

Zip

Welcome to 7-Zip 18.00 beta

7-Zip is a file archiver with a high compression ratio.

Free documentation for 7-Zip includes:

- [General information about 7-Zip](#)
- [User's Guide for 7-Zip File Manager](#)
- [User's Guide for command line version](#)

Free Also

- [Frequently Asked Questions \(FAQ\)](#)

Copyright (c) 1999-2018 Igor Pavlov

Web site: www.7-zip.org

Support: www.7-zip.org/support.html

General Information

The main features of 7-Zip

- [Powerful file manager](#)
- [High compression ratio and high speed](#)
- [Big number of supported archive formats](#)
- [Additional command line version](#)

See Also

- [License for use and distribution](#)

Supported formats

Format	Creation	Filename Extensions
7z	X	7z
BZIP2	X	bz2 bzip2 tbz2 tbz
GZIP	X	gz gzip tgz
TAR	X	tar
WIM	X	wim swm esd
XZ	X	xz txz
ZIP	X	zip zipx jar xpi odt ods docx xlsx epub
APM		apm
AR		ar a deb lib
ARJ		arj
CAB		cab
CHM		chm chw chi chq
COMPOUND		msi msp doc xls ppt
CPIO		cpio
CramFS		cramfs
DMG		dmg
Ext		ext ext2 ext3 ext4 img
FAT		fat img
HFS		hfs hfsx
HXS		hxs hxi hxr hxq hxw lit
iHEX		ihex
ISO		iso img
LZH		lzh lha
LZMA		lzma
MBR		mbr
MsLZ		mslz
Mub		mub
NSIS		nsis
NTFS		ntfs img

MBR	mbr
RAR	rar r00
RPM	rpm
PPMD	ppmd
QCOW2	qcow qcow2 qcow2c
SPLIT	001 002 ...
SquashFS	squashfs
UDF	udf iso img
UEFIc	scap
UEFIs	uefif
VDI	vdi
VHD	vhd
VMDK	vmdk
XAR	xar pkg
Z	z taz



7-Zip creates ZIP compatible archives. 7-Zip supports the following ZIP compression methods:

- 0 - Store
- 1 - Shrink (decompression only)
- 6 - Implode (decompression only)
- 8 - Deflate
- 9 - Deflate64
- 12 - BZip2
- 14 - LZMA
- 95 - xz (decompression only)
- 98 - PPMd
- 99 - WinZip AES

Files compressed with other ZIP compression methods can't be extracted by the current version of the 7-Zip. But these supported methods are the most popular today, and therefore 7-Zip can decompress most ZIP archives. To extract files compressed with non-supported methods you must use some other ZIP utility.

7-Zip supports the Zip64 extension of ZIP format.

.H

7-Zip supports LZH archives only for listing, browsing and decompressing. 7-Zip supports -lh0-, -lh4-, -lh5-, -lh6- and -lh7- methods.

: Format

7z is a new archive format, providing a high compression ratio.

The main features of the **7z** format:

- Open architecture
- High compression ratio
- Strong AES-256 encryption
- Ability to use any compression, conversion or encryption method
- Supports files with sizes up to 16000000000 GB
- Unicode file names
- Solid compression
- Archive headers compression

7z has an open architecture, so it can support any new compression methods.

The following methods currently are integrated into **7z**:

Method	Description
LZMA	Improved and optimized version of LZ77 algorithm
LZMA2	LZMA-based compression method. It provides better multithreading support than LZMA
PPMD	Dmitry Shkarin's PPMdH with small changes
BCJ	Converter for 32-bit x86 executables
BCJ2	Converter for 32-bit x86 executables
BZip2	Standard BWT algorithm
Deflate	Standard LZ77-based algorithm

LZMA is the default and general compression method of **7z** format. The main features of the **LZMA** method:

- High compression ratio
- Variable dictionary size (up to 4 GB)

- Compression speed: about 1 MB/s on 2 GHz CPU
- Decompression speed: about 10-20 MB/s on 2 GHz CPU
- Small memory requirement for decompression (depends from dictionary size)
- Small code size for decompression: about 5 KB
- Supports multi-threading and P4's hyper-threading

The **LZMA** compression algorithm is very suitable for embedded applications. If you want to use **LZMA** code, you can ask for consultation, custom code programming, and required developer licenses at

www.7-zip.org/support.html

IS encryption

7-Zip supports encryption with the AES-256 algorithm. This algorithm uses a cipher key with length of 256 bits. To create the key, 7-Zip uses a derivation function based on an SHA-256 hash algorithm. A key derivation function produces a derived key from a text password defined by the user. To increase the cost of an exhaustive search for passwords, 7-Zip uses a big number of iterations to produce the cipher key from the text password.

ns for selecting password length

Here is an estimate of the time required for an exhaustive password search attack, when the password is a random sequence of lowercase Latin letters.

The most complex task for password search attack is SHA-256 calculation. Special SHA-256 hardware or GPU can be used to accelerate password search attack. Now modern GPU can provide about 10 times more performance for SHA-256 calculation than modern CPU. And special SHA-256 hardware can provide about 20 times more performance than GPU.

We suppose that one user with a budget of about \$2000 (for GPUs) can check 10000 passwords per second and an organization with a budget of about 10^9 USD (one thousand million US dollars) can check $3 * 10^{12}$ passwords per second. We also suppose that the processor in use doubles its performance every two years; so, each additional Latin letter of a long password adds about 9 years to an exhaustive key search attack.

The result is this estimate of the time to succeed in an attack:

Password Length Single User Attack Organization Attack

1	1 s	1 s
2	1 s	1 s
3	2 s	1 s
4	1 min	1 s
5	30 min	1 s
6	12 hours	1 s
7	14 days	1 s
8	1 year	1 s
9	10 years	2 s
10	19 years	1 min
11	28 years	30 min
12	37 years	12 hours
13	46 years	14 days

14	55 years	1 year
15	64 years	10 years
16	73 years	19 years
17	82 years	28 years
18	91 years	37 years
19	100 years	46 years

Performance

We compared **7-Zip** with WinRAR 5.20.

FILE SETS: Mozilla Firefox 34.0.5 for Windows and Google Earth 6.2.2.6613 for Windows.

Archiver	Mozilla Firefox		Google Earth	
	65 files 85 280 391 bytes Compressed size	Ratio	483 files 110 700 519 bytes Compressed size	Ratio
7-Zip 9.35 -mx	39 357 375	100%	15 964 369	100%
WinRAR 5.20 -m5 -s -ma5 - md128m	41 789 543	106%	17 035 432	107%

Compression ratio results are very dependent upon the data used for the tests. Usually, **7-Zip** compresses to 7z format 30-70% better than to zip format. And **7-Zip** compresses to zip format 2-10% better than most of other zip compatible programs.

Frequently Asked Questions (FAQ)

Can I use 7-Zip in a commercial organization?

Yes, 7-Zip is free software. You can use it on any computer. You don't need to register or pay for 7-Zip.

How can I set file associations to 7-Zip in Windows 7 and Windows Vista?

You must run 7-Zip File Manager in administrator mode. Right-click the icon of 7-Zip File Manager, and then click **Run as administrator**. Then you can change file associations and some other options.

Why 7z archives created by new version of 7-Zip can be larger than archives created by old version of 7-Zip?

New versions of 7-Zip (starting from version 15.06) use another file sorting order by default for solid 7z archives.

Old version of 7-Zip (before version 15.06) used file sorting "by type" ("by extension").

New version of 7-Zip supports two sorting orders:

- sorting by name - default order.
- sorting by type, if **qs** is specified in **Parameters** field in "Add to archive" window, (or **-mqs** switch for command line version).

You can get big difference in compression ratio for different sorting methods, if dictionary size is smaller than total size of files. If there are similar files in different folders, the sorting "by type" can provide better compression ratio in some cases.

Note that sorting "by type" has some drawbacks. For example, NTFS volumes use sorting order "by name", so if an archive uses another sorting, then the speed of some operations for files with unusual order can fall on HDD devices (HDDs have low speed for "seek" operations).

You can increase compression ratio with the following methods:

- Increase dictionary size. It can help when 'qs' is not used.
- Specify 'qs' in **Parameters** field (or use **-mqs** switch for command line version).

If you think that unusual file order is not problem for you, and if better compression ratio with small dictionary is more important for you, use 'qs' mode.

Why can't 7-Zip open some ZIP archives?

In 99% of these cases it means that the archive contains incorrect headers. Other ZIP programs can open some archives with incorrect headers, since these programs just ignore errors.

If you have such archive, please don't call the 7-Zip developers about it. Instead try to find the program that was used to create the archive and inform the developers of that program that their software is not ZIP-compatible.

There are also some ZIP archives that were encoded with methods unsupported by 7-Zip, for example, WAVPack (WinZip).

Why does drag-and-drop archive extraction from 7-Zip to Explorer use temp files?

7-Zip doesn't know folder path of drop target. Only Windows Explorer knows exact drop target. And Windows Explorer needs files (drag source) as decompressed files on disk. So 7-Zip extracts files from archive to temp folder and then 7-Zip notifies Windows Explorer about paths of these temp files. Then Windows Explorer copies these files to drop target folder.

To avoid temp file usage, you can use Extract command of 7-Zip or drag-and-drop from 7-Zip to 7-Zip.

Why doesn't the command line version add files without extensions to archive?

You're probably using a *.* wildcard. 7-Zip doesn't use the operating system's wildcard mask parser, and consequently treats *.* as any file that has an extension. To process all files you must use the * wildcard

instead or omit the wildcard altogether.

Why doesn't -r switch work as expected?

In most cases you don't need -r switch. 7-Zip can compress subfolders even without -r switch.

Example 1:

```
7z.exe a c:\a.7z "C:\Program Files"
```

compresses "C:\Program Files" completely, including all subfolders.

Example 2:

```
7z.exe a -r c:\a.7z "C:\Program Files"
```

searches and compresses "Program Files" in all subfolders of C:\ (for example, in "C:\WINDOWS").

If you need to compress only files with some extension, you can use -r switch:

```
7z a -r c:\a.zip c:\dir\*.txt
```

compresses all *.txt files from folder c:\dir\ and all its subfolders.

Why can't 7-Zip use big dictionary in 32-bit Windows?

32-bit Windows allocates only 2 GB of virtual space per one application. Also that block of 2 GB can be fragmented (for example, by some DLL file), so 7-Zip can't allocate one big contiguous block of virtual space. There are no such limitations in 64-bit Windows. So you can use any dictionary in Windows x64, if you have required amount of physical RAM.

How can I install 7-Zip in silent mode?

For exe installer: Use the "/S" parameter to do a silent installation and the "/D=dir" parameter to specify the "output directory". These options are case-sensitive.

For msi installer: Use the /q INSTALLDIR="C:\Program Files\7-Zip" parameters.

How can I recover corrupted 7z archive?

There are some possible cases when archive is corrupted:

- You can open archive and you can see the list of files, but when you press Extract or Test command, there are some errors: Data Error or CRC Error.
- When you open archive, you get message "Can not open file 'a.7z' as archive"

It's possible to recover some data. Read about recovering procedure:

[Recover corrupted 7z archive](#)

cense

7-Zip Copyright (C) 1999-2018 Igor Pavlov.

Licenses for files are:

- 7z.dll: GNU LGPL + unRAR restriction
- All other files: GNU LGPL

The GNU LGPL + unRAR restriction means that you must follow both GNU LGPL rules and unRAR restriction rules.

Note: You can use 7-Zip on any computer, including a computer in a commercial organization. You don't need to register or pay for 7-Zip.

Read file License.txt for full information about license.

anks

I would like to thank:

- myspace from sf.net - for porting to Unix/Linux (p7zip), testing, bug fixing and more.
- ARJ Software - for ARJ.
- Eric Biggers - for LZMS description (wimlib).
- Wei Dai - for SHA-256 code and Crypto++ Library.
- Jeff Gilchrist - for compressing tests.
- Szymon Grabowski - for some ideas.
- Microsoft Corporation - for CAB/WIM formats descriptions, and WiX (Windows Installer XML).
- Nullsoft, Inc. and NSIS team - for NSIS (Nullsoft Scriptable Install System).
- Robert Martinez - 7-Zip Logo.
- Alexander Ratushnyak - for compressing tests.
- Eugene Roshal - for RAR.
- Matthew Russotto - for CHM/CAB format description.
- Julian Seward - for BZip2.
- Eugene Shelwien - for some ideas and algorithms.
- Dmitry Shkarin - for PPMd algorithm.
- Yoshioka Tsuneo - for TAR32 program.
- Vadim Yoochin - for compressing tests.
- Bulat Ziganshin - for some ideas.
- John Michael Williams - for help file corrections.
- Apple Inc. - for HFS.

- vu1tur(DMG2IMG) - for .DMG hints.
- Yutaka Sawada - for hints to ZIP Strong Encryption format.
- NTFS-3G team - for hints to NTFS.
- Phillip Lougher - for SquashFS.
- Haruyasu Yoshizaki - for LHA/LZH.

Zip File Manager

The 7-Zip File Manager is a program for manipulating files and folders.

The 7-Zip File Manager can work with two panels. You can switch between panels by pressing the Tab button. Most of the operations can be executed using keyboard shortcuts or by right-clicking on items and selecting the appropriate command from menu.

This Section

[Menu Items and Shortcut Keys](#)

Describes the menu items and keyboard shortcut.

[Options Dialog Box](#)

Describes what settings you can change that affect the 7-Zip File Manager.

[Benchmark](#)

Describes a 7-Zip Benchmark for measuring CPU performance.

