

**UNd64**

**COLLABORATORS**

	<i>TITLE :</i> UNd64		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 30, 2023	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>UNd64</b>	<b>1</b>
1.1	UNd64 Guide, yea yea yea...	1
1.2	UNd64 Guide, yea yea yea...	1
1.3	_FIRST_ there was UNd64...	2
1.4	.d64 Section	2
1.5	.x64 Section	3
1.6	.t64 Section	3
1.7	.p00 Section	3
1.8	DISCLAIMER:	3
1.9	DESCRIPTION:	4
1.10	REQUIREMENTS:	4
1.11	WHAT ARE .d64 DISK IMAGES?	4
1.12	WHAT ARE .X64 DISK IMAGES?	5
1.13	WHAT ARE .t64 TAPE IMAGES?	5
1.14	.t64 Output	5
1.15	.t64 Notes...	6
1.16	.X64 Notes...	6
1.17	WHAT ARE .p00 PROGRAM IMAGES?	6
1.18	.p00 Output	7
1.19	Usage	7
1.20	Program Output:	8
1.21	Progress Messages	9
1.22	Compatibility	10
1.23	Notes...	11
1.24	Version Benchmarks:	11
1.25	History:	12
1.26	TO DO...	14
1.27	About the author...	14
1.28	And...The Quote of the Day:	15
1.29	The Amiga 600 From Hell	15
1.30	Thanks to...	16

# Chapter 1

## UNd64

### 1.1 UNd64 Guide, yea yea yea...

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

This program has become emailware, if you would like to be considered a registered user, you have to send me email! If you can't send me email, postcards or other forms of communication are perfectly acceptable.  
(smoke signals, etc...)

Also, it will help me figure out what kind of user base this program has.

To The Docs....

### 1.2 UNd64 Guide, yea yea yea...

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

Disclaimer

Description

Requirements

Usage

Progress Messages

.d64 Support

.x64 Support

.t64 Support

.p00 Support

Compatibility

History

TODO

About the Author...

Quote of the day

`_FIRST_` there was UNd64..

Thanks to...

### 1.3 `_FIRST_` there was UNd64...

And it still IS UNd64!!!

There's another program out there which DARES to imply that it sends my program into obsolescence with this line, and considering my program (IMHO) is FAR more robust and idiot-proof (not to mention speed), I'd avoid this other program.

COMPARE AND SAVE !!!

BENCHMARKS:

-----

Extracting from COMMODOR.D64 with output redirected to >NIL:

UNd64 6 seconds

Brand-X 35 seconds

I'll give the other program this: it very nicely "simulates" a 1541 drive, especially speedwise, and the way it reacts to circular links (I pity anyone with a SCSI-II and '060 equipped A4000T with a big hard drive. That other program should successfully fill that son of a b\* up real damn quick.)

Oh, yea...my program's smaller too.

### 1.4 .d64 Section

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

What's a .d64 diskimage?

Output

---

Notes

Benchmarks

## 1.5 .x64 Section

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

What's a .x64 diskimage

Notes

## 1.6 .t64 Section

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

What's a .t64 tapeimage?

Output

Notes

## 1.7 .p00 Section

UNd64 v 38.6 (17.2.96)

© Copyright 1994-1996 Jess Sosnoski

What's a .p00 program image?

Output

## 1.8 DISCLAIMER:

DISCLAIMER:

-----

Use this program at your own risk!

I assume no responsibility or liability for problem(s) and/or damage(s) that occur by the use, modification, and/or existence of this product, and/or its parts in any form.

(I.E. If it makes you spill beer on your computer, ugh..the thought of it)

---

This program is copyrighted by Jess Sosnoski and is available as e-mailware. It may be distributed only under the following conditions:

- 1.) Program and Documentation is not modified in any way.
- 2.) Program and Documentation must be distributed together.
- 3.) This package may not be used for commercial purposes.  
Without my permission.
- 4.) Sale price must not exceed the cost of media and shipping plus a nominal copying fee equaling no greater than fred fish charges.
- 5.) I give permission to release this on CD-ROM.  
Especially Aminet and Fred Fish.

/\* I like the idea of being immortalized in a piece of laser-etched aluminum encased in nice non bio-degradable plastic. \*/

## 1.9 DESCRIPTION:

### DESCRIPTION

-----  
This program is designed to extract, list or test files from .d64 disk images, .t64 tapeimages, and .p00 program images from PeeCee C64 emulator disk/tape/program image files.

