these does not specify the file mode:

- Drop a file (or multiple files) onto the XBrowser88 window.
- On the **Home** tab, in the **Organise** group, click **Add**.

When overwriting an existing file, XBrowser88 will keep the file mode of the existing file. When adding a new file, binary mode will be used.

Using binary mode will not be suitable when adding a new BASIC or ASCII file to the disk image. It is recommended that you specify the file mode.

To add or replace a file by specifying the file mode:

- On the **Home** tab, in the **Organise** group, click the arrow below the **Add** icon, and then click the file mode that you want to use.

XBrowser88 can automatically convert the line endings of ASCII text files so that the files can be opened correctly in text editors, such as Microsoft Notepad. However, in certain circumstances you may want to disable this feature.

To disable automatic line conversions:

- On the **Home** tab, in the **Settings** group, clear the **Convert line endings** check box.

This setting takes effect when files are extracted or opened from the disk image, and when files are added or replaced in the disk image.

Deleting a file

You can delete files from the disk image either individually, or by selecting a group of files. For more information about selecting a file in the file list, see "Selecting a file" on page 5.

To delete a file, select it in the file list, and then there are two ways of deleting the file:

- Right-click the selected file and click **Delete**.
- On the **Home** tab, in the **Organise** group, click **Delete**.

Deleting a file is permanent. You cannot restore deleted files with XBrowser88.

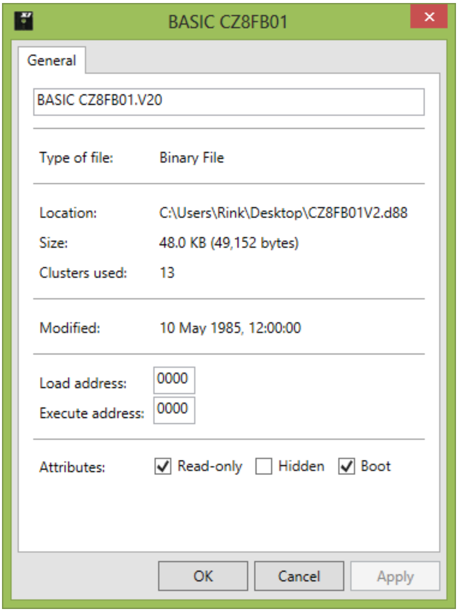
Changing the properties of a file

The Properties window displays all of the information known about a file.

From this window you can also: rename files, change the load and execute addresses, mark the file as hidden, and protect the file from accidental modifications or deletion.

To view the properties of a file, first select it in the file list, and then there are two ways of opening the Properties window:

- Right-click the file name and click **Properties**.
- On the **Home** tab, in the **Open** group, click **Properties**.



BASIC CZ8FB01.V20

Type of file: Binary File

Location: C:\Users\Rink\Desktop\CZ8FB01V2.d88

Size: 48.0 KB (49,152 bytes)

Clusters used: 13

Modified: 10 May 1985, 12:00:00

Load address: 0000

Execute address: 0000

Attributes: ☑ Read-only □ Hidden ☑ Boot

Figure 2. Properties window

The Properties window shows the file name in an editable text box near to the top of the window. Click inside the text box to begin renaming a file. You can enter up to 17 characters in this box — 13 characters make up the first part of the file name, and the remaining four characters make up the optional file extension (including the dot). Use JIS X 0201 encoding to enter Japanese characters.

If you type more than 13 characters for the first part of the file name, or type more than four characters for the extension, then XBrowser88 shortens the file name for you.

The load and execute address boxes are also editable. Any values that you type in to these boxes should be 16-bit hexadecimal numbers, to a maximum of four characters.

Three file attributes are shown as check boxes near to the bottom of the window:

To	Do this
Protect a file from modifications or deletion in Sharp-HuBASIC	Select the **Read-only** check box.
Hide a file from file lists in Sharp-HuBASIC	Select the **Hidden** check box.
Mark a file as bootable	Select the **Boot** check box. For more information, see "Making a file bootable" on page 10.

Finally, there are three buttons at the bottom of the window:

To	Do this
Discard any changes that you have made and close the window	Click **Cancel**.
Save any changes that you have made	Click **Apply**.
Save any changes that you have made and close the window	Click **OK**.

Making a file bootable

The boot sector of a Sharp X1 floppy disk contains information that points to which (if any) sector on the disk contains a startup program. The initial program loader (IPL) loads and executes this program when the system is booted.

Some floppy disks place this bootable code in an area of the disk that is reserved by the system. XBrowser88 cannot extract or replace code stored in these sectors. However, it is also possible to tell the IPL to load a startup program from the main file area of the disk, which XBrowser88 is able to work with.