[About Dialog Box](#)

Provides information about 7-Zip.

[Plugins](#)

Describes plugins for the 7-Zip File Manager.

You have access to the following items from root folder in 7-Zip File Manager:

- **Computer:** computer's disks
- **Documents:** My Documents folder of current user
- **Network**
- **\\.** : the folder for low-level access to disks. You must have administrator's rights to use that folder. You can

- copy .iso images from CD/DVD to file.
- view NTFS/FAT partitions via 7-Zip parser. It provides read-only access to files.

ptions Dialog Box

You can change many of the settings that affect the 7-Zip File Manager in the Options dialog box. To access this dialog box select Options from the Tools menu.

The options dialog box contains the following pages:

- [System Page](#)
- [7-Zip](#)
- [Folders Page](#)
- [Editor Page](#)
- [Settings Page](#)
- [Language Page](#)

System Page

Allows you to specify the behavior of 7-Zip in the filing system.

Associate 7-Zip with

Allows to associate 7-Zip with file name extensions. There are two list of associations: for "Current User" and for "All Users". To set associations for "All Users" you must run 7-Zip File Manager with administrator rights. Each association can have up to 4 states:

- no association.
- AppName - the extension is associated with another program (not 7-Zip).
- [7-Zip] - the extension is associated with another copy (edition) of 7-Zip.
- 7-Zip - the extension is associated with this copy of 7-Zip.

You can switch between these states by clicking on cell. Or you can change states for all extension by clicking on "+" button.

Zip Page

Allows you to specify behavior of 7-Zip in Windows Shell.

Integrate 7-Zip to shell context menu

Select this check box to add 7-Zip commands to Shell Context Menu. There are two check boxes in 64-bit system. So it's possible to enable 7-Zip context menu for 32-bit and 64-bit applications.

Cascaded context menu

Select this check box to group 7-Zip context menu items to one **7-Zip** submenu.

Icons in context menu

Select this check box to show 7-Zip icon in context menu items.

Eliminate duplication of root folder

That option allows to eliminate duplication of root folder for "Extract to" operations, if all files in archive are placed in folder with name that is identical to name of output folder.

Context menu items

Lists the optional context menu items. If an item is checked, 7-Zip will add this item to the context menu.

Folders Page

Allows you to set the folders which 7-Zip will use for internal purposes.

Working folder

The working folder section allows you to specify a folder which will be used for temporary archive files:

System temp folder

7-Zip will use the Windows temp folder.

Current

7-Zip will use the folder containing the target archive.

Specified

7-Zip will use the folder specified in the following edit control.

Use for removable drives only

Specifies whether 7-Zip should use the specified folder settings only for removable drives; for other drives, 7-Zip will create temporary archives in the folder containing the target archive. If you clear this check box, 7-Zip always will use the specified folder settings mode always.

7-Zip uses temporary archive files for all update operations. So, for speed reasons, it is recommended that you select the **System temp folder** option and select **Use for removable drives only** check box.

Editor Page

Allows you to specify settings for the editor.

View

Specifies the path to the viewer that will be used for "View" command (F3).

Editor

Specifies the path to the editor that will be used for "Edit" command (F4).

Diff

Specifies the path to a file comparison utility that shows the differences between two files.

Settings Page

Allows you to specify some settings.

Show .. item

Shows .. item in file list.

Show real file icons

Shows real file icons in file list. If this option is enabled, listing updates can be slower.

Full row select

When an item is selected, the item and all its subitems are highlighted.

Show grid lines

Displays gridlines around items and subitems.

Single-click to open an item

If enabled, File Manager opens files and folders by single-clicking them.

Alternative selection mode

If enabled, File Manager keeps selection mark when you move cursor.

Show system menu

Shows a system context sub-menu in the File menu.

Use large memory pages

If enabled, 7-Zip will try to use large pages. This feature allows an increase in speed of compression. This may cause 7-Zip to pause when starting compression, because of allocation of the large pages. Also, the Windows Task Manager doesn't show the real memory usage of the program, if 7-Zip uses large pages. This feature works only on Windows 2003 / XP x64. Also you must have administrator's

rights for your system. Recommended size of RAM: 1 GB or more.
To install this feature you must run 7-Zip File Manager at least once,
close it and reboot the system.

Language Page

Allows you to change the default language.

Language

Lists the available languages.

enchmark

This form allows you to measure the performance of your computer.

There are two tests:

1. Compression with LZMA method
2. Decompression with LZMA method

The benchmark shows a rating in MIPS (million instructions per second). The rating value is calculated from the measured speed, and it is normalized with results of Intel Core 2 CPU with multi-threading option switched off. So if you have modern CPU from Intel or AMD, rating values in single-thread mode must be close to real CPU frequency.

You can change the dictionary size to increase memory usage. Also you can change the number of threads.

The **CPU Usage** column shows the percentage of time the processor is working. It's normalized for a one-thread load. For example, 180% CPU Usage for 2 threads can mean that average CPU usage is about 90% for each thread.

The **Rating / Usage** column shows rating normalized for 100% of CPU usage. That column shows performance of the one CPU thread. It must be close to real CPU frequency, if you have modern CPU.

The **Total rating** shows averages of the compressing and decompression ratings.

Compression speed and rating strongly depend from memory (RAM) latency.

Decompression speed and rating strongly depend on CPU integer operations. For example, an Intel Pentium 4 has big branch misprediction penalty (which is effect of long pipeline) and pretty slow multiply and shift operations. So, the Pentium 4 has pretty low decompressing ratings.

Also the program checks possible errors. If the program shows some error message, in most cases it means that your RAM is defective. If so, don't use 7-Zip for compressing data, since such errors can lead to data

losses.

About Dialog Box

The About dialog box provides information about 7-Zip. This dialog box is available from the Help menu.

- Press **www.7-zip.org** button for opening 7-Zip Home Page.

ugins

The 7-Zip File Manager can use internal and external plugins.

Currently 7-Zip uses the following plugins:

7-Zip

Plugin for manipulating archives.

Zip Plugin

7-Zip is a plugin for manipulating archives. It makes all archive files look like usual folders in File Manager.

Most of the operations with compressed files can be initiated by right-clicking on files in the File Manager (or in Windows Explorer) and selecting the appropriate command from a menu.

ing 7-Zip

Opening archive

There are two ways to open an archive file:

- If you have the file type associated with 7-Zip, then you can open the file with 7-Zip by double-clicking the file or by right-clicking and selecting the **Open** command.
- You can right-click the archive file, point to **7-Zip**, and then click the **Open** command item.

Extracting archive

There are two ways to extract files from an archive:

- To extract all files from an archive right-click the archive file, point to **7-Zip**, and then click the **Extract files...** command item.
- To extract specific files from an archive, open it by [Open archive with 7-Zip](#), select the items to extract, and run the **Copy To...** command.

[Extract Dialog](#) will appear.

Testing archive

To test an archive, right-click the archive file, point to **7-Zip**, and then click the **Test archive** command item.

Creating and updating files in an archive

For creating or updating an archive file, right-click the file(s) or folder(s) you want to compress, point to **7-Zip**, and then click the **Add to archive...** command item. [Add to Archive Dialog](#) will appear.

Add to Archive Dialog Box

Allows you to specify options for creating or updating an archive.

How to call this dialog box

1. In Windows Explorer or in 7-Zip, right-click the file(s) or folder(s) you want to compress.
2. Point to **7-Zip**, and then click the **Add to archive...** command item.

Parameters

Archive

Provides a space for you to specify a destination archive name. You can click "..." button to display "Open" dialog box that you can use to locate archive.

Archive format

Specifies a format of created archive. Some formats (gzip and bzip2) do not support compressing more than one file per archive.

Compression level

Specifies compression level. There are 6 levels of compression:

Value	Meaning
Store	Files will be copied to archive without compression.
Fastest	Fastest compression.
Fast	Fast compression.
Normal	Compression with balanced settings.
Maximum	Can give a higher compression ratio than Normal level. But it can be slower, and it can require more memory.
Ultra	Can give a higher compression ratio than Maximum level. But it can be slower, and it can require more memory.

Compression method

Specifies compression method. Each archive format can have its own compression methods:

Method	Description
LZMA	It's base compression method for 7z format. Even old versions of 7-Zip can decompress archives created with LZMA method. It provides high compression ratio and very fast decompression.
LZMA2	Default compression method of 7z format. LZMA2 is LZMA-based compression method. It provides better multithreading support than LZMA. But compression ratio can be worse in some cases. For best compression ratio with LZMA2 use 1 or 2 CPU threads. If you use LZMA2 with more than 2 threads, 7-zip splits data to chunks and compresses these chunks independently (2 threads per each chunk).
PPMd	Dmitry Shkarin's PPMdH algorithm with small changes. Usually it provides high compression ratio and high speed for text files.
BZip2	Standard compression method based on BWT algorithm. Usually it provides high speed and pretty good compression ratio for text files.
Deflate	Standard compression method of ZIP and GZip formats. Compression ratio is not too high. But it provides pretty fast compressing and decompressing. Deflate method supports only 32 KB dictionary.
Deflate64	Modified version of Deflate algorithm with bigger dictionary (64KB).

Estimated 7-Zip performance and memory requirements for 2 GHz dual core CPU:

Method	Level	Dictionary Size	Compressing Speed	Decompressing Speed	Mem Comp
	fastest	64 KB	4.5 MB/s		3 MB
	fast	1 MB	3 MB/s		10 ME
LZMA	normal	16 MB	2 MB/s	15 MB/s	186 M
	maximum	32 MB	1.8 MB/s		376 M

	ultra	64 MB	1.6 MB/s		709 M
	fast	4 MB	1.4 MB/s		6 MB
PPMD	normal	24MB	1.2 MB/s		26 ME
	maximum	64 MB	1.0 MB/s		66 ME
	ultra	192 MB	0.9 MB/s		194 M
	fast		15 MB/s		3 MB
Deflate	normal		3.5 MB/s	40 MB/s	
	maximum	32 KB	1.5 MB/s		4MB
	ultra		0.4 MB/s		
	normal		3 MB/s		
BZip2	maximum	900 KB	1.2 MB/s	16 MB/s	20 ME
	ultra		0.4 MB/s		

Dictionary size

Specifies Dictionary size for compression method.

Usually, a higher Dictionary size gives a higher compression ratio. But compressing can be slower and it can require more memory.

Memory (RAM) usage for LZMA compressing is about 11 times more than dictionary size. Memory usage for LZMA decompressing is close to value of dictionary size. Memory usage for PPMd compressing and decompressing is almost equal to dictionary size.

Word size

Specifies the length of words, which will be used to find identical sequences of bytes for compression.

Usually for LZMA and Deflate, big Word size gives a little bit better compression ratio and slower compression process. A big Word size parameter can significantly increase compression ratio for files which contain long identical sequences of bytes. For PPMd, the Word size strongly affects both compression ratio and compression/decompression speed.

Solid Block size

Specifies the size of a solid block. You can also disable solid mode.

In solid mode all files will be compressed as continuous data blocks. Usually compressing to a solid archive improves the compression ratio. You can use this option only for 7z archives. The updating of solid .7z archives can be slow, since it can require some recompression.

Number of CPU threads

Specifies the number of threads for compressing. A big number of threads can speed up compression speed on Multi-Processor systems. Sometimes it can increase speed even on single-core CPU.

Split to volumes

`{Size}[b | k | m | g]`

Specifies volume sizes in Bytes, Kilobytes (1 Kilobyte = 1024 bytes), Megabytes (1 Megabyte = 1024 Kilobytes) or Gigabytes (1 Gigabyte = 1024 Megabytes). If you specify only {Size}, 7-zip will treat it as bytes. It's possible to specify several values. Example:

`10k 15k 2m`

The first volume will be 10 KB, the second will be 15 KB, and all others will be 2 MB.

Parameters

Allows you to specify parameters for compression. See the [-m \(Method\)](#) switch description for more details. Omit the -m prefix (as in -m switch) when using this dialog box.

Examples

`f=delta:4`

uses Delta:4 filter (if you want to compress WAV files).

`f=bcj2`

uses BCJ2 filter (for x86 executables).

Update mode

Specifies update mode:

Value	Meaning
Add and replace files	Add all specified files to the archive.
Update and add files	Update older files in the archive and add files that are new to the archive.
Freshen existing files	Update specified files in the archive that are older than the selected disk files.
Synchronize files	Replace specified files only if added files are newer. Always add those files, which are not present in the archive. Delete from archive those files, which are not present on the disk.

Path mode

Specifies how path names will be stored in archive:

Value	Meaning
Relative pathnames	Store file paths relative to current folder.
Full pathnames	Store file paths relative to root of the volume, excluding volume name prefix.
Absolute pathnames	Store fully qualified file paths including volume name.

Options

Specifies compression options:

Option	Meaning
Create SFX archive	Create self-extracting archive. You can use this option only for 7z archives. Look to -sfx (Create SFX archive) switch description for more details about SFX modules.
Compress shared files	Compress files open for writing by another applications.
Delete files after	Delete files after including to archive. So it works like moving files to archive. 7-Zip deletes

compression files at the end of operation and only if archive was successfully created.

Encryption

Specifies password and encryption options.

Enter password

Specify password here

Reenter password

Reenter password here for verification

Show Password

Shows Password

Encryption method

Specifies the encryption method. For 7z format, it can be only AES-256. For ZIP format you can select ZipCrypto or AES-256. Use ZipCrypto, if you want to get archive compatible with most of the ZIP archivers. AES-256 provides stronger encryption, but now AES-256 is supported only by 7-Zip, WinZip and some other ZIP archivers.