Good luck finding such archives around the 'net anymore, since Escom is making good ol' 64's again.

## 1.10 REQUIREMENTS:

### REQUIREMENTS

-----  
An Amiga with Kickstart 1.1+ (I don't have 1.0 so I can't test it on that) about 180K free ram, .d64's, .t64's, and/or .p00 files and a storage device that will hold them--and possibly the output files.

## 1.11 WHAT ARE .d64 DISK IMAGES?

### WHAT ARE .d64 DISK IMAGES?

-----  
.d64 files are the sector-by-sector contents of a Commodore 1541 formatted disk.

These files are commonly used by PeeCee C-64 emulators to store C-64 files, and to xfer 1541 disks into a convenient method of storage for use by the emulators. (I believe the emulator is called C64S)

Unfortunately, none of our Amiga C-64 emulators can take advantage of this. (HINT! HINT! Questronix/A-64!!!)

This program reads in the archive file and allows one to extract, list or test the files within the archive.

There also exists an archive on Aminet that allows one to create .d64 archives on a 5.25" drive equipped Amiga. The archive also contains a program for de-arc'ing files from .d64 archives.

(I believe it's in misc/emu/1541.lha)

---

## 1.12 WHAT ARE .X64 DISK IMAGES?

WHAT ARE .X64 DISK IMAGES?

-----  
 .X64 files are the diskimage format used by X64, a Unix C-64 emulator.  
 They cover 1541, 1570, 1571, and 1581 formats.  
 This program currently only supports 1541/1570 formats.

## 1.13 WHAT ARE .t64 TAPE IMAGES?

WHAT ARE .t64 TAPE IMAGES?

-----  
 .t64 files are a format developed by Miha Petemel for the PeeCee C-64  
 emulator C64s. Just another format that the emulator supports/uses to  
 conveniently store C-64 files.

## 1.14 .t64 Output

UNd64 38.6 (17.2.96)

© Copyright 1994-96 Jess Sosnoski

Tape Name: imported files  
 Number of Entries: 30  
 Number Used : 1  
 Version : 1.0

Size	Type	Filename
37632	PRG	1a.C64

Output Description

-----  
 Tape Name: Name that the creator of the archive named the archive...

Number of Entries: So far, all of the .t64's I've ran into had 30  
 entries, as did the specs for the .t64 format file  
 I have. If there are more than 30 entries, the  
 program will get mad.

Number Used: The number of files stored in the .t64 image. There is  
 one instance where I found it to be wrong, and it is in  
 older .t64 files. (see  
     .t64 notes  
   )

Version: Version of the .t64 tapeimage. This program supports version  
 1.0 only, I don't know of any other versions, however, if you  
 have or know of any higher versions, or the specs for higher  
 versions, let  
     me



know!

Bytes: Number of bytes in the file.

Type: This will be ---, PRG, or IMG. --- stands for an empty space in the directory, whether these files can be extracted or not, I'm not sure, I never ran into this type. PRG are normal C-64 files. IMG are image files (according to T64.DOC), the program skips over these, but will list them. I'm not sure if the file format conforms to the same standards as the PRG type, however if I run into an IMG file and it is the same format, I'll only have to change one line to extract them.

Filename: Hmmm....still haven't figured that out yet...

NOTE: Filenames are converted to AmigaASCII, and any characters <32 or >127 are converted to printable characters.  
Characters /"?:#\*, are converted to -

## 1.15 .t64 Notes...

.t64 Notes

-----

There are 2 types of .t64 files this program recognizes.

One has the header C64 tape image file, and C64S tape file, the former is I guess an older version of the tapeimage.

These older images may have some problems with them, one of which is that the filesize in some of them can be wrong, and the other is that some of them may claim that there aren't any entries, but still contain one. Another problem is that they use a filetype #68 for PRG files, and I don't know if there are any more besides that, normally 0 is empty, 1 is PRG, and 2+ are IMAge files.

This program will STILL check for files even if the archive says there aren't any in it.

If anybody has any further information on any other filetypes in the old archiving method.

## 1.16 .X64 Notes...

.X64 Notes

-----

Basically a 1541/1570 .x64 diskimage is the same as a standard .d64 diskimage with a 64-byte header in front describing what type of image it is.