To instruct the IPL to load and execute a file when the system is booted:

1. Select the file to be made bootable in the file list.
2. Do one of the following:

- Right-click the selected file and click **Properties**.
- On the **Home** tab, in the **Open** group, click **Properties**.

3. In the Properties window, select the **Boot** check box.
4. Click **OK** to save the change and close the window.

Only one file can be bootable at any one time. When you mark a file as bootable, any file that was previously marked this way is immediately made non-bootable.

It also acceptable for a disk not to have any bootable files. For example, disks that contain only data files for use with other programs.

As a general rule, only mark binary files as bootable. The IPL always tries to execute the file, even if it is a BASIC program or ASCII text file.

4 Converting a disk image

XBrowser88 has several features for creating and converting Pasopia D88 floppy disk images. These features can help you: work with files downloaded from the Internet, test your own programs in Sharp X1 emulators, and transfer disk images to SD cards for use with an HxC Floppy Disk Emulator.

Creating a blank disk

XBrowser88 can create a blank D88 floppy disk image that is ready to use with Sharp-HuBASIC.

To create a blank disk image:

1. On the **Tools** tab, in the **New** group, click **Disk**.

2. Save the new D88 disk image to a convenient location.

Converting from .2d disk images

.2d disk images are an older format. However, it is still common to find Sharp X1 floppy disks distributed online in this format.

XBrowser88 can not directly manipulate the contents of these files, they must first be converted to D88 disk images.

To convert a .2d image to Pasopia D88 format and open it:

1. Open the .2d image in XBrowser88. For more information about opening a disk image, see "Opening a disk image" on page 3.

2. Save the new D88 disk image to a convenient location.

To convert a .2d image to Pasopia D88 format without opening it:

1. On the **Tools** tab, in the **Conversions** group, click **From .2D**.

2. Choose the .2d file that you would like to convert.

3. Save the new D88 disk image to a convenient location.

Converting to HxC Floppy Emulator format (.hfe)

XBrowser88 can convert the open D88 disk image to the disk image format used by HxC Floppy Emulators.

This feature is unavailable until a D88 floppy disk image has been opened in XBrowser88. For information about opening a disk image, see “Opening a disk image” on page 3.

To convert the open disk image to HxC Floppy Emulator format:

1. On the **Tools** tab, in the **Conversions** group, click **To .HFE**.

2. Save the new .hfe file to an SD card or elsewhere on your system.

Appendix I

Sharp-HuBASIC file system

This document describes the format of the Sharp-HuBASIC floppy disk file system, based on a 5.25-inch 320KB, double-sided, double-density floppy disk as used by Sharp-HuBASIC CZ-8FB01. The file system itself supports other disk types and sizes.

Please note:
This document represents the state of the investigation by the X1s.org team, at the time of writing. Information may be missing or factually incorrect.

Overview

A disk formatted by Sharp-HuBASIC is divided into clusters, and their size is determined by the size of a sector and the number of sectors on each cylinder.

For example, on a standard 5.25-inch 320KB floppy disk that has been formatted by Sharp-HuBASIC, there are 80 cylinders per side. Sharp-HuBASIC defines a cluster as one cylinder, with each cluster containing 16 sectors. Each sector is 256 bytes. This gives a cluster size of 4KB.

Clusters alternate between sides of the disk. For example, the first cluster is on cylinder 0, head 0. The next cluster is on cylinder 0, head 1. Converting from a cluster number to a cylinder and head reference can be done by taking the least-significant bit of the cluster number as the head, and the most significant 7 bits represent the cylinder number.

Clusters 0 and 1 are reserved by the system, and contain the boot sector, file allocation table (FAT), and root directory. The remaining clusters are available for use.

Data types

Four data types are used here to describe information in the Sharp-HuBASIC file system:

Data type	Length (bytes)	Description
Char	1	An unsigned byte representing an ASCII character.

Data type	Length (bytes)	Description
Byte	1	An unsigned number stored in a single byte.
Word	2	An unsigned 16-bit number.
DateTime	6	For more information about the format of date and time values, see “Dates and times” on page 18.

In this description, data types that are marked with square brackets indicate a string or array — a sequence of values. The number of items in the sequence is shown by the value between the brackets.

Multi-byte data types are stored with little-endian ordering.

Boot sector

The first physical sector on the disk (cylinder 0, head 0, sector 1) is known as the boot sector. The initial program loader (IPL) uses this sector to load and execute a program from the floppy disk when the system is booted.

The boot sector contains a structure 32 bytes long, and this structure describes the program to be executed:

Name	Data type	Description
Boot flag	Byte	Always 0x01 for bootable floppy disks.
Startup label	Char[13]	The 13 character string that is displayed by the IPL when loading. Use JIS X 0201 encoding.
File extension	Char[4]	Always ‘Sys’ padded with a space character (0x20).
Size	Word	Size (in bytes) of the program to load.