Encrypt file names

Enables or disables archive header encryption, including file name encryption.

Extract Dialog Box

Allows you to specify options for extracting files from archive.

How to run this dialog box

There are two ways to run this dialog to extract all files from an archive:

- In Windows Explorer or in 7-Zip, right-click the archive file, point to **7-Zip**, and then click **Extract files....**
- Select the archive file in 7-Zip and press **Extract** button.

Parameters

Extract to

Provides a space for you to specify an output folder. You can click "..." button to display "Browse for folder" dialog box that you can use to locate output folder.

Path mode

Specify extract mode:

Value	Meaning
Full pathnames	Extract files with full pathnames.
No pathnames	Extract files without folder paths.
Absolute pathnames	Extract files with absolute pathnames. 7-Zip doesn't use folder specified in "Extract to" field for files in archive that have absolute pathnames. Absolute path can include drive name.

Eliminate duplication of root folder

That option allows to eliminate duplication of root folder, if all files in archive are placed in folder with name that is identical to name of output folder.

Overwrite mode

Specify overwrite mode for files that already present on disk:

Value	Meaning
Ask before overwrite	Ask before overwriting existing files.
Overwrite without prompt	Overwrite existing files without prompt.
Skip existing files	Skip extracting of existing files.
Auto rename	Rename extracted files, if a file with the same name already exists. For example, file document.txt will be renamed to document_1.txt.
Auto rename existing files	Rename existing files, if a file with the same name already exists. For example, file document.txt will be renamed to document_1.txt.

Password

Specifies a password for encrypted archives.

Show Password

Show a password in Password field.

Restore file security

Restore file security information, if archive contains that information. That feature now is implemented only for WIM archives.

Command Line Version User's Guide

7z.exe is the command line version of 7-Zip. 7z.exe uses 7z.dll from the 7-Zip package. 7z.dll is used by the 7-Zip File Manager also.

7za.exe (a = alone) is a standalone version of 7-Zip. 7za.exe supports only 7z, lzma, cab, zip, gzip, bzip2, Z and tar formats. 7za.exe doesn't use external modules.

- [Command Line syntax](#)
- [Exit Codes](#)
- [Commands](#)
- [Switches](#)

Command Line Syntax

```
7z <command> [<switch>...] <base_archive_name> [<arguments>...]
```

```
<arguments> ::= <switch> | <wildcard> | <filename>
```

```
<switch> ::= -{switch_name} | @{listfile}
```

Expressions in square brackets (between '[' and ']') are optional.

Expressions in curly braces ('{' and '}') mean that instead of that Expression (including braces), the user must substitute some string.

Expression

```
expression1 | expression2 | ... | expressionN
```

means that any (but only one) from these expressions must be specified.

[Commands](#) and [switches](#) can be entered in upper or lower case.

Command is the first non-switch argument.

The "base_archive_name" must be the first filename on the command line after the command.

The switches and other filenames can be in any order.

Wildcards or filenames with spaces must be quoted:

```
"Dir\Program files\*"
Dir "\"Program files\"*
```

Switch options can be combined to save command line length. However, some switch options take optional string arguments and therefore, must be the last option in a combined argument token string because 7-Zip accepts the rest of the argument token as the optional argument.

7-Zip uses wild name matching similar to Windows 95:

- '*' means a sequence of arbitrary characters.
- '?' means any character.

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all

files, you must use a * wildcard.

Examples:

*.txt means all files with an extension of ".txt"

?a* means all files with a second character of "a"

1 means all names that contains character "1"

..* means all names that contain two at least "." characters

The default wildcard "*" will be used if there is no filename/wildcard in the command line.

Slash ('\') at the end of a path means a directory. Without a Slash ('\') at the end of the path, the path can refer either to a file or a directory.

st file

You can supply one or more filenames or wildcards for special list files (files containing lists of files). The filenames in such list file must be separated by new line symbol(s).

For list files, 7-Zip uses UTF-8 encoding by default. You can change encoding using [-SCS](#) switch.

Multiple list files are supported.

For example, if the file "listfile.txt" contains the following:

```
My programs\*.cpp  
Src\*.cpp
```

then the command

```
7z a -tzip archive.zip @listfile.txt
```

adds to the archive "archive.zip" all "*.cpp" files from directories "My programs" and "Src".

Short and Long File Names

7-Zip supports short file names (like FILENA~1.TXT) in some cases. However, it's strongly recommended to use only the real (long) file names.

Exit Codes from 7-Zip

7-Zip returns the following exit codes:

Code	Meaning
0	No error
1	Warning (Non fatal error(s)). For example, one or more files were locked by some other application, so they were not compressed.
2	Fatal error
7	Command line error
8	Not enough memory for operation
255	User stopped the process

Command Line Commands

The command is the first non-switch argument on the command line.

Command names are not case sensitive.

See also [Command Line Syntax](#) for more details about using the command line.

Commands quick reference

Command	Description
a	Add
b	Benchmark
d	Delete
e	Extract
h	Hash
i	Show information about supported formats
l	List
rn	Rename
t	Test
u	Update
x	eXtract with full paths

(Add) command

Adds files to archive.

amples

```
7z a archive1.zip subdir\
```

adds all files and subfolders from folder subdir to archive archive1.zip. The filenames in archive will contain subdir\ prefix.

```
7z a archive2.zip .\subdir\*
```

adds all files and subfolders from folder subdir to archive archive2.zip. The filenames in archive will not contain subdir\ prefix.

```
cd /D c:\dir1\  
7z a c:\archive3.zip dir2\dir3\
```

The filenames in archive c:\archive3.zip will contain dir2\dir3\ prefix, but they will not contain c:\dir1\ prefix.

```
7z a Files.7z *.txt -r
```

adds all *.txt files from current folder and its subfolders to archive Files.7z.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-i \(Include\)](#) [-m \(Method\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-sdel \(Delete files after including to archive\)](#)

[-sfx \(create SFX\)](#)

-si (use StdIn)
-sni (Store NT security information)
-sns (Store NTFS alternate Streams)
-so (use StdOut)
-spf (Use fully qualified file paths)
-ssw (Compress shared files)
-stl (Set archive timestamp from the most recently modified file)
-t (Type of archive)
-u (Update)
-v (Volumes)
-w (Working Dir)
-x (Exclude)

e also

Commands: d (Delete), u (Update)

Switches: -u (Update)

(Benchmark) command

Measures speed of the CPU.

Benchmark execution also can be used to check RAM for errors.

ntax

```
b [number_of_iterations] [-mmt{N}] [-md{N}] [-mm={Method}]
```

The LZMA benchmark is default benchmark for benchmark command.

There are two tests for LZMA benchmark:

1. Compressing with LZMA method
2. Decompressing with LZMA method

The LZMA benchmark shows a rating in MIPS (million instructions per second). The rating value is calculated from the measured speed, and it is normalized with results of Intel Core 2 CPU with multi-threading option switched off. So if you have modern CPU from Intel or AMD, rating values in single-thread mode must be close to real CPU frequency.

You can change the upper dictionary size to increase memory usage by `-md{N}` switch. Also, you can change the number of threads by `-mmt{N}` switch.

The **Dict** column shows dictionary size. For example, 21 means $2^{21} = 2$ MB.

The **Usage** column shows the percentage of time the processor is working. It's normalized for a one-thread load. For example, 180% CPU Usage for 2 threads can mean that average CPU usage is about 90% for each thread.

The **R / U** column shows the rating normalized for 100% of CPU usage. That column shows the performance of one average CPU thread.

Avr shows averages for different dictionary sizes.

Tot shows averages of the compression and decompression ratings.

The test data that is used for compression in that test is produced with special algorithm, that creates data stream that has some properties of real data, like text or execution code. Note that the speed of LZMA for real data can be slightly different.

MA benchmark details

Compression speed strongly depends from memory (RAM) latency, Data Cache size/speed and TLB. Out-of-Order execution feature of CPU is also important for that test.

Decompression speed strongly depends on CPU integer operations. The most important things for that test are: branch misprediction penalty (the length of pipeline) and the latencies of 32-bit instructions ("multiply", "shift", "add" and other). The decompression test has very high number of unpredictable branches. Note that some CPU architectures (for example, 32-bit ARM) support instructions that can be conditionally executed. So such CPUs can work without branches (and without pipeline flushing) in many cases in LZMA decompression code. And such CPUs can have some speed advantages over other architectures that don't support complex conditionally execution. Out-of-Order execution capability is not so important for LZMA Decompression.

The test code doesn't use FPU and SSE. Most of the code is 32-bit integer code. Only some minor part in compression code uses also 64-bit integers. RAM and Cache bandwidth are not so important for these tests. The latencies are much more important.

The CPU's IPC (Instructions per cycle) rate is not very high for these tests. The estimated value of test's IPC is 1 (one instruction per cycle) for modern CPU. The compression test has big number of random accesses to RAM and Data Cache. So big part of execution time the CPU waits the data from Data Cache or from RAM. The decompression test has big number of pipeline flushes after mispredicted branches. Such low IPC means that there are some unloaded CPU resources. But the CPU with Hyper-Threading feature can load these CPU resources using two threads. So Hyper-Threading provides pretty big improvement in these tests.

.MA benchmark in multithreading mode

When you specify $(N*2)$ threads for test, the program creates N copies of LZMA encoder, and each LZMA encoder instance compresses separated block of test data. Each LZMA encoder instance creates 3 unsymmetrical execution threads: two big threads and one small thread. The total CPU load for these 3 threads can vary from 140% to 200%. To provide better CPU load during compression, you can test the mode, where the number of benchmark threads is larger than the number of hardware threads.

Each LZMA encoder instance in multithreading mode divides the task of compression into 3 different tasks, where each task is executed in separated thread. Each of these tasks is simpler than original task, and it uses less memory. So each thread uses the data cache and TLB more effectively in multithreading mode. And LZMA encoder is slightly more effective in multithreading mode in value of "the Speed" divided to "CPU usage".

Note that there is some data traffic between 3 threads of LZMA encoder. So data exchange bandwidth via memory between CPU threads is also can be important, especially in multi-core system with big number of cores or CPUs.

All LZMA decoder threads are symmetrical and independent. So the decompression test uses all hardware threads, if the number of hardware threads is used.

Zip benchmark

With `-mm=*` switch you can run a complex benchmark for 7-Zip code. It tests hash calculation methods, compression and encryption codecs of 7-Zip. Note that the tests of LZMA have big weight in "total" results. And the results are normalized with AMD K8 cpu in that complex benchmark.

The **CPU** rows show CPU frequency. It's measured for sequence of simple CPU instructions. Note: It can be inaccurate, if hyper-threading is used.

The **Effec** column shows Efficiency - the Rating normalized to CPU frequency.

The **E / U** column shows the Efficiency normalized for 100% of CPU usage.

amples

```
7z b
```

runs benchmarking.

```
7z b -mmt1 -md26
```

runs benchmarking with one thread and 64 MB dictionary.

```
7z b 30
```

runs benchmarking for 30 iterations. It can be used to check RAM for errors.

```
7z b -mm=*
```

runs complex 7-Zip benchmark.

```
7z b -mm=* -mmt=*
```

runs complex 7-Zip benchmark for different number of threads : (1, max/2, max), where max is number of available hardware threads. So it can test 3 main modes: single-thread, multi-thread without hyper-threading, multi-thread with hyper-threading.

(Delete) command

Deletes files from archive.

ample

```
7z d archive.zip *.bak -r
```

deletes *.bak files from archive archive.zip.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-i \(Include\)](#) [-m \(Method\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-sns \(Store NTFS alternate Streams\)](#)

[-u \(Update\)](#)

[-w \(Working Dir\)](#)

[-x \(Exclude\)](#)

e also

Commands: [a \(Add\)](#), [u \(Update\)](#)

Switches: [-spf \(Use fully qualified file paths\)](#)

[-stl \(Set archive timestamp from the most recently modified file\)](#)

[-u \(Update\)](#)

(Extract) command

Extracts files from an archive to the current directory or to the output directory. The output directory can be specified by [-o \(Set Output Directory\)](#) switch.

This command copies all extracted files to one directory. If you want extract files with full paths, you must use [x \(Extract with full paths\)](#) command.

7-Zip will prompt the user before overwriting existing files unless the user specifies the [-y \(Assume Yes on all queries\)](#) switch. If the user gives a **no** answer, 7-Zip will prompt for the file to be extracted to a new filename. Then a **no** answer skips that file; or, **yes** prompts for new filename.

7-Zip accepts the following responses:

Answer	Abbr.	Action
Yes	y	
No	n	
Always	a	Assume YES for ALL subsequent queries of the same class
Skip	s	Assume NO for ALL subsequent queries of the same class
Quit	q	Quit the program

Abbreviated responses are allowed.

amples

```
7z e archive.zip
```

extracts all files from archive archive.zip to the current directory.

```
7z e archive.zip -oc:\soft *.cpp -r
```

extracts all *.cpp files from archive archive.zip to c:\soft folder.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-ai \(Include archives\)](#) [-an \(Disable parsing of archive_name\)](#)
[-ao \(Overwrite mode\)](#)
[-ax \(Exclude archives\)](#)
[-i \(Include\)](#)
[-m \(Method\)](#)
[-o \(Set Output Directory\)](#)
[-p \(Set Password\)](#)
[-r \(Recurse\)](#)
[-si \(use StdIn\)](#)
[-sni \(Store NT security information\)](#)
[-sns \(Store NTFS alternate Streams\)](#)
[-so \(use StdOut\)](#)
[-spf \(Use fully qualified file paths\)](#)
[-t \(Type of archive\)](#)
[-x \(Exclude\)](#)
[-y \(Assume Yes on all queries\)](#)

e also

Commands: [x \(Extract with full paths\)](#)

(Hash) command

Calculate hash values for files.

ntax

```
h [-scrc{Method}] [files]
```

Supported methods: CRC32, CRC64, SHA1, SHA256, BLAKE2sp.
Default method is CRC32.

amples

```
7z h a.txt
```

calculates CRC32 for a.txt.

```
7z h -scrcsha256 a.iso
```

calculates SHA256 for a.iso.

```
7z h *
```

calculates CRC32 for all files in current folder and all subfolders.

tes

7-Zip shows hash values for each file, the sum of hash values and the sum that includes all hash values of data and all hash values for filenames.