For all intents and purposes, UNd64 just reads this header to make sure it's reading the right format, and then treats it as a .d64 file.

Therefore.....for more info, check the .d64 section...

## 1.17 WHAT ARE .p00 PROGRAM IMAGES?

## WHAT ARE .p00 PROGRAM IMAGES?

-----  
.p00 files are a format used on the PeeCee C-64 emulator C-64s, they are basically a C-64 program with a header on front with an identifier and the C-64 filename. I guess this format was created to make storing C-64 files on a PeeCee a bit easier, since the C-64 has 16 character-long filenames.

## 1.18 .p00 Output

UNd64 38.6 (17.2.96)  
© Copyright 1994-96 Jess Sosnoski

```
Size  Filename
-----
33247      logo
```

Not Much to Explain here...

## 1.19 Usage

### USAGE

-----  
From the Shell/CLI (only):  
und64 [command]{option} [filename] {path}

If run from workbench, it yells at you.  
If run without parameters, usage information will be displayed.  
At any point during program execution, pressing CTRL-C will abort.

### Commands:

l List file(s) in archive.  
x Extract file(s) from archive.  
t Test file(s) in archive. (If used on a .p00 file, it will act like list.)

### Options:

d Handle DELETED files (.d64 ONLY!)  
\* WARNING \* Resurrecting DELETED files is risky business, if a new file is partially written over part of a DEL'd file, you may also get the end of another file as a surprise bonus. Or maybe you'll just get a part of an already complete file.

a Handle ALL files. (.d64 ONLY!)  
This will allow extracting of splat files, or in other words, incorrectly closed files. Same warning as DEL files.  
Splat files have a filetype prefix of \*.

n No Filename Conversion. This leaves the filenames alone, however, they might look pretty weird in the Amiga's ASCII.  
Illegal characters are still replaced during extraction.

---

**Filename:**

Name of the file including the .d64/.t64/.p00 extension (if exists).  
The filename will now accept up to 255 characters.

**Path:**

The optional output path, if you don't want to extract to the same directory you have your archives in.  
(Just thought it MIGHT be useful, perhaps)  
If the output path includes a directory, you must put a slash after it.  
i.e. RAM:junk/  
otherwise (as with the above example, but with no trailing slash), it would extract to RAM:  
\* NOTE \* the path may not exceed 255 characters or you may get unpredictable results.  
(I'd say 255 characters is a pretty fair limit, given the fact that nobody out there should want to bury c64 files that deep in directories)

**Example:**

```
und64 la FooBar.d64
```

\*NOTE\* The program autodetects the filetype of the archive, and acts accordingly.

## 1.20 Program Output:

**Program Output:**

```
UNd64 38.6 (17.2.96)
© Copyright 1994-96 Jess Sosnoski
```

```
Disk Name:old games          00 2a
  Approx
Blks  Bytes  Type  Filename
-----
   2    508  *PRG  truck and field
  71  18034  *DEL<  track
  97  24638  *PRG<  field
  33   8382   PRG   spacedinvaders
  33   8382   PRG   knight driver
  33   8382   PRG   race
  33   8382   PRG   wizzard of war
  33   8382   PRG   kickinthehead
   1    254   PRG   amigaforever!!
  33   8382   PRG   bewareilive
  66  16764   PRG   beamryder
   3    762  *DEL   attack
  55  13970  *DEL   .attack
   4   1016  *DEL   kl
  33   8382   PRG   ratrace
 117  29718   PRG  jumping jacks j2
   0     0   *DEL   froggee ii
```

**Output Description**

-----  
Blocks: number of disk blocks it took up on the disk.

---

A disk block=256 bytes, 254 used for data.

Approx Bytes: 254\*Blocks. This is to give an approximate filesize  
Actual filesize should be smaller than or equal to this.

NOTE: If someone messed with the dir structure and manually changed the number of blocks, this could give false information. If blocks>664, which is physically impossible, the length is set to 0.