Name	Data type	Description
Load address	Word	Location in memory where the data from the floppy disk should be copied to.
Execute address	Word	The memory address where program execute should begin.
Modified date	DateTime	Six bytes detailing the last modified date of the program.
Start sector	Word	The location (as a sector offset) of the file to be loaded and executed.

This boot sector format is used for all floppy disks compatible with the Sharp X1, even those that have not been formatted as a Sharp-HuBASIC disk.

Note that there is no requirement for the boot sector to refer to a file that is present in the root directory. It is acceptable for small executables to be stored in the first cluster between the boot sector and the file allocation table — making them invisible to Sharp-HuBASIC.

File allocation table

The file allocation table (FAT) is stored on sectors 15 and 16 of the first cluster. It contains a map of which clusters are in use by files, and which clusters are free.

On a 5.25-inch 320KB disk formatted by Sharp-HuBASIC, there are 80 entries in the FAT — one for each cluster on the disk.

There are several possible values for each entry:

- Zero indicates that the cluster is unused by the file system and is free space.
- A value 0x80–0x8F indicates that a file terminates in this cluster. The second nibble (0x0–0xF) indicates the number of sectors in this cluster that have been used by the file.
- Any other value points to the next cluster used by a file that is too large to fit in only one cluster.

The first two entries in the FAT represent two clusters that are reserved by the system. These values are typically 0x01 and 0x8F.

Root directory

The root directory occupies the entirety of the second cluster, and consists of a list of files and directories that are stored within. As Sharp-HuBASIC CZ-8FB01 has no support for directories, the root directory contains references for all accessible files on the disk.

Each entry in the directory is represented by a structure 32 bytes long:

Name	Data type	Description
File mode / attributes	Byte	For more information about file modes and file attributes, see "File modes and attributes" on page 17.
File name	Char[13]	The 13 character name of the file. Use JIS X 0201 encoding.
File extension	Char[4]	A three character extension, terminated by a space (0x20). Use JIS X 0201.
Size	Word	Size (in bytes) of the file.[a]
Load address	Word	Location in memory where executable files should be stored when loaded.[b]
Execute address	Word	The memory address where executable files should begin executing from.[c]
Modified date	DateTime	Six bytes detailing the last modified date of the file.
Start cluster	Word	The cluster where the start of the file is stored.

a. ASCII files may have an incorrect size — loading continues until the character 0x0D is followed by character 0x1A.

b. Load addresses are typically 0x0000 for BASIC and ASCII files, as Sharp-HuBASIC exercises control over the loading.

c. Execute addresses are typically 0x0000 for BASIC and ASCII files, as they are not executable binaries.

The directory continues across the sectors in the cluster until a file mode of 0xFF is reached.

Each file or directory on the disk is allocated its own cluster. This means that there is an effective limit on the number of files that can be stored on a disk. A Sharp-HuBASIC 80-track disk, with 80 clusters, can hold a maximum of 80 files.

Directories

Sharp-HuBASIC CZ-8FB01 is able to recognise the existence of directories, but has no commands for processing them or accessing their contents. Full support for directories is included with Sharp-HuBASIC CZ-8FB02 for the X1 Turbo.

When a directory is created, a cluster is allocated and this contains a new directory list following the same 32 byte format as described earlier.

File modes and attributes

File modes in Sharp-HuBASIC govern how a file can be opened, and by which command. In the file system, the mode (and several flags indicating file attributes) are encoded into a single byte value.

Values of zero typically indicate that the file has been deleted, and its entry in the directory can be overwritten.

For non-zero values, the high nibble defines the file's attributes:

Bit number	7	6	5	4
Function	Directory	Read-only	Verify[a]	Hidden

a. The Verify attribute appears to have been included to maintain compatibility with cassette BASIC, where a file could be marked and would be verified automatically when it was next saved.

The low nibble defines the file's mode:

Bit number	3	2	1	0
Function	ASCII[a]	ASCII	BASIC	Binary

a. Whilst setting bit 2 (0x04) is the preferred value for indicating ASCII files, bit 3 can also be set to achieve the same effect.

When Sharp-HuBASIC examines a file's mode, it first checks if bit 0 is set. If it is, the file is treated as binary irrespective of any other values in the low nibble. Next, if bit 1 is set then the file is treated as a BASIC program, and the setting of either of the ASCII bits is ignored.

Dates and times

Last modified dates on Sharp-HuBASIC disks contain bytes representing the date, the day of the week, and the time.

Most values are stored as binary coded decimal (BCD), and their appearance when viewed as a hexadecimal number provides an accurate visual representation. For example, the decimal number 85 is encoded with BCD and appears in hexadecimal as 0x85.