7-Zip represents hash values for CRC32 and CRC64 as integer numbers in hex.

7-Zip represents hash values For SHA1, SHA256 and BLAKE2sp as sequence of bytes in hex.

itches that can be used with this command

[-i \(Include\)](#) [-m \(Method\)](#)

[-r \(Recurse\)](#)

-src (Set hash method)
-si (use StdIn)
-sns (Store NTFS alternate Streams)
-ssw (Compress shared files)
-x (Exclude)

List contents of archive) command

Lists contents of archive.

amples

```
7z l archive.zip
```

lists all files from archive archive.zip.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-ai \(Include archives\)](#) [-an \(Disable parsing of archive_name\)](#)

[-ax \(Exclude archives\)](#)

[-i \(Include\)](#)

[-slt \(Show technical information\)](#)

[-sns \(Store NTFS alternate Streams\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-t \(Type of archive\)](#)

[-x \(Exclude\)](#)

(Rename) command

Renames files in archive.

ntax

```
rn <archive_name> <src_file_1> <dest_file_1> [ <src_file_2> <dest_file_2> ... ]
```

ample

```
7z rn a.7z old.txt new.txt 2.txt folder\2new.txt
```

renames old.txt to new.txt and 2.txt to folder\2new.txt .

tes

itches that can be used with this command

[-i \(Include\)](#) [-m \(Method\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-stl \(Set archive timestamp from the most recently modified file\)](#)

[-u \(Update\)](#)

[-w \(Working Dir\)](#)

[-x \(Exclude\)](#)

Test integrity of archive) command

Tests archive files.

ample

```
7z t archive.zip *.doc -r
```

tests *.doc files in archive archive.zip.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-ai \(Include archives\)](#) [-an \(Disable parsing of archive_name\)](#)

[-ax \(Exclude archives\)](#)

[-i \(Include\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-sns \(Store NTFS alternate Streams\)](#)

[-x \(Exclude\)](#)

(Update) command

Update older files in the archive and add files that are not already in the archive.

Note: the updating of solid .7z archives can be slow, since it can require some recompression.

ample

```
7z u archive.zip *.doc
```

updates *.doc files to archive archive.zip.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-i \(Include\)](#) [-m \(Method\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-sfx \(create SFX\)](#)

[-si \(use StdIn\)](#)

[-so \(use StdOut\)](#)

[-sni \(Store NT security information\)](#)

[-sns \(Store NTFS alternate Streams\)](#)

[-ssw \(Compress shared files\)](#)

[-spf \(Use fully qualified file paths\)](#)

[-stl \(Set archive timestamp from the most recently modified file\)](#)

[-t \(Type of archive\)](#)

[-u \(Update\)](#)

[-w \(Working Dir\)](#)

[-x \(Exclude\)](#)

e also

Commands: [a \(Add\)](#), [d \(Delete\)](#),

Switches: [-u \(Update\)](#)

(Extract with full paths) command

Extracts files from an archive with their full paths in the current directory, or in an output directory if specified.

See the [e \(Extract\)](#) command description for more details.

amples

```
7z x archive.zip
```

extracts all files from the archive archive.zip to the current directory.

```
7z x archive.zip -oc:\soft *.cpp -r
```

extracts all *.cpp files from the archive archive.zip to c:\soft folder.

tes

7-Zip doesn't use the system wildcard parser. 7-Zip doesn't follow the archaic rule by which *.* means any file. 7-Zip treats *.* as matching the name of any file that has an extension. To process all files, you must use a * wildcard.

itches that can be used with this command

[-ai \(Include archives\)](#) [-an \(Disable parsing of archive_name\)](#)

[-ao \(Overwrite mode\)](#)

[-ax \(Exclude archives\)](#)

[-i \(Include\)](#)

[-m \(Method\)](#)

[-o \(Set Output Directory\)](#)

[-p \(Set Password\)](#)

[-r \(Recurse\)](#)

[-si \(use StdIn\)](#)

[-sni \(Store NT security information\)](#)

[-sns \(Store NTFS alternate Streams\)](#)

[-so \(use StdOut\)](#)

[-spf \(Use fully qualified file paths\)](#)

[-t \(Type of archive\)](#)

-x (Exclude)

-y (Assume Yes on all queries)

e also

Commands: e (Extract)

Command Line Switches

Syntax

```
<switch;> ::= -<switch_characters>[<option>]
```

On the command line, a switch consists of a switch specifier, either a dash (-) or a forward slash (/), followed by the name of the switch. Switch names cannot be abbreviated.

Some switches take an argument after the switch name. No spaces or tabs are allowed within a switch specification. Switch names are not case sensitive, but arguments can be case sensitive.

Switch can be used in any place in command line.

See also [Command Line Syntax](#) for more details about using the command line.

7zip quick reference

Switch	Description
--	Stop switches parsing
-ad	Show dialog box in GUI version (7zg)
-ai	Include archive filenames
-an	Disable parsing of archive_name
-ao	Overwrite mode
-ax	Exclude archive filenames
-bb[0-3]	Set output log level
-bd	Disable progress indicator
-bs{o e p} {0 1 2}	Set output stream for output/error/progress
-bt	Show execution time statistics
-i	Include filenames
-m	Set Compression Method
-o	Set Output directory
-p	Set Password
-r	Recurse subdirectories
-sa	Set Archive name mode
-scc	Set charset for for console input/output
-scrc	Set hash function
-scs	Set charset for list files
-sdel	Delete files after including to archive
-seml	Send archive by email
-sfx	Create SFX archive
-si	Read data from StdIn
-slp	Set Large Pages mode
-slt	Show technical information
-sni	Store NT security information

-sns	Store NTFS alternate Streams
-snt[-]	Replace trail dots and spaces in file names for Extract operation
-snc	Extract file as alternate stream, if there is ':' character in name
-snr	Replace ':' character to '_' character in paths of alternate streams
-snh	Store hard links as links (WIM and TAR formats only)
-sni	Store symbolic links as links (WIM and TAR formats only)
-so	Write data to StdOut
-spd	Disable wildcard matching for file names
-spe	Eliminate duplication of root folder for extract archive command
-spf	Use fully qualified file paths
-ssc	Set Sensitive Case mode
-sse	Stop archive creating, if 7-Zip can't open some input file
-ssw	Compress files open for writing
-stl	Set archive timestamp from the most recently modified file
-stm{HexMask}	Set CPU thread affinity mask (hexadecimal number).
-stx{Type}	Exclude archive type
-t	Type of archive
-u	Update options
-v	Create Volumes
-w	Set Working directory
-x	Exclude filenames
-y	Assume Yes on all queries

(Stop switches parsing) switch

Disables switch parsing after "--" on the command line. This is to allow 7-Zip to use file names that start with "-".

ntax

--

amples

```
7z t -- -ArchiveName.7z
```

```
tests -ArchiveName.7z archive.
```

i (Include archive filenames) switch

Specifies additional include archive filenames and wildcards.

Multiple include switches are supported.

ntax

```
-ai[<recurse_type>]<file_ref>
```

```
<recurse_type> ::= r[- | 0]
```

```
<file_ref> ::= @{listfile} | !{wildcard}
```

Parameters

<recurse_type>

Specifies how wildcards and file names in this switch must be used. If this option is not given, recursion will be not used. For more details see specification of the [-r \(Recurse\)](#) switch.

```
<recurse_type> ::= r[- | 0]
```

<file_ref>

Specifies filenames and wildcards or list file that specify processed files.

```
<file_ref> ::= @{listfile} | !{wildcard}
```

Option	Description
{listfile}	Specifies name of list file. See List file description.
{wildcard}	Specifies wildcard or filename.

Examples

```
7z t -an -air!*.7z
```

tests *.7z archives in current directory and all it's subdirectories.

Commands that can be used with this switch

[e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-ax \(Exclude archives\)](#) [-an \(Disable parsing of archive_name\)](#)

n (Disable parsing of archive_name) switch

Disables parsing of the archive_name field on the command line. This switch must be used with the [-ai \(Include archives\) switch](#). If you use a file list for your archives, you specify it with the -ai switch, so you need to disable parsing of archive_name field from command line.

ntax

-an

amples

```
7z t -an -ai!* .7z -ax!a* .7z
```

tests all *.7z archives, except a*.7z archives.

mmands that can be used with this switch

[e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-ai \(Include archives\)](#) [-ax \(Exclude archives\)](#)

o (Overwrite mode) switch

Specifies the overwrite mode during extraction, to overwrite files already present on disk.

ntax

-ao[a | s | t | u]

Switch	Description
-aoa	Overwrite All existing files without prompt.
-aos	Skip extracting of existing files.
-aou	aUto rename extracting file (for example, name.txt will be renamed to name_1.txt).
-aot	auto rename existing file (for example, name.txt will be renamed to name_1.txt).

amples

```
7z x test.zip -aoa
```

extracts all files from test.zip archive and overwrites existing files without any prompt.

mmands that can be used with this switch

[e \(Extract\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-y \(assume Yes on all queries\)](#),

x (Exclude archive filenames) switch

Specifies archives to be excluded from the operation.

Multiple exclude archive switches are supported.

ntax

```
-ax[<recurse_type>]<file_ref>
```

```
<recurse_type> ::= r[- | 0]
```

```
<file_ref> ::= @{listfile} | !{wildcard}
```

See [-xi \(Include archive filenames\)](#) switch description for information about option parameters.

amples

```
7z t -an -ai!*.7z -ax!a*.7z
```

tests all *.7z archives, except a*.7z archives.

mmands that can be used with this switch

[e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-ai \(Include archives\)](#) [-an \(Disable parsing of archive_name\)](#)

b (Set output log level) switch

The switch sets output log level for Delete/Add/Update/Extract operations.

ntax

-bb[0-3]

Log level	Description
------------------	--------------------

-bb0	disable log (default).
------	------------------------

-bb1	
------	--

or -	show names of processed files in log.
------	---------------------------------------

bb	
----	--

	show names of additional files that were processed internally in
--	--

-bb2	solid archives: skipped files for "Extract" operation, repacked files for "Add" / "Update" operations.
------	--

-bb3	show information about additional operations (Analyze, Replicate) for "Add" / "Update" operations.
------	--

amples

```
7z a a.7z *.txt -bb
```

adds *.txt files to a.7z archive and shows all files that were processed.

```
7z u a.7z *.txt -bb3
```

updates *.txt files in a.7z archive and shows information about additional operations.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

s (Set output stream for output/error/progress ie) switch

ntax

-bs{o|e|p}{0|1|2}

{id}	Stream Type
o	standard output messages
e	error messages
p	progress information

{N}	Stream Destination
0	disable stream
1	redirect to stdout stream
2	redirect to stderr stream

Default values: o1, e2, p1.

amples

```
7z t a.7z -bse1 > messages.txt
```

tests a.7z archive and sends error messages to stdout that is redirected to messages.txt

```
7z a -si -so -bsp2 -txz -an < file.tar > file.tar.xz
```

compresses file.tar (from stdin) to file.tar.xz (stdout stream) and shows progress information in stderr stream that can be seen at console window.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [h \(Hash\)](#), [l \(List\)](#), [e \(Extract\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

(Include filenames) switch

Specifies additional include filenames and wildcards.

Multiple include switches are supported.

ntax

```
-i[<recurse_type>]<file_ref>
```

```
<recurse_type> ::= r[- | 0]
```

```
<file_ref> ::= @{listfile} | !{wildcard}
```

Parameters

<recurse_type>

Specifies how wildcards and file names in this switch must be used. If this option is not given, then the global value, assigned by the [-r \(Recurse\)](#) switch will be used. For more details see specification of the [-r \(Recurse\)](#) switch.

```
<recurse_type> ::= r[- | 0]
```

<file_ref>

Specifies filenames and wildcards, or a list file, for files to be processed.

```
<file_ref> ::= @{listfile} | !{wildcard}
```

Option	Description
{listfile}	Specifies name of list file. See List file description.
{wildcard}	Specifies wildcard or filename.

Examples

```
7z a -tzip src.zip *.txt -ir!DIR1\*.cpp
```

adds to src.zip archive all *.txt files from current directory and all *.cpp files from directory DIR1 and from all it's subdirectories.

Commands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [h \(Hash\)](#), [l \(List\)](#), [rn \(Rename\)](#), [t \(Test\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

See also

Switches: [-r \(Recurse\)](#), [-x \(Exclude\)](#)

1 (Set compression Method) switch

Specifies the compression method.

ntax

-m<method_parameters>

The format for this switch depends on the archive type.

- [Zip](#)
- [GZip](#)
- [BZip2](#)
- [7z](#)
- [XZ](#)
- [WIM](#)

-m switch also can specify hash method for [h \(Hash\)](#) command,

Notes: "Default value" in switches descriptions means the value that will be used if switch is not specified.

It's allowed to use reduced forms for boolean switches: **sw+** or **sw** instead **sw=on**, and **sw-** instead of **sw=off**.