Type: This will be DEL, PRG, SEQ, or USR. This is for informational purposes only, and does not affect outputted files. For those of you out there who are not Commodore users, PRG files are typically programs/executables. SEQ files are typically text files. USR can be about anything. DEL files are deleted, you can list or attempt to extract them when using the d option. (DEL files are always marked as splat files) There are also REL(ative) files, which are not supported. However, the program will list them. (and ONLY list them). REL files are 3-d type files with all kinds of side sectors and records and junk--so I chose not to deal with them. (as if they could be properly stored on a filesystem which does not support them!) Splat files are prefixed with a \*, and locked files are postfixed with a <. Splat files are improperly closed files or deleted files, and on a Commodore disk are read protected. Locked files on a Commodore disk are write/delete protected. These file attributes will not affect the output files either.

Filename: Gee, I dunno.

NOTE: Filenames are converted to AmigaASCII, and any characters <32 or >127 are converted to printable characters. Illegal characters /"#:\*, are converted to -

## 1.21 Progress Messages

Progress Messages

-----

\*\*\*Break

You pressed CTRL-C and stopped execution.

\* Extracting

File is being extracted.

\* Testing

File is being tested for integrity

>>Extracting/Testing Errors:

\* Not a d64/t64/p00 archive!!

a.) The filelength wasn't EXACTLY 174848 bytes, which all legal .d64's are.

b.) The file's header didn't contain any of the following identifier strings: C64S tape file, C64 tape image file, C64File.

(first 2 are .t64 identifiers, the third is a p00 identifier).

However, if you have a file of exactly 174848 bytes, or put one of the mentioned id strings at the front of the file, you can fool the program

(results should be interesting).

\* Circular Link Found, File Truncated

UNd64 found a circular link in a .d64 archive and cut the output file off. A circular link occurs when a pointer to the next track & sector of a file points to a track & sector earlier in a file, which can cause an infinite loop (not good).

\* Circular Link Found in Directory

Same as above, except it's in the directory structure. UNd64 will stop going thru the directory if it finds one of these nasties.

\* File size error (n bytes)

In a .d64 file this is either caused by circular links, improper file size in the directory block, or a corrupt file (possible disk error). In a .t64 file this is caused by an improper file length stored in the directory header. I.E. if the filelength + file offset > .t64 filesize. The number in parenthesis is the number of bytes the file actually was.

\* Illegal Track or Sector

Relax, there's nothing wrong with your Amiga's drive, this just means that a link to a track and sector in a file on the .d64 diskimage is corrupt (probably xferred from the original disk). And most likely (or absolutely likely) a 15x1 drive would complain the same way if you tried to access the file in question from the original disk.

\* Illegal Track or Sector in Directory

Same as above, except that it was in the directory structure instead of a file structure.

\* File Exists! (R)ename (O)verwrite (S)kip (A)bort

If a file already exists with the same name, this message appears. At this point you would select the first letter of the appropriate option and press return. Rename will ask you for a new name to give the file, Overwrite will overwrite the file, Skip will skip extracting the file, and Abort will stop the program A-la CTRL-C.

.t64 file is a higher version than currently supported!  
(fairly self explanatory...)

```
*** Cant Read Input File!  
*** Cant Write Output File!  
*** Couldnt Allocate Memory!  
*** Error Reading Input File!  
*** Error Writing Output File!
```

These 5 errors are all fatal, and the program will exit immediately. They're pretty self-explanatory when they happen. (There's almost always a guarantee that the Amiga will throw a requester of some kind at you before one of the above I/O errors)

## 1.22 Compatibility

COMPATIBILITY

-----

---

This program was developed on:  
                   The Amiga 600 From Hell  
                   Tested on:  
 A600 2Chip      1.3/1.3  
 A600 2Chip 4Fast  2.05/2.1  
 A600 2Chip 4Fast  3.1/3.1  
 A1000 512K Chip  1.1,1.2,1.3  
 A3000 2Chip 8Fast 2 16BitFast 2.04/2.1, 3.1/3.1  
 (25 mhz '030 with LOTS of SCSI devices on it)

...And it worked on all of them, but bug reports are still welcome.  
 (I know it needs at least ~180K free memory to run)

## 1.23 Notes...

### NOTES

-----  
 You might have noticed a couple of .d64's that only have one or two files on it, and that those files don't seem to do anything. Well, if this is the case, it is likely that the program(s) you extracted are the boot code for a trackloader-type program. This means that the information for the program is stored on the disk in another way besides a 1541-dos-compliant style. Basically, this means, you can't do much with them. If you need to use them that badly, put the .d64 image back onto a 1541 disk, then use the 1541 thru A64's interface. Oh, yea, if anybody knows of such a program for the 64, and where to get it, let  
                   me  
                   know.