Name	Data type	Description
Year	Byte	Two digits representing the year.[a]
Month + day of the week	Byte	Packs the month and the day of the week into a single value. The high nibble is the number of the month[b], and the lower nibble is the day of the week.[c]
Day	Byte	The day of the month.
Hours	Byte	The hour of the day in 24-hour time notation.
Minutes	Byte	The minutes of the hour.
Seconds	Byte	The seconds of the minute.

[a] XBrowser88 assumes values below 80 refer to the 21st century, and values above 79 refer to the 20th century.

[b] Values range from 1 (January) through 12 (December).

[c] Values range from 0 (Sunday) through 6 (Saturday).

Appendix II
D88 disk image format

The Pasopia D88 format is a way of saving the contents and structure of a soft-sectored floppy disk to a single file, commonly with the file extensions .d88 or .88d.

It takes its name from the system that it was designed to work with, the Toshiba Pasopia 7.

It is not compressed — making it simple for emulators and disk utilities to work with — but some disk features are not preserved, and so some copy-protected disk formats may not be interpreted correctly.

A D88 floppy disk image consists of a 32-byte file header, an array of track offsets, and then a sector block for each sector on the disk.

Data types

Four data types are used in the D88 disk image format:

Data type	Length (bytes)	Description
Char	1	An unsigned byte representing an ASCII character.
Byte	1	An unsigned number stored in a single byte.
Word	2	An unsigned 16-bit number.
Dword	4	An unsigned 32-bit number.

In this description of the D88 disk image format, data types marked with square brackets indicate a string or array — a sequence of values. The number of items in the sequence is shown by the value between the brackets.

Multi-byte data types are stored with little-endian ordering.

File header

The D88 file header contains structural information about the disk that the file represents. For a 5.25-inch 320KB Sharp X1 disk, this information is often the same for each file.

The header begins on the first byte of a Pasopia D88 file.

Name	Data type	Description
Image name	Char[17]	17 characters indicating the volume name of the disk image and terminated with a zero. This field is rarely used.
Reserved	Byte[9]	Nine reserved bytes. Typically all set to zero.
Protect	Byte	0x10 (or any non-zero value) indicates that the disk is write protected.
Media type	Byte	0x00 - 2D 0x10 - 2DD 0x20 - 2HD 0x30 - 1D[a] 0x40 - 1DD[b]
Disk size	Dword	Four bytes containing the size of the disk image (including all header information).

[a] The media type 1D may not be recognised by all software.

[b] The media type 1DD may not be recognised by all software.

Track offsets

To help software process the disk image quickly, the Pasopia D88 file format includes an array of track offsets.

The track offsets array begins at offset 0x20 from the start of the file, and is 656 bytes long. It includes 164 dword values, regardless of whether the current disk image requires this many. If the disk is unformatted, only the first value in the array will be non-zero.

For example, a 5.25-inch 320KB Sharp X1 disk contains 40 tracks on each side of the disk. The track offset array contains values for the first 80 dwords

— each one pointing to where a track begins, calculated as an offset from the beginning of the image file. The remaining 84 offsets are simply set to zero.

Sector blocks

Each sector on the disk is stored as a block. The block consists of a sector header, and this is followed by the actual byte values from the disk. The length of a sector block will vary in size between different disk images, depending on the size of a sector on the disk.

Sector headers consist of 16 bytes as described below:

Name	Data type	Description
Cylinder (C)	Byte	Cylinder number.
Head (H)	Byte	Head/side number.
Sector (R)	Byte	Sector/record number.
Sector size (N)	Byte	The number of bytes in a sector, stored as 128 << N.
Number of sectors	Word	The number of sectors in a track. This value should be the same in all the sectors of that track.
Density	Byte	0x00 - Double density 0x40 - Single density
DDAM	Byte	Deleted-data address-mark (DDAM) status code. 0x00 is normal.
FDC status	Byte	Floppy disk controller (FDC) status code. 0x00 is normal.
Reserved	Byte[5]	Five reserved bytes. Typically all set to zero.
Actual size	Word	Actual number of bytes following the sector header. This value will usually be the same as the sector size, but it is not guaranteed.

It is possible for a sector header to appear in an image file with zero bytes following it. In this case, the sector size may indicate the correct size of a sector, but the actual size will be zero.

Sectors will usually be written to a disk image file in the order that they are read from the physical disk. They are not guaranteed to appear in the disk image in an order based on their cylinder, head or sector numbers.

X1S.ORG

Information in this document is subject to change without notice and does not represent a commitment on the part of X1s.org. The software described in this document is provided under a license agreement. It is against the law to copy this software except as specifically allowed in the agreement. No part of this document may be reproduced or transmitted, except as part of an unmodified distribution package, without the express written permission of X1s.org.