3

Parameter	Default	Description
x=[0 1 3 5 7 9]	5	Sets level of compression.
m={MethodID}	Deflate	Sets a method: Copy, Deflate, Deflate64, BZip2, LZMA, PPMd.
fb={NumFastBytes}	32	Sets number of Fast Bytes for Deflate encoder.
pass={NumPasses}	1	Sets number of Passes for Deflate encoder.
d={Size}[b k m]	900000	Sets Dictionary size for BZip2
mem={Size}[b k m]	24	Sets size of used memory for PPMd.
o={Size}	8	Sets model order for PPMd.
mt=[off on {N}]	on	Sets multithreading mode.
em={EncryptionMethodID}	ZipCrypto	Sets a encryption method: ZipCrypto, AES128, AES192, AES256
tc=[off on]	on	Stores NTFS timestamps for files: Modification time, Creation time, Last access time.
cl=[off on]	off	7-Zip always uses local code page for file names.
cu=[off on]	off	7-Zip uses UTF-8 for file names that contain non-ASCII symbols.
cp={CodePage}	off	Sets code page

By default (if **cl** and **cu** switches are not specified), 7-Zip uses UTF-8 encoding only for file names that contain symbols unsupported by local code page.

[x=\[0 | 1 | 3 | 5 | 7 | 9\]](#)

Sets level of compression. x=0 means Copy mode (no compression).

Deflate / Deflate64 settings:

Level	NumFastBytes	NumPasses	Description
-------	--------------	-----------	-------------

1			Fastest
---	--	--	---------

3	32	1	Fast
5			Normal
7	64	3	Maximum
9	128	10	Ultra

x=1 and x=3 with Deflate method set fast mode for compression.

BZip2 settings:

Level	Dictionary	NumPasses	Description
1	100000		Fastest
3	500000	1	Fast
5			Normal
7	900000	2	Maximum
9		7	Ultra

fb={NumFastBytes}

Sets the number of fast bytes for the Deflate/Deflate64 encoder. It can be in the range from 3 to 258 (257 for Deflate64). Usually, a big number gives a little bit better compression ratio and a slower compression process. A large fast bytes parameter can significantly increase the compression ratio for files which contain long identical sequences of bytes.

pass={NumPasses}

Sets number of passes for Deflate encoder. It can be in the range from 1 to 15 for Deflate and from 1 to 10 for BZip2. Usually, a big number gives a little bit better compression ratio and a slower compression process.

d={Size}[b|k|m]

Sets the Dictionary size for BZip2. You must specify the size in bytes, kilobytes, or megabytes. The maximum value for the Dictionary size is 900000b. If you do not specify any symbol from set [b|k|m], dictionary size will be calculated as $\text{DictionarySize} = 2^{\text{Size}}$ bytes.

mem={Size}[b|k|m]

Sets the size of memory used for PPMd. You must specify the size in

bytes, kilobytes, or megabytes. The maximum value is 256 MB = 2^{28} bytes. The default value is 24 (16MB). If you do not specify any symbol from the set [b|k|m], the memory size will be calculated as (2^{Size}) bytes. PPMd uses the same amount of memory for compression and decompression.

o={Size}

Sets the model order for PPMd. The size must be in the range [2,16]. The default value is 8.

mt=[off | on | {N}]

Sets multithread mode. If you have a multiprocessor or multicore system, you can get a speed increase with this switch. This option affects only compression (with any method) and decompression of BZip2 streams. Each thread in the multithread mode uses 32 MB of RAM for buffering. If you specify {N}, 7-Zip tries to use N threads.

Zip

GZip uses the same parameters as Zip, but GZip compresses only with Deflate method. So GZip supports only the following parameters: x, fb, pass.

7zip2

Parameter	Default	Description
x=[1 3 5 7 9]	5	Sets level of compression.
pass={NumPasses}	1	Sets number of Passes for Bzip2 encoder.
d={Size}[b k m]	900000	Sets Dictionary size for BZip2
mt=[off on {N}]	on	Sets multithreading mode.

x=[1 | 3 | 5 | 7 | 9]

Sets level of compression

Level	Dictionary	NumPasses	Description
-------	------------	-----------	-------------

1	100000		Fastest
3	500000	1	Fast
5			Normal
7	900000	2	Maximum
9		7	Ultra

d={Size}[b|k|m]

Sets the Dictionary size for BZip2. You must specify the size in bytes, kilobytes, or megabytes. The maximum value for the Dictionary size is 900000b. If you do not specify any symbol from set [b|k|m], dictionary size will be calculated as DictionarySize = 2^Size bytes.

pass={NumPasses}

Sets the number of passes. It can be in the range from 1 to 10. The default value is 1 for normal mode, 2 for maximum mode and 7 for ultra mode. A bigger number can give a little bit better compression ratio and a slower compression process.

mt=[off | on | {N}]

Sets multithread mode. If you have a multiprocessor or multicore system, you can get a speed increase with this switch. If you specify {N}, for example mt=4, 7-Zip tries to use 4 threads.

Parameter	Default	Description
x=[0 1 3 5 7 9]	5	Sets level of compression.
yx=[0 1 3 5 7 9]	5	Sets level of file analysis.
s=[off on [e] [{N}f {N}b {N}k {N}m {N}g {N}t]	on	Sets solid mode.
qs=[off on]	off	Sort files by type in solid archives.
f=[off on FilterID]	on	Enables or disables filters. FilterID: Delta:{N}, BCJ, BCJ2, ARM, ARMT, IA64, PPC, SPARC.
hc=[off on]	on	Enables or disables archive header compressing.
he=[off on]	off	Enables or disables archive header encryption.
b{C1}[s{S1}]:{C2}[s{S2}] {N}={MethodID} [:param1][:param2][..]	LZMA2	Sets binding between coders. Sets a method: LZMA, LZMA2, PPMd, BZip2, Deflate, Delta, BCJ, BCJ2, Copy.
mt=[off on {N}]	on	Sets multithreading mode.
mtf=[off on]	on	Set multithreading mode for filters.
tm=[off on]	on	Stores last Modified timestamps for files.
tc=[off on]	off	Stores Creation timestamps for files.
ta=[off on]	off	Stores last Access timestamps for files.
tr=[off on]	on	Stores file attributes.

x=[0 | 1 | 3 | 5 | 7 | 9]

Sets level of compression

Level Method Dictionary FastBytes MatchFinder Filter Descriptio

0 Copy

No
compressio

1	LZMA2	64 KB	32	HC4	BCJ	Fastest compressir
3	LZMA2	1 MB	32	HC4	BCJ	Fast compressir
5	LZMA2	16 MB	32	BT4	BCJ	Normal compressir
7	LZMA2	32 MB	64	BT4	BCJ	Maximum compressir
9	LZMA2	64 MB	64	BT4	BCJ2	Ultra compressir

Note: "x" works as "x=9".

yx=[0 | 1 | 3 | 5 | 7 | 9]

Sets level of file analysis

Level	Description
-------	-------------

0	No analysis.
---	--------------

1 or more	WAV file analysis (for Delta filter).
-----------	---------------------------------------

7 or more	EXE file analysis (for Executable filters).
-----------	---

9 or more	analysis of all files (Delta and executable filters).
-----------	---

Default level is 5: "yx=5".

"yx" works as "yx=9".

If the level of analysis is smaller than 9, 7-Zip analyses only files that have some file name extensions: EXE, DLL, WAV. 7-Zip reads small data block at the beginning of file and tries to parse the header. It supports only some formats: WAV, PE, ELF, Mach-O. Then it can select filter that can increase compression ratio for that file.

By default 7-Zip uses x86 filters (BCJ or BCJ2) for PE files (EXE, DLL). 7-Zip doesn't use analysis in default (yx=5) mode. If (yx=7), then analysis is used for PE files, and it can increase compression ratio for files for non-x86 platforms like ARM.

s=[off | on | [e] [{N}f] [{N}b | {N}k | {N}m | {N}g | {N}t]

Enables or disables solid mode. The default mode is s=on. In solid

mode, files are grouped together. Usually, compressing in solid mode improves the compression ratio.

e Use a separate solid block for each new file extension. You need to use **qs** option also.

{N}f Set the limit for number of files in one solid block

{N}b | {N}k | {N}m | {N}g | {N}t / KiB / MiB / GiB / TiB. Set a limit for the total size of a solid block in bytes

These are the default limits for the solid block size:

Compression Level	Solid block size
Store	0 B
Fastest	16 MB
Fast	128 MB
Normal	2 GB
Maximum	4 GB
Ultra	4 GB

Limitation of the solid block size usually decreases compression ratio but gives the following advantages:

- Decreases losses in case of future archive damage.
- Decreases extraction time of a group of files (or just one file), so long as the group doesn't contain the entire archive.

The updating of solid .7z archives can be slow, since it can require some recompression.

Example:

```
s=100f10m
```

set solid mode with 100 files & 10 MB limits per one solid block.

qs=[off | on]

Enables or disables sorting files by type in solid archives. The default mode is qs=off.

Old versions of 7-Zip (before version 15.06) used file sorting "by type" ("by extension").

New versions of 7-Zip (starting from version 15.06) support two sorting orders:

- qs- : sorting by name : it's default order.
- qs : sorting by type (by file extension).

You can get big difference in compression ratio for different sorting methods, if dictionary size is smaller than total size of files. If there are similar files in different folders, the sorting "by type" can provide better compression ratio in some cases.

Note that sorting "by type" has some drawbacks. For example, NTFS volumes use sorting order "by name", so if an archive uses another sorting, then the speed of some operations for files with unusual order can fall on HDD devices (HDDs have low speed for "seek" operations).

If "qs" mode provides much better compression ratio than default "qs-" mode, you still can increase compression ratio for "qs-" mode by increasing of dictionary size.

If you think that unusual file order is not problem for you, and if better compression ratio with small dictionary is more important for you, use "qs" mode.

Note: There are some files (for example, executable files), that are compressed with additional filter. 7-Zip can't use different compression methods in one solid block, so 7-zip can create several groups of files that don't follow "by name" order in "qs-" mode, but files inside each group are still sorted by name in "qs-" mode.

f=[off | on | FilterID]

Enables or disables compression filters. The default mode is f=on, when 7-zip uses filter only for executable files: dll, exe, ocx, sfx, sys. It uses BCJ2 filter in Ultra mode and BCJ filter in other modes. If f=FilterID if specified, 7-zip uses specified filter for all files. FilterID can be: Delta:{N}, BCJ, BCJ2, ARM, ARMT, IA64, PPC, SPARC.

hc=[off | on]

Enables or disables archive header compressing. The default mode is `hc=on`. If archive header compressing is enabled, the archive header will be compressed with LZMA method.

he=[off | on]

Enables or disables archive header encryption. The default mode is `he=off`.

b{C1}[s{S1}]:{C2}[s{S2}]

Binds output stream S1 in coder C1 with input stream S2 in coder C2. If stream number is not specified, stream with number 0 will be used.

Usually coder has one input stream and one output stream. In 7z some coders can have multiple input and output streams.

For example, [BCJ2](#) encoder has one input stream and four output streams.

mt=[off | on | {N}]

Sets multithread mode. If you have a multiprocessor or multicore system, you can get an increase with this switch. 7-Zip supports multithread mode only for LZMA / LZMA2 compression and BZip2 compression / decompression. If you specify {N}, for example `mt=4`, 7-Zip tries to use 4 threads. LZMA compression uses only 2 threads.

{N}={MethodID}[:param1][:param2] ... [:paramN]

Sets compression method. You can use any number of methods. The default method is LZMA2.

{N} sets the index number of method in methods chain. Numbers must begin from 0. Methods that have smaller numbers will be used before others.

Parameters must be in one of the following forms:

- {ParamName}={ParamValue}.
- {ParamName}{ParamValue}, if {ParamValue} is number and {ParamName} doesn't contain numbers.

Supported methods:

MethodID	Description
LZMA	LZ-based algorithm
LZMA2	LZMA-based algorithm
PPMd	Dmitry Shkarin's PPMdH with small changes
BZip2	BWT algorithm
Deflate	LZ+Huffman
Copy	No compression

Supported filters:

MethodID	Description
Delta	Delta filter
BCJ	converter for x86 executables
BCJ2	converter for x86 executables (version 2)
ARM	converter for ARM (little endian) executables
ARMT	converter for ARM Thumb (little endian) executables
IA64	converter for IA-64 executables
PPC	converter for PowerPC (big endian) executables
SPARC	converter for SPARC executables

Filters increase the compression ratio for some types of files. Filters must be used with one of the compression method (for example, BCJ + LZMA).

MA

LZMA is an algorithm based on Lempel-Ziv algorithm. It provides very fast decompression (about 10-20 times faster than compression). Memory requirements for compression and decompression also are different (see [d={Size}\[b|k|m|g\]](#) switch for details).

Parameter	Default	Description
a=[0 1]	1	Sets compressing mode
d={Size}[b k m g]	24	Sets Dictionary size
mf={MF_ID}	bt4	Sets Match Finder

fb={N}	32	Sets number of Fast Bytes
mc={N}	32	Sets Number of Cycles for Match Finder
lc={N}	3	Sets number of Literal Context bits - [0, 8]
lp={N}	0	Sets number of Literal Pos bits - [0, 4]
pb={N}	2	Set number of Pos Bits - [0, 4]

a=[0|1]

Sets compression mode: 0 = fast, 1 = normal. Default value is 1.

d={Size}[b|k|m|g]

Sets Dictionary size for LZMA. You must specify the size in bytes, kilobytes, or megabytes. The maximum value for dictionary size is 1536 MB, but 32-bit version of 7-Zip allows to specify up to 128 MB dictionary. Default values for LZMA are 24 (16 MB) in normal mode, 25 (32 MB) in maximum mode (-mx=7) and 26 (64 MB) in ultra mode (-mx=9). If you do not specify any symbol from the set [b|k|m|g], the dictionary size will be calculated as DictionarySize = 2^Size bytes. For decompressing a file compressed by LZMA method with dictionary size N, you need about N bytes of memory (RAM) available.

mf={MF_ID}

Sets Match Finder for LZMA. Default method is bt4. Algorithms from hc* group don't provide a good compression ratio, but they often work pretty fast in combination with fast mode (a=0). Memory requirements depend on dictionary size (parameter "d" in table below).

MF_ID	Dictionary	Memory Usage	Description
bt2		9.5 * d	2 bytes hashing
bt3		11.5 * d	3 bytes hashing
	64 KB ... 48 MB	11.5 * d	Binary Tree
bt4	64 MB ... 1024 MB	10.5 * d + 4 MB	
		d	4 bytes

	64 KB ... 48 MB	7.5 * d		hashing
hc4	64 MB ... 1024 MB	6.5 * d	Hash Chain	

Note: Your operation system also needs some amount of physical memory for internal purposes. So keep at least 32MB of physical memory unused.

fb={N}

Sets number of fast bytes for LZMA. It can be in the range from 5 to 273. The default value is 32 for normal mode and 64 for maximum and ultra modes. Usually, a big number gives a little bit better compression ratio and slower compression process.

mc={N}

Sets number of cycles (passes) for match finder. It can be in range from 0 to 1000000000. Default value is $(16 + \text{number_of_fast_bytes} / 2)$ for BT* match finders and $(8 + \text{number_of_fast_bytes} / 4)$ for HC4 match finder. If you specify mc=0, LZMA will use default value. Usually, a big number gives a little bit better compression ratio and slower compression process. For example, mf=HC4 and mc=10000 can provide almost the same compression ratio as mf=BT4.

lc={N}

Sets the number of literal context bits (high bits of previous literal). It can be in range from 0 to 8. Default value is 3. Sometimes lc=4 gives gain for big files.

lp={N}

Sets the number of literal pos bits (low bits of current position for literals). It can be in the range from 0 to 4. The default value is 0. The lp switch is intended for periodical data when the period is equal to 2^{value} (where lp=value). For example, for 32-bit (4 bytes) periodical data you can use lp=2. Often it's better to set lc=0, if you change lp switch.

pb={N}

Sets the number of pos bits (low bits of current position). It can be in

the range from 0 to 4. The default value is 2. The pb switch is intended for periodical data when the period is equal 2^{value} (where $lp = \text{value}$).

MA2

LZMA2 is modified version of LZMA. it provides the following advantages over LZMA:

- Better compression ratio for data than can't be compressed. LZMA2 can store such blocks of data in uncompressed form. Also it decompresses such data faster.
- Better multithreading support. If you compress big file, LZMA2 can split that file to chunks and compress these chunks in multiple threads.

Parameter	Default	Description
c={Size}[b k m g]	dictSize * 4	Sets Chunk size

If you don't specify ChunkSize, LZMA2 sets it to $\max(\text{DictionarySize}, \min(256\text{M}, \max(1\text{M}, \text{DictionarySize} * 4)))$.

LZMA2 also supports all LZMA parameters, but $lp + lc$ cannot be larger than 4.

LZMA2 uses: 1 thread for each chunk in x1 and x3 modes; and 2 threads for each chunk in x5, x7 and x9 modes. If LZMA2 is set to use only such number of threads required for one chunk, it doesn't split stream to chunks. So you can get different compression ratio for different number of threads. You can get the best compression ratio, when you use 1 or 2 threads.

Md

PPMd is a PPM-based algorithm. This algorithm is mostly based on Dmitry Shkarin's PPMdH source code. PPMd provides very good compression ratio for plain text files. There is no difference between compression speed and decompression speed. Memory requirements for compression and decompression also are the same.

Parameter	Default	Description
-----------	---------	-------------

`mem={Size}[b|k|m|g]` 24 Sets size of used memory for PPMd.
`o={Size}` 6 Sets model order for PPMd.

mem={Size}[b|k|m|g]

Sets the size of memory used for PPMd. You must specify the size in bytes, kilobytes, or megabytes. The maximum value is 2GB = 2^{31} bytes. The default value is 24 (16MB). If you do not specify any symbol from the set [b|k|m|g], the memory size will be calculated as (2^{Size}) bytes. PPMd uses the same amount of memory for compression and decompression.

o={Size}

Sets the model order for PPMd. The size must be in the range [2,32]. The default value is 6.

:J2

BCJ2 is a Branch converter for 32-bit x86 executables (version 2). It converts some branch instructions for increasing further compression.

A BCJ2 encoder has one input stream and four output streams:

- s0: main stream. It requires further compression.
- s1: stream for converted CALL values. It requires further compression.
- s2: stream for converted JUMP values. It requires further compression.
- s3: service stream. It is already compressed.

If LZMA is used, the size of the dictionary for streams s1 and s2 can be much smaller (512 KB is enough for most cases) than the dictionary size for stream s0.

Parameters:

d={Size}[b|k|m|g]

Sets section size for BCJ2 filter. Default section size is 64 MB. If you do not specify any symbol from the set [b|k|m|g], the section size will

be calculated as $\text{SectionSize} = 2^{\text{Size}}$ bytes. This parameter doesn't affect memory consumption. Compression ratio is better, if the section size is equal or slightly larger than size of largest execution section in file. Example: `f=BCJ2:d9M`, if largest executable section in files is smaller than 9 MB.

lta

It's possible to set delta offset in bytes. For example, to compress 16-bit stereo WAV files, you can set `"0=Delta:4"`. Default delta offset is 1.

:

XZ supports only LZMA2 codec now. The switches are similar to switches for 7z format.

Parameter	Default	Description
x=[1 3 5 7 9]	5	Sets level of compression
f=FilterID		Sets compression filter. FilterID: Delta:{N}, BCJ, ARM, ARMT, IA64, PPC, SPARC
{N}={MethodID} [:param1][:param2] [.]	LZMA2	Sets compression method: LZMA2: [param1]:[param2]:[...]
mt=[off on {N}]	on	Sets multithreading mode
s=[off on [{N}b {N}k {N}m {N}g]	off	Sets solid mode.

s=[off | on | [{N}b | {N}k | {N}m | {N}g])

Enables or disables solid mode. The default mode is s=off. In solid mode, there is only one block per file or stream.

{N}b | {N}k | {N}m | {N}g Set a limit for the total size of a solid block in bytes

If size of solid block is not specified, default value of solid block size will be calculated, that depends from "compression level" and "dictionary size":

dictionary_size	Default solid block size
smaller than 256 KB	1 MB
256 KB - 64 MB	dictionary_size * 4
64 MB - 256 MB	256 MB
larger than 256 MB	dictionary_size

block size must be equal or large than dictionary size.

If you use multiple blocks:

- the compression ratio with small blocks usually is worse.

- blocks are independent. So losses in case of data damage is limited only to damaged blocks.
- it's possible to extract some particular block of data faster.
- there is index record at the end of xz stream that contains information about position and size of each block.

Note: xz uses: 1 thread for each block in x1 and x3 modes; and 2 threads for each block in x5, x7 and x9 modes. If xz is set to use only such number of threads required for one block, it doesn't split stream to blocks. So you can get different compression ratio for different number of threads. You can get the best compression ratio, when you use 1 thread (for x1 and x3 modes) or 2 threads (for x5, x7 and x9 modes).

Note: each xz block contains LZMA2 stream of data. And LZMA2 also can be divided to independent blocks (chunks). The difference between xz blocks and LZMA2 blocks, that each xz block contains also checksum (crc or sha), and there is index record at the end of xz stream that points to each xz block. 7-Zip by default uses xz blocks. But it's possible to specify the mode when it will use one xz block, and multiple LZMA2 blocks instead.

Examples:

```
s=16m
```

use 16 MB blocks.

```
s
```

use one solid xz block per file.

```
s 0c16m
```

use one solid xz block per file and 16 MiB LZMA2 blocks.

WIM

Parameter	Default	Description
im={ImageNumber}		Sets image number.
is=[off on]	off	Show image number in paths.

If image number is specified, 7-Zip works only with that image inside WIM archive. Other images will be not changed. By default 7-Zip doesn't show image number, if there is only one image in WIM archive, or if image number is specified. But if the switch "is" specified, 7-Zip shows image number.

Examples

```
7z a archive.zip *.jpg -mx0
```

adds *.jpg files to archive.zip archive without compression.

```
7z a archive.7z *.exe *.dll -m0=BCJ -m1=LZMA:d=21
```

adds *.exe and *.dll files to solid archive archive.7z using LZMA method with 2 MB dictionary and BCJ filter.

```
7z a archive.7z a.tar -mf=BCJ2 -mx
```

adds a.tar files to archive archive.7z using BCJ2 filter.

```
7z a archive.7z *.wav -mf=Delta:4
```

adds *.wav files to archive archive.7z using Delta:4 filter.

```
7z a a.7z *.exe *.dll -m0=BCJ2 -m1=LZMA:d25 -m2=LZMA:d19 -m3=LZMA
```

adds *.exe and *.dll files to archive a.7z using BCJ2 filter, LZMA with 32 MB dictionary for main output stream (s0), and LZMA with 512 KB dictionary for s1 and s2 output streams of BCJ2.

```
7z a archive.7z *.txt -m0=PPMd
```

adds *.txt files to archive archive.7z using PPMd method.

```
7z a a.tar.xz a.tar -mf=bcj -mx
```

adds a.tar files to archive a.tar.xz using BCJ filter.

Commands that can be used with this switch

[a \(Add\)](#), [h \(Hash\)](#), [d \(Delete\)](#), [rn \(Rename\)](#), [u \(Update\)](#)

See also

Switches: [-t \(set Type of archive\)](#),

(set Output directory) switch

Specifies a destination directory where files are to be extracted.

This switch can be used only with extraction commands.

ntax

-o{dir_path}

{dir_path}

This is the destination directory path. It's not required to end with a backslash. If you specify * in {dir_path}, 7-Zip substitutes that * character to archive name.

ample

```
7z x archive.zip -oc:\Doc
```

extracts all files from the archive.zip archive to the c:\Doc directory.

```
7z x *.zip -o*
```

extracts all *.zip archives to subfolders with names of these archives.

mmmands that can be used with this switch

[e \(Extract\)](#), [x \(Extract with full paths\)](#)

(set Password) switch

Specifies password.

ntax

-p{password}

{password}

Specifies password.

amples

```
7z a archive.7z -psecret -mhe *.txt
```

compresses *.txt files to archive.7z using password "secret". Also it encrypts archive headers (-mhe switch), so filenames will be encrypted.

```
7z x archive.zip -psecret
```

extracts all files from archive.zip using password "secret".

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [rn \(Rename\)](#), [t \(Test\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

(Recurse subdirectories) switch

Specifies the method of treating wildcards and filenames on the command line.

ntax

-r[- | 0]

Switch	Description
-r	Enable recurse subdirectories.
-r-	Disable recurse subdirectories. This option is default for all commands.
-r0	Enable recurse subdirectories only for wildcard names.

amples

```
7z l archive.zip *.doc -r-
```

lists all *.doc files that belong to the archived root directory in the archive.zip archive.

```
7z a -tzip archive.zip -r src\*.cpp src\*.h
```

adds all *.cpp and *.h files from directory src and all it's subdirectories to the archive.zip archive.

```
7z a archive.7z folder1\
```

adds all files from directory folder1 and all it's subdirectories to the archive.7z archive.

```
7z a archive.7z -r folder2\
```

searches all folder2 directories in all subdirectories, and adds them (including all subdirectories) to the archive.7z archive.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [h \(Hash\)](#), [l \(List\)](#), [rn \(Rename\)](#), [t \(Test\)](#), [u](#)

[\(Update\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-i \(Include\)](#), [-x \(Exclude\)](#)

a (set Archive name mode) switch

Specifies Archive name mode.

ntax

-sa[a | e | s]

Switch	Description
-saa	Always add archive type extension.
-sae	Use exact name specified in command.
-sas	Add extension only if specified name has no extension. It's default option.

amples

```
7z a arc 1.txt
```

creates arc.7z

```
7z a arc.spec 1.txt
```

creates arc.spec

```
7z a arc 1.txt -sae
```

creates arc

```
7z a file.pdf file.pdf -saa -tzip
```

creates file.pdf.zip

mmands that can be used with this switch

[a \(Add\)](#),

cc (Set charset for console input/output) switch

Sets charset for for console input/output.

ntax

```
-scc{UTF-8 | WIN | DOS}
```

Default charset is DOS.

UTF-8

Unicode UTF-8 character set.

WIN

Default character set of Windows.

DOS

DOS (OEM) character set of Windows.

ample

```
7z l archive.7z -sccUTF-8
```

lists files from archive.7z in UTF-8 encoding.

cs (Set charset for list files) switch

Sets charset for list files.

ntax

```
-scs{UTF-8 | UTF-16LE | UTF-16BE | WIN | DOS | {id} }
```

Default charset is UTF-8.

Value	Description
UTF-8	Unicode UTF-8 character set.
UTF-16LE	Unicode UTF-16 little-endian character set.
UTF-16BE	Unicode UTF-16 big-endian character set.
WIN	Default character set of Windows.
DOS	Default DOS (OEM) character set of Windows.
{id}	The code page number (as specified in Microsoft Windows).