## 1.24 Version Benchmarks:

### BENCHMARKS

-----  
       Versions/Time in Seconds  
 Command  37.5-  38.0+  Factor  
 -----  
 la (list)  8      1  ~800%  
 ta (test)  27      3  ~900%  
 xa (extract) 68     18  ~377%

Tested on archive commodor.d64 from WATSON.MBB.SFU.CA.  
 und64 la commodor.d64 >NIL:  
 in RAM: on:

                  The Amiga 600 From Hell  
                   Results may vary depending on archive and machine.

## 1.25 History:

### HISTORY

-----

- 37.1 First Release. Written and compiled in PCQ Pascal 1.2d (the PD release), assembled on A68k 2.71, disassembled with IRA 1.02, re-assembled and optimized with PhxAss 3.30, linked throughout with PhxLnk 2.03, and hunkmerged with Imploder 4.0.  
Basically, it does what is says it does.  
It has a styleguide-compliant version string.  
(or seems to be)  
Size: 9340 (19.4.94)
- 37.2 Added circular-link checking and fixed a ctrl-c bug when attempting to break while extracting (37.1 didn't stop). Unfortunately c-l checking slows down extraction a wee bit. Fixed REL file bug, it no longer lists or attempts to extract these nasty little files. Improved illegal filename character checking. Rewrote some parts of this doc.  
Size: 10080 (21.4.94)
- 37.3 Speeded up circular-link check to take constant time  $O(1)$ , instead of summatory time, improved output. Added duplicate filename checking. Improved illegal filename character checking. Rewrote some more of the doc.  
Size: 10684 (27.4.94)
- 37.4 Now primarily assembled with SNMA 1.39 instead of A68K, and re-assembled & optimized with Phxass 3.71, producing shorter faster code. Added ability to process deleted and splat files, and put some output filesize checking in.  
Size: 9944 (4.5.94)
- 37.5 Assembled with SNMA 1.95, and skipped Phxass because it's optimization was causing GURU's-and I didn't want to mess with the program to get it to work at 3am. Added file testing, support for locked, etc.. files, removed a few bugs, rewrote some parts of code. Improved output.  
Size: 11888 (10.6.94)
- 37.6 Assembled and optimized with SNMA 1.97. Rewrote reading/writing routines for a MAJOR speed increase. Rewrote parts of doc.  
Size: 12168 (3.1.95)
- 37.7 Added archive filesize checking in for almost perfect .d64 archive recognition.  
Size: 13000 (28.1.95)
- 38.0 TOTALLY rewrote reading/writing routines, MUCH faster. Rewrote and cleaned up even more code, which fixed quite a few nasty bugs.  
Added AmigaGuide documentation, with a text editor!  
Fixed bug when file was improper length, now reports correct length. Fixed filename bug, now filenames aren't
-

padded with spaces. Output redirection works again.  
Fixed benchmarks in the doc, more accurate results made  
by sending program's output to >NIL:.  
Improved output. Now also lists (but ONLY lists) REL files.  
Now locked and splat files are marked as such.  
(i.e. all file attributes list just like an old 1541)  
Beefed up error checking for all kinds of stuff.  
Added .t64 and .p00 support, both types are auto-detected.  
Primarily assembled & optimized with SNMA 1.97, linked  
with PhxLnk 4.03, disassembled with IRA 1.02, reassembled  
and optimized even more with PhxAss 4.10.  
Now compiled with the 3.1 PCQ Includes.  
Size: 16028 (14.3.95)

