

**Arc64**

**COLLABORATORS**

	<i>TITLE :</i> Arc64		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		February 12, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Arc64</b>	<b>1</b>
1.1	Arc64 documentation (18th Sept.96) . . . . .	1
1.2	Disclaimer . . . . .	1
1.3	About Arc64 . . . . .	2
1.4	Requirements . . . . .	2
1.5	How to use Arc64 . . . . .	2
1.6	What is still left to do? . . . . .	3
1.7	Thanx to... . . . .	3
1.8	Bugs . . . . .	4
1.9	History . . . . .	4
1.10	How to contact the author . . . . .	4
1.11	P00 . . . . .	5
1.12	T64 . . . . .	5
1.13	LYNX . . . . .	6
1.14	D64 . . . . .	6
1.15	ZipCode . . . . .	7

---

# Chapter 1

## Arc64

### 1.1 Arc64 documentation (18th Sept.96)

Arc64 V1.1  
© 1996 by John 'Graham' Selck

Disclaimer

About

Requirements

How to use

Future

Thanx to...

Known Bugs

History

Contact

### 1.2 Disclaimer

Arc64 is written and copyright © 1996 by  
John Selck

.

This program is freely distributable unless no changes  
are made to the archive.

The author is not liable for any damaga/problems/loss of data  
this program might directly or indirectly cause.

---

No parts of this program may be altered or resourced in any way.

### 1.3 About Arc64

Arc64 is a program designed to give you fast access to files stored in c64 archives and 1541 disk images.

It was mainly written because 'UNd64' (© by Jess Sosnoski) didn't support ZipCode archives.

However, Arc64 is meant as an alternative towards UNd64.

In comparision Arc64 has the following advantages:

- ZipCode support
- 6600 bytes length
- less textoutput
- c64-like directory output
- supports T64 V1.1

### 1.4 Requirements

Arc64 has been tested on the following configurations:

A500/512KB Chip/Kick1.3  
A500/512KB Chip/512KB Fast/Kick2.04  
A1200/2MB Chip/12MB Fast/Kick3.0 (68030/50MHz)  
A1200/2MB Chip/16MB Fast/Kick3.1 (68060/50MHz)

Someone can test it with Kick1.1???

### 1.5 How to use Arc64

Arc64 is used via CLI.

It supports

P00  
,  
T64  
,  
Lynx  
,  
D64  
and  
ZipCode  
.

Arc64 {command} {archive name}

Commands are: l = List directory/archived files

---

x = Extract files from archive  
c = Convert ZipCode to D64 / D64 to ZipCode  
a = Add directory to logfile

Example:

```
Arc64 x HD2:D64/vendetta.d64
```

...will extract all files from the d64 image into the actual directory.

```
Arc64 c DH1:1!Demo_archive
```

...will convert the ZipCode archive "Demo\_archive" into a plain d64 archive. (This also works the other way round...)

I recommend that you use Arc64 from DirOpus or something similar.

If you wonder about the 'a' command, i have to say that i included this one for myself.

```
Arc64 a CD0:Arc/xxx.d64
```

will basically do the same as

```
Arc64 l CD0:Arc/xxx.d64 >Arc64_Log
```

with the difference that the name of the actual archive will still be written onto the screen.

(The logfile will be created on the actual path as 'Arc64\_Log'.)

Errors which may appear while listing:

Illegal Link

- Same as Illegal Track/Sector, only for the directory structure.

Looped Directory

- This happens quite often since people always used to play around with the directory.

## 1.6 What is still left to do?

- Adding files to existing archives.

## 1.7 Thanx to...

Thanx to...

Geert Verschueren (Sorex/WOW)

---

... for that strange Lynx file with that 0 blocks USR files

Paul Gardner-Stephen (Highlander/FLT)

... for his encouragement giving me information on ZipCode  
(although i found out myself...)

Jess Sosnoski

... for UNd64

## 1.8 Bugs

Known Bugs/Problems

No bugs known yet...

For bugreports, press  
this one  
...

## 1.9 History

History

1.0 - Initial Release

1.1 - T64 support (1.0 & 1.1)  
P00 support  
BF: saveroutine only saved 'ok'-files. fixed.  
The Lynx routines now notice a broken Lynx archive,  
same goes to T64.

## 1.10 How to contact the author

Contact me at:

John Selck (Graham/Oxyron)  
Suederholz 13  
24885 Sieverstedt  
Germany

E-Mail: [j.selck@flensburg.netsurf.de](mailto:j.selck@flensburg.netsurf.de)

Or visit our homepage:

<http://www.amigasupport.com/oxyron/>

---

## 1.11 P00

P00 archives are file archives containing only one file. They were introduced by PC64, a C64 emulator for MS-DOS.