Notes: The list file in Unicode charset can start with the BOM (byte order mark) character (U+FEFF). In that case 7-Zip checks that encoding of BOM corresponds to encoding specified with this switch (for UTF-16LE and UTF-16BE).

amples

```
7z a archive.7z @listfile.txt -scsWIN
```

compresses files from listfile.txt list, that contains list of files in default character set of Windows.

```
7z a archive.7z @listfile.txt -scs1251
```

compresses files from listfile.txt list, that contains list of files in 1251 (Cyrillic Windows) codepage.

mmands that can be used with this switch

[a \(Add\)](#), [u \(Update\)](#)

crc (Set hash function) switch

Sets hash function for "extract" and "hash" commands.

ntax

`[-scrc{Method}]`

Supported methods: CRC32, CRC64, SHA1, SHA256, BLAKE2sp, *.
Default method is CRC32.

amples

```
7z t -scrcSHA256 archive.gz
```

tests archive archive.gz and calculated SHA-256 for decompressed data.

```
7z h -scrcsha1 *.iso
```

calculates SHA-1 for *.iso files.

```
7z h -scrc* file.iso
```

calculates hash values for file.iso file.

mmands that can be used with this switch

[h \(Hash\)](#), [e \(Extract\)](#), [x \(Extract with full paths\)](#)

del (Delete files after compression) switch

ntax

-sdel

If -sdel switch is specified, 7-Zip deletes files after including to archive. So it works like moving files to archive.

7-Zip deletes files at the end of operation and only if archive was successfully created.

amples

```
7z a a.7z *.txt -sdel
```

moves txt files from disk's directory to a.7z archive.

mmands that can be used with this switch

[a \(Add\)](#)

eml (Send archive by email) switch

Sends an archive by e-mail.

ntax

-sem1[.]

[.]

Causes the archive to be deleted after attaching a copy of it to the email message.

ample

```
7z a archive.7z -sem1 a.txt
```

compresses the a.txt file and sends it in archive.7z by email.

mmands that can be used with this switch

[a \(Add\)](#), [u \(Update\)](#)

fx (Create SFX archive) switch

Creates self extracting archive.

ntax

-sfx[[{SFX_Module}](#)]

{SFX_Module}

Specifies the SFX module that will be combined with the archive. This module must be placed in the same directory as the 7z.exe. If {SFX_Module} is not assigned, 7-Zip will use standard console SFX module 7zCon.sfx.

SFX_Module	Description
7z.sfx	Windows version.
7zCon.sfx	Console version.
7zS.sfx	Windows version for installers.
7zSD.sfx	Windows version for installers (uses MSVCRT.dll).

SFX module can unpack 7z archive or 7z multivolume archive. For example, if you have name.7z or name.7z.001 archive, just rename sfx module to name.exe and place to same folder with archive.

X modules for installers

SFX modules for installers are included in an external package (7z_extra). You can download these modules from www.7-zip.org. SFX modules for installers (7zS.sfx and 7zSD.sfx) allow you to create your own installation program. Such a module extracts the archive to the user's temp folder, and runs a specified program, and removes the temp files after the program finishes. A self-extracting archive for installers must be created as joining the following files: SFX_Module, Installer_Config (optional), 7z_Archive. You can use the following command to create an installer self-extracting archive:

```
copy /b 7zS.sfx + config.txt + archive.7z archive.exe
```

An optimally small installation package size can be achieved, if the installation files are uncompressed before including them in the 7z archive.

-y switch for installer module specifies quiet mode extraction.

Installer Config file format

This config file contains commands for the Installer. The file begins with the string `;!@Install@!UTF-8!` and ends with `;!@InstallEnd@!`. The file must be written in UTF-8 encoding. The file contains any or all these string pairs:

ID_String="Value"

ID_String	Description
Title	Title for messages
BeginPrompt	Begin Prompt message
Progress	Value can be "yes" or "no". Default value is "yes".
RunProgram	Command for executing. Default value is "setup.exe". Substring <code>%%T</code> will be replaced with path to temporary folder, where files were extracted
Directory	Directory prefix for "RunProgram". Default value is <code>.\</code>
ExecuteFile	Name of file for executing
ExecuteParameters	Parameters for "ExecuteFile"

You may omit any pair.

There are two ways to run a installation program: **RunProgram** and **ExecuteFile**. Use **RunProgram**, if you want to run a program from the .7z archive. Use **ExecuteFile**, if you want to open a document from the .7z archive, or if you want to execute a command from Windows.

If you use **RunProgram**, and if you specify empty directory prefix: **Directory=""**, the system searches for the executable file in the following sequence:

1. The directory from which the application (installer) loaded.
2. The temporary folder, where files were extracted.

3. The Windows system directory.

nfig file Examples

```
;!@Install@!UTF-8!  
Title="7-Zip 4.00"  
BeginPrompt="Do you want to install the 7-Zip 4.00?"  
RunProgram="setup.exe"  
;!@InstallEnd@!
```

```
;!@Install@!UTF-8!  
Title="7-Zip 4.00"  
BeginPrompt="Do you want to install the 7-Zip 4.00?"  
ExecuteFile="7zip.msi"  
;!@InstallEnd@!
```

```
;!@Install@!UTF-8!  
Title="7-Zip 4.01 Update"  
BeginPrompt="Do you want to install the 7-Zip 4.01 Update?"  
ExecuteFile="msiexec.exe"  
ExecuteParameters="/i 7zip.msi REINSTALL=ALL REINSTALLMODE=vomus"  
;!@InstallEnd@!
```

amples

```
7z a -sfx a.exe *.txt
```

adds *.txt files to self extracting archive a.exe using the default console SFX module.

```
7z a -sfx7z.sfx a.exe *
```

adds all files to self extracting archive a.exe with module 7z.sfx using windows version of SFX module.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [u \(Update\)](#),

i (read data from stdin) switch

Causes 7-Zip to read data from stdin (standard input) instead of from disc files.

ntax

`-si{file_name}`

{file_name}

Specifies a name that will be stored in the archive for the compressed data. If file_name is not specified, data will be stored without a name.

Note: The current version of 7-Zip support reading of archives from stdin only for xz, lzma, tar, gzip and bzip2 archives.

amples

```
7z a archive.gz -tgzip -siDoc2.txt < Doc.txt
```

compresses input stream from file Doc.txt to archive.gz archive using Doc2.txt file name.

```
7z x 7z905.tar.gz -so | 7z x -si -ttar
```

decompresses tar.gz archive.

mmands that can be used with this switch

[a \(Add\)](#), [e \(Extract\)](#), [h \(Hash\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

o (write data to stdout) switch

Causes 7-Zip to write output data to stdout (standard output stream).

ntax

-so

If the -so switch is used with the command that creates archive, it works only with some archive formats: xz, gzip, bzip2 and tar.

amples

```
7z x archive.gz -so > Doc.txt
```

decompresses archive.gz archive to output stream and then redirects that stream to Doc.txt file.

```
7z a dummy -tgzip -so Doc.txt > archive.gz
```

compresses the Doc.txt file to the 7-Zip standard output stream and writes that stream to archive.gz file.

mmmands that can be used with this switch

[a \(Add\)](#), [e \(Extract\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

lp (Set Large Pages mode) switch

Sets Large Pages mode.

ntax

-slp[-]

Switch	Description
-slp	Enables Large Pages mode.
-slp-	Disables Large Pages mode. This option is default for all commands.

Large Pages mode increases the speed of compression. However, there is a pause at the start of compression while 7-Zip allocates the large pages in memory. If 7-Zip can't allocate large pages, it allocates usual small pages. Also, the Windows Task Manager doesn't show the real memory usage of the program, if 7-Zip uses large pages. This feature doesn't work on Windows 2000 / 32-bit Windows XP. Also, it requires administrator's rights for your system. The recommended size of RAM for this feature is 3 GB or more. To install this feature, you must run the 7-Zip File Manager with administrator's rights at least once, close it, and then reboot the system.

Notes: if you use -slp mode, your Windows system can hang for several seconds when 7-zip allocates memory blocks. When Windows tries to allocate large pages from RAM for 7-Zip, Windows can hang other tasks for that time. It can look like full system hang, but then it resumes, and if allocation is successful, 7-Zip works faster. Don't use -slp mode, if you don't want other tasks be hanged for several seconds. Also it's senseless to use -slp mode to compress small data sets (less than 100 MB). But if you compress big data sets (300 MB or more) with LZMA method with large dictionary, you can get 5%-10% speed improvement with -slp mode.

ample

```
7z a archive.7z -slp a.iso
```

compresses a.iso file with Large Pages mode switched on.

It (Show technical information) switch

Sets technical mode for [l\(List\)](#) command.

ntax

-slt

ample

```
7z l -slt archive.7z
```

shows detailed technical information for the files in archive.7z.

mmands that can be used with this switch

[l\(List\)](#)

ni (Store NT security information) switch

ntax

-sni

Use this switch to store and restore NT (NTFS) security information for files and directories. Note that only NTFS file system supports that feature.

Current version of 7-Zip can store NT security information only to WIM archives.

amples

```
7z a a.wim -sni *.txt
```

stores txt files with NT security information.

```
7z x a.wim -sni
```

unpacks a.wim and restores NT security information.

mmands that can be used with this switch

[a \(Add\)](#), [e \(Extract\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-sns](#)

ns (Store NTFS alternate Streams) switch

ntax

-sns[-]

Switch	Description
-sns	Enable "Store NTFS alternate streams" mode. It's default option, if you extract archive.
-sns-	Disable "Store NTFS alternate streams" mode. It's default option, if you create archive or call "list" command.

If -sns mode is enabled, 7-Zip processes NTFS Alternate Data Streams for files and folders.

Current version of 7-Zip can store NTFS alternate streams only to WIM archives.

Note: 7-Zip can't include alternate streams to archives on 32-bit Windows XP and older systems.

amples

```
7z a a.wim -sns *.txt
```

stores txt files including alternate data streams.

```
7z x a.wim
```

unpacks a.wim including alternate data streams.

```
7z x a.wim -sns-
```

unpacks a.wim without alternate data streams.

```
7z l a.wim -sns
```

lists files in a.wim including alternate data streams.

mmmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [h \(Hash\)](#), [l \(List\)](#), [t \(Test\)](#), [u \(Update\)](#), [x](#)

[\(Extract with full paths\)](#)

e also

Switches: [-sni](#)

pf (Use fully qualified file paths) switch

Switch	Description
-spf	Use absolute paths including drive letter.
-spf2	Use full paths without drive letter.

Enables the mode that allows to use fully qualified file paths in archives. If -spf switch is not specified, 7-Zip reduces file paths to relative paths when it adds files to archive, and 7-Zip converts paths to relative paths when you extract archive. If -spf switch is specified, 7-Zip doesn't try to process or convert paths.

Fully qualified file paths begin with one of the following:

- A UNC name, which starts with two backslash characters, for example, "\\Server1\".
- A disk designator with a backslash, for example "C:\".
- A single backslash, for example, "\Folder".

If -spf switch is specified, but the path is not fully qualified, 7-Zip will use specified path, it will not convert the path to fully qualified path.

Please be careful, if you use -spf switch with "extract" command. Check that file names in archive are correct. Note that with -spf switch 7-Zip can try to rewrite any file with path specified in archive.

ntax

-spf

amples

```
7z a a.7z -spf c:\Files\test.txt d:\test.txt
```

stores both txt files with full paths.

```
7z x a.7z -spf
```

extracts files from a.7z archive with exact file paths specified in archive.

Commands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

e also

Switches:

sc (Set Sensitive Case mode) switch

Sets sensitive case mode for file names.

ntax

-scs[-]

Switch	Description
-ssc	Set case-sensitive mode. It's default for Posix/Linux systems.
-ssc-	Set case-insensitive mode. It's default for Windows systems.

ample

```
7z a archive.7z A*.txt -ssc -r
```

compresses all A*.txt files from current directory and all it's subdirectories. That command doesn't compress a*.txt files.

mmmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

sw (Compress files open for writing) switch

Compresses files open for writing by another applications. If this switch is not set, 7-zip doesn't include such files to archive.

ntax

-SSW

ample

```
7z a archive.7z -ssw *.txt
```

compresses all *.txt files in current folder including files open for writing by another applications.

mmands that can be used with this switch

[a \(Add\)](#), [h \(Hash\)](#), [u \(Update\)](#)

tl (Set archive timestamp from the most recently modified file) switch

ntax

-stl

If -stl switch is specified, 7-Zip sets timestamp for archive file as timestamp from the most recently modified file in that archive.

amples

```
7z u a.7z -stl *.txt
```

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [rn \(Rename\)](#), [u \(Update\)](#)

tx (Exclude archive type) switch

The switch disables the code related to specified archive type and an archive can not be open with specified archive type.

Multiple -stx switches are supported.

ntax

-stx{archive_type}

{archive_type}

Specifies the type of archive. It can be: 7z, xz, split, zip, gzip, bzip2, tar,

ample

```
7z x -stxpe archive.exe
```

extracts files from archive.exe archive and doesn't use PE format handler.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

(set Type of archive) switch

Specifies the type of archive.

ntax

`-t{archive_type}[:s{Size}][:r][:e][:a]`

{archive_type}

Specifies the type of archive. It can be: *, #, 7z, xz, split, zip, gzip, bzip2, tar,

***:r**

Default mode. 7-Zip opens archive and subfile, if it's supported by format.

Opens only one top level archive.

***:s{Size}[b | k | m | g]**

Sets upper limit for start of archive position. Default scan size is 8 MBytes `"*:s8m"`. Example: `"*:s0"` means that it will open only file that has no any stub before archive.

#

Opens file in Parser mode, and ignores full archives.

#:a

Same as `*`, but it opens files with unknown extensions that contain archives in Parser Mode.

#:e

Opens file in Parser mode and checks all byte positions as start of archive.

If `-t{archive_type}` switch is not specified, 7-Zip uses extension of archive filename to detect the type of archive. If you create new archive, -

`t{archive_type}` switch is not specified and there is no extension of archive, 7-Zip will create .7z archive.

If `-t{archive_type}` switch is not specified and archive name contains incorrect extension, the program will show the warning.

It's possible to use the combined type (for example, `mbr.vhd`) for "Extract" and "List" commands for some archives.

When you extract archive of some types that contains another archive without compression (for example, MBR in VHD), 7-Zip can open both levels in one step. If you want to open/extract just top level archive, use `-t*` switch.

Note: `xz`, `gzip` and `bzip2` formats support only one file per archive. If you want to compress more than one file to these formats, create a tar archive at first, and then compress it with your selected format.

ample

```
7z a -tzip archive.zip *.txt
```

adds all *.txt files from current directory to zip archive archive.zip.

```
7z t -t7z.split archive.7z.001
```

tests all files in archive.7z.001. It also checks that archive is multivolume .7z archive.

```
7z x -t# sfxarchive.exe
```

extracts sfxarchive.exe in parser mode.

```
7z x -tiso archive.iso
```

extracts files from archive.iso open as ISO archive.

```
7z x -tudf archive.iso
```

extracts files from archive.iso open as UDF archive.

mmmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [u \(Update\)](#), [x \(Extract with](#)

[full paths\)](#)

(Update options) switch

Specifies how to update files in an archive and (or) how to create new archives.

ntax

-u[-]<action_set>[!{new_archive_name}]

<action_set> ::= <state_action>...

<state_action> ::= <state><action>

<state> ::= p | q | r | x | y | z | w

<action> ::= 0 | 1 | 2 | 3

rameters

dash (-)

Disables any updates in the base archive.

The term **base archive** means the archive assigned by "base_archive_name" on the command line. See [Command line syntax](#) for more details.

{new_archive_name}

Specifies the path name of the new archive to be created. All options in this switch will refer to this new archive.

If not assigned, then all options in this switch will refer to the base archive of the command.

<state>

Specifies the state of a particular file to be processed.

<state> ::= p | q | r | x | y | z | w

For each unique filename there are 6 variants of state:

<state>	State condition	File on Disk	File in Archive
p	File exists in archive, but is not matched with wildcard.		Exists, but is not matched
q	File exists in archive, but doesn't exist on disk.	Doesn't exist	Exists
r	File doesn't exist in archive, but exists on disk.	Exists	Doesn't exist
x	File in archive is newer than the file on disk.	Older	Newer
y	File in archive is older than the file on disk.	Newer	Older
z	File in archive is same as the file on disk	Same	Same
w	Can not be detected what file is newer (times are the same, sizes are different)	?	?

<action>

Specifies the action for a given [<state>](#).

`<action> ::= 0 | 1 | 2 | 3`

For each state you can specify one of the three variants of actions:

<action>	Description
0	Ignore file (don't create item in new archive for this file)
1	Copy file (copy from old archive to new)
2	Compress (compress file from disk to new archive)
3	Create Anti-item (item that will delete file or directory during extracting). This feature is supported only in 7z format.

marks

Any update command (such as [a \(Add\)](#), [d \(Delete\)](#), [u \(Update\)](#)) can be assigned in these terms.

The following table shows action sets for update commands.

command \ <u><state></u>	p	q	r	x	y	z	w
<u>d (Delete)</u>	1	0	0	0	0	0	0
<u>a (Add)</u>	1	1	2	2	2	2	2
<u>u (Update)</u>	1	1	2	1	2	1	2
Freshen	1	1	0	1	2	1	2
Synchronize	1	0	2	1	2	1	2

If you don't specify a !{new_archive_name} option, then all options will refer to the main archive (the archive assigned on the command line after the 7z command). If you specify !{new_archive_name} option, then 7-Zip also will create a new archive with the specified name and all options will refer to that new archive.

Multiple update switches are supported. 7-Zip can create any number of new archives during one operation.

By default, the action set for each new archive is assigned as the action set of the main command. There are 3 different action sets for commands: a (Add), d (Delete), u (Update). You can overload any <state_action> pair.

ne zone notes

If you change time zone (when you move your computer to another time zone or if there are clock changes for daylight saving in your zone), you can have some problems with update commands that depend from file's modification time. It's strongly recommended to use only file system that uses Coordinated Universal Time (UTC) and archive format that also uses UTC. In that case you will have no problems with time zone changes. Also it's recommended to use only UTC formats in other cases, for example, if you send files to someone in another time zone.

Also in some cases there are no problems, if both file system and archive format use local time, for example, FAT file system and ZIP format.

- UTC file systems: NTFS
- UTC archive formats: .zip with -mtc switch, 7z, tar, gzip2, iso, wim

- Local time file systems : FAT, FAT32
- Local time archive formats : rar, zip, cab

amples

```
7z u c:\1\exist.7z -u- -up0q3x2z0!c:\1\update.7z *
```

creates a new archive update.7z and writes to this archive all files from current directory which differ from files in exist.7z archive. exist.7z archive will not be changed.

```
7z u c:\1\exist.7z -up0q3x2z0!c:\1\update.7z * -ms=off
```

creates a new archive update.7z and writes to this archive all files from the current directory which differ from files in exist.7z archive.

Note: the updating of solid .7z archives can be slow, since it can require some recompression.

mmmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [rn \(Rename\)](#), [u \(Update\)](#),

(Create Volumes) switch

Specifies volume sizes.

ntax

`-v{Size}[b | k | m | g]`

{Size}[b | k | m | g]

Specifies volume size in Bytes, Kilobytes (1 Kilobyte = 1024 bytes), Megabytes (1 Megabyte = 1024 Kilobytes) or Gigabytes (1 Gigabyte = 1024 Megabytes). if you specify only {Size}, 7-zip will treat it as bytes.

It's possible to specify several -v switches.

NOTE: Please don't use volumes (and don't copy volumes) before finishing archiving. 7-Zip can change any volume (including first volume) at the end of archiving operation.

amples

```
7z a a.7z *.txt -v10k -v15k -v2m
```

creates multivolume a.7z archive. First volume will be 10 KB, second will be 15 KB, and all others will be 2 MB.

mmands that can be used with this switch

[a \(Add\)](#),

/ (set Working directory) switch

Sets the working directory for the temporary base archive. By default, 7-Zip builds a new base archive file in the same directory as the old base archive file. By specifying this switch, you can set the working directory where the temporary base archive file will be built. After the temporary base archive file is built, it is copied over the original archive; then, the temporary file is deleted.

ntax

-w[[{dir_path}](#)]

{dir_path}

Specifies the destination directory path. It's not required that a path end with a backslash.

If <dir_path> is not assigned, then 7-Zip will use the Windows temporary directory.

ample

```
7z a -tzip archive.zip *.cpp -wc:\temp
```

adds *.cpp files to the archive.zip archive, creating a temporary archive in c:\temp folder.

mmmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [rn \(Rename\)](#), [u \(Update\)](#),

(Exclude filenames) switch

Specifies which filenames or wildcarded names must be excluded from the operation.

Multiple exclude switches are supported.

ntax

```
-x[<recurse_type>]<file_ref>
```

```
<recurse_type> ::= r[- | 0]
```

```
<file_ref> ::= @{listfile} | !{wildcard}
```

See [-i \(Include\)](#) switch description for information about option parameters.

amples

```
7z a -tzip archive.zip *.txt -x!temp.*
```

adds to the archive.zip all *.txt files, except temp.* files.

```
7z a archive.7z Folder1\ -xr!*.*png
```

adds to the archive.7z all files from Folder1 and its subfolders, except *.png files.

mmands that can be used with this switch

[a \(Add\)](#), [d \(Delete\)](#), [h \(Hash\)](#), [e \(Extract\)](#), [l \(List\)](#), [t \(Test\)](#), [rn \(Rename\)](#), [u \(Update\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-r \(Recurse\)](#), [-i \(Include\)](#)

(assume Yes on all queries) switch

Disables most of the normal user queries during 7-Zip execution. You can use this switch to suppress overwrite queries in the [e \(Extract\)](#) and [x \(Extract with full paths\)](#) commands.

ntax

-y

amples

```
7z x src.zip -y
```

extracts all files from src.zip archive. All overwrite queries will be suppressed and files on disk with same filenames as in archive will be overwritten.

mmands that can be used with this switch

[e \(Extract\)](#), [x \(Extract with full paths\)](#)

e also

Switches: [-ao \(Overwrite mode\)](#),

Menu Items

Note: 'Grey' refers to the numeric keypad.

e

Menu item	Shortcut	Description
System		Submenu with menu commands from system shell. It can be enabled in Options.
Open	Enter	Open current item
Open Inside	Ctrl+PgDn	Open current item as folder inside 7-Zip
Open Inside *		Open only one top level archive container
Open Inside #		Open file in parser mode
Open Outside	Shift+Enter	Open current item in new window
View	F3	Open selected item with viewer
Edit	F4	Open selected item with editor
Rename	F2	Rename selected item
Copy To...	F5	Copy selected items
Move To...	F6	Move selected items
Delete	Delete	Delete selected items
Split file...		Split file to parts
Combine files...		Combine files to one file
Properties	Alt+Enter	Show Properties of file
Comment	Ctrl+Z	Set comment for file
Calculate checksum		Calculates CRC checksum for files
Diff		Show the differences between two files. It can be enabled, if the diff program is set in Options
Create	F7	Create new folder

Folder		
Create File	Shift+F4	Creates new file
Link...		Create symbolic link or hard link
Alternate Streams		Show alternate file streams at NTFS.
Exit	Alt+F4	Closes the program.

lit

Menu Item	Shortcut	Description
Select All	Shift+[Grey +]	Select all items
Deselect All	Shift+[Grey -]	Select all items
Invert Selection	[Grey *]	Select / Deselect all items
Select...	[Grey +]	Select specified items
Deselect...	[Grey -]	Deselect specified items
Select by Type	Alt+[Grey +]	Select all items with the same extension as current item
Deselect by Type	Alt+[Grey -]	Deselect all items with the same extension as current item

əw

Menu Item	Shortcut	Description
Large Icons	Ctrl+1	Displays items by using large icons
Small Icons	Ctrl+2	Displays items by using small icons
List	Ctrl+3	Displays items in a list
Details	Ctrl+4	Displays items in a list with detailed information about each item
Name	Ctrl+F3	Sort items by Name

Type	Ctrl+F4	Sort items by Type
Date	Ctrl+F5	Sort items by Date
Size	Ctrl+F6	Sort items by Size
Unsorted	Ctrl+F7	Do not sort items
Flat View		Switch Flat view mode for list of files
2 Panels	F9	Switch On/Off second panel
Toolbars		Menu items for toolbars handling
Open		
Root	\	Open root computer folder
Folder		
Up One Level	Backspace	Open the folder one level up
Folders History...	Alt+F12	Open folders history
Refresh	Ctrl+R	Refresh items list
Auto Refresh		If "Auto Refresh" is enabled, 7-Zip reloads list of files, if there are changes in files on drive.

avorites

Menu Item	Shortcut	Description
Add folder to Favorites as		Adds folder to favorites

ols

Menu Item	Shortcut	Description
Options...		Open Options dialog box
Benchmark		Measure the performance of your computer with 7-Zip LZMA benchmark

lp

Menu Item	Shortcut	Description
Contents...	F1	Opens 7-Zip Help

About 7-Zip...

Opens About dialog box

scellaneous commands

Shortcut	Description
Tab	Switch between panels
Insert	Select / Deselect current item
Shift+F10	Display the shortcut menu for the selected items
RightCtrl+0 Alt+0 ...	Open folder bookmark
RightCtrl+9 Alt+9 Shift+RightCtrl+0 Shift+Alt+0 ...	Creates folder bookmark
Shift+RightCtrl+9 Shift+Alt+9	
Alt+F1	Edit the Folder Address on left panel
Alt+F2	Edit the Folder Address on right panel
Alt+Up	Open same folder in other panel
Alt+Left, Alt+Right	Open current folder in other panel
Ctrl+[Grey +]	Adjust optimal column width for items
Ctrl+C	Copy the name of selected file to clipboard

Keyboard Shortcuts

Key	Normal	Ctrl	Alt	Shift
Tab	Switch panel			
F1	Help		Left Path	
F2	Rename		Right Path	
F3	View	Sort by Name		
F4	Edit	Sort by Type	Exit	Create File
F5	Copy	Sort by Date		
F6	Move	Sort by Size		
F7	Create Folder	Unsorted		
F9	1/2 Panels			
F10	Menu			Shortcut Menu
F12			Folders History	
0-9		R: Folder Bookmark	Folder Bookmark	
A		Select All		
C		Copy to clipboard		
N		Create File		
R		Refresh		
Z		Comment		
Backspace	Up One Level			
Enter	Open		Properties	Open outside
Insert	Select Item			
Delete	Delete Item			
Page Down		Open Inside		

Page Up

Up One Level

Up

Open same
folder

Left

Open current
folder

Right

Open current
folder

**** Open root

Grey / Open root

Grey * Select All

Grey + Select

Adjust columns

Select by Type

Select All

Grey - Deselect

Deselect by
Type

Deselect All