- 38.1 Fixed a bug in the command parsing which required one  
to select another option in order to use the no filename  
conversion option.  
Assembled with snma 1.99, reassembled with PhxAss 4.18, and  
linked with PhxLnk 4.17.  
Size: 15960 (5.7.95)
- 38.2 Rewrote fileexists function, parts of the filesize function,  
file extraction code, directory track parsing code,  
and track & sector link parsing code.  
File sector parsing now reports illegal tracks/sectors.  
Dir track parsing code now checks for illegal track/sectors  
as well as circular links.  
Added output path. (something useful in a dearcng utility)  
Added more types of error messages.  
Assembled with snma 2.03, reassembled with PhxAss 4.23.  
Size: 19692 (24.10.95)
- 38.3 Fixed a bug in the filesize function which was not freeing  
up a fileinfoblock in certain situations. Also fixed a  
possible bug in the fileexists function  
(if it wasn't a bug, it won't matter anyway due to the fix)  
Cleaned up some ancient code and vars, and did a wee bit  
of optimization in some routines. (smaller code)  
Size: 19600 (28.10.95)
- 38.4 Removed a bit of redundant code and cleaned up the  
filesize function (again! :< ), seems that in the  
unlikely situation if it were able to lock the file,  
allocate a fileinfoblock, and then fail to examine it,  
it wouldn't free up the fileinfoblock...FIXED (finally).  
Optimized bits'n'pieces of this and that, etc...  
Assembled with snma 2.04 and phxass 4.25 now.  
(smaller code than 38.2 or 38.3 !)  
Size: 19376 (26.11.95)
- 38.5 Added dates to the history.  
Now assembled with phxass 4.26 and linked with  
phxlnk 4.20.  
Fixed (I hope!) mungwall hits and crashes when a long  
pathname is used or long filename is used.  
Now accepts up to 255 characters for filenames.  
Size: 19372 (13.1.96)
-



38.6 Fixed the date of the last version in the docs.  
(damn, I hate when I forget what year it is)  
Added X64 .x64 diskimage support for 1541/1570 diskimages.  
Now extracts .t64 IMAge files.  
(if there's any problems, contact me)  
Size: 20164 (17.2.96)

## 1.26 TO DO...

TODO

----

- SAS/C (6.56 on my friend's A3000)
- Wildcard extraction & support
- Shut off the cursor during output (anyone know the ansi sequences for this?)
- !-Deeper .d64 bitmap checking when resurrecting deleted/splat files to  
! warn you if the file was overwritten by a legal file.  
! (and maybe chop the file off there too...)
- !-Rewrite more of the .d64 code so that the program doesn't eat up 180K  
every time you run it, even if you're not processing a .d64 file.
- Other c-64 emulators file support. (I just need specs & files to test 'em)

HARDER things TO DO

-----

- Make it possible to create/modify .d64 diskimage files. Copying files  
both from and to, renaming, deleting, etc...  
(I'm not sure if it's actually worth it, I've never had the need to do so)
- Make the above into a virtual disk .d64filesystem/device so that any  
program can make diskimages that may be xferred to C-64's & emulators.
- Repair circular links (so the whole file can be extracted).
- Make it possible to create/modify .t64 tapeimage files.  
This will be hellishly easier than doing so for .d64's.  
(Although...same reason as above for modifying .d64 files)  
...some of these would be neat, but more current c64 emulators are  
making the idea of a standalone utility a bit obsolete.  
(but I may still do it just for the sheer hell of it)

## 1.27 About the author...

The Author's Addresses...

-----

Jess Sosnoski  
651 Hillside Drive  
Mount Carmel, PA 17851-2463  
USA

If you're nuts enough to call..  
Phone: (717) 339-4796

My \*NEW\* Internet address:  
starblaz@postoffice.ptd.net

My "web page"...if you want to call it that (under HEAVY construction)  
<http://home.ptd.net/~starblaz/>

Please also send mail regarding Kick3lonA600, TinyPALNTSC, AddFreq, or NT70 here too!

If you feel the irresistible need to send me money, gifts or an Amiga 4000 Tower, I will give you a very gracious thank you. (I'm also accepting U.S. Addresses for anyone carrying that nice new Apollo A620 '020 accellerator!-the '600 from hell MUST get one!)

```

      .
     /|
    /_|
   _/ _|MIGA...It Lives!

```

## 1.28 And...The Quote of the Day:

Quote of the Day:

"Who's the king baby!"  
 -Al Bundy

Runner Up:

"It was tricky, but I think we've pulled it off."  
 -Jeff Lynne

## 1.29 The Amiga 600 From Hell

The Amiga 600 From Hell:

Kickstart/Workbench 3.1 (Kickstart 40.63 to be exact, and works PERFECTLY!)  
 4 MB PCMCIA Fast ram  
 2 MB Chip  
 120MB Conner 2 1/2" internal IDE HD  
 DataFlyer SCSI+ 1200 Scsi interface