\*\*\* Format of P00 list output:

Archive: battlflld.p00

67 "BATTLEFIELD" PRG

\*\*\* Format of T64 extract output:

Extracting from: battlflld.p00

battlefield ok

\*\*\* Possible errors:

ok No error.

Skipped... Another file with the same filename is already on disk.

## 1.12 T64

T64 archives are file archives containing one or more files. The T64 format was introduced by C64S, a C64 emulator for MS-DOS.

There are currently two versions of the T64 format. The first (1.0) supports only single files and the second (1.1) supports multiple files. Both are supported by Arc64.

\*\*\* Format of T64 list output:

Archive: gt021.t64

```
"GAME TAPE #021          " <- Tape Header
1  "LOADER                " PRG <- Complete file
157 "MAIN.DAT             "*PRG <- Incomplete file
33  "E000                 "*DEL <- The archive doesn't contain the file
```

\*\*\* Format of T64 extract output:

Extracting from: gt021.t64

```
loader_____ ok
main.dat_____ Size error!
e000_____ Skipped...
```

\*\*\* Possible errors:

ok No error.

---



Size error! Either the archive is too small  
or the file length not indicated correctly.  
However, there are a lot of T64 files which  
contain only 1 file and have the header from  
the original C64S demo tape image.

Skipped... The file is not within the archives range or  
another file with the same filename is already  
on disk.

## 1.13 LYNX

Lynx archives are file archives containing one or more files.

\*\*\* Format of Lynx list output:

Archive: iffconverterv1.lnx

```
0  "-----" USR
0  "-   1995   -" USR
0  "-----" USR
13 "IFFCONV V1.0/MSP" PRG
17 "IFFCONV DOCS/MSP" PRG
108 "TESTPICTURE.LBM" *SEQ <- Incomplete file
```

\*\*\* Format of Lynx extract output:

Extracting from: iffconverterv1.lnx

```
----- Skipped...
-   1995   - Skipped...
----- Skipped...
iffconv_v1.0/msp ok
iffconv docs/msp ok
testpicture.lbm Size error!
```

\*\*\* Possible errors:

ok No error.

Size error! Either the archive is too small  
or the file length not indicated correctly.  
However, there are a lot of T64 files which  
contain only 1 file and have the header from  
the original C64S demo tape image.

Skipped... The file is not within the archives range or  
the file has a size of 0 blocks or another  
file with the same filename is already on disk.

## 1.14 D64

---

D64 archives are complete 1541 images.  
The D64 format was introduced by C64S, a C64 emulator for MS-DOS.

\*\*\* Format of D64 list output:

Archive: Utils1.d64

```
0 "KICKIN' ASS IN " 96 2A <- Disk header & ID
12 "TURBOCOPY 1.0" PRG <- File
166 "URIDIUM++ /DCS" PRG< <- Write protected file
108 "TOOLBOX V1.0/UCC"*PRG <- Unclosed file
48 "DEFENDER" PRG
4004 BLOCKS FREE
```

\*\*\* Format of D64 extract output:

Extracting from: Utils1.d64

```
turbocopy_1.0 ok
uridium+___\dcs ok
toolbox_v1.0\ucc Illegal Track/Sector!
defender Looped filestructure!
```

\*\*\* Possible errors:

ok No error.

Illegal Track/Sector! A blockpointer in the file points to a  
block out of the range of a 1541 disk.

Looped filestructure! A blockpointer points to a block earlier in  
the file.

Corrupted EOF! The last block of a file contains an invalid  
bytesize.

Skipped... A file with the same name is already on disk.

## 1.15 ZipCode

ZipCode archives are complete 1541 images.

\*\*\* Format of ZipCode list output:

Archive: 1!Utils1

```
0 "KICKIN' ASS IN " 96 2A <- Disk header & ID
12 "TURBOCOPY 1.0" PRG <- File
166 "URIDIUM++ /DCS" PRG< <- Write protected file
108 "TOOLBOX V1.0/UCC"*PRG <- Unclosed file
48 "DEFENDER" PRG
4004 BLOCKS FREE
```

\*\*\* Format of ZipCode extract output:

---

Extracting from: 1!Utils1

turbocopy\_1.0 ok  
uridium+\_\_\_\dcs ok  
toolbox\_v1.0\ucc Illegal Track/Sector!  
defender Looped filestructure!

\*\*\* Possible errors:

ok No error.

Illegal Track/Sector! A blockpointer in the file points to a  
block out of the range of a 1541 disk.

Looped filestructure! A blockpointer points to a block earlier in  
the file.

Corrupted EOF! The last block of a file contains an invalid  
bytesize.

Skipped... A file with the same name is already on disk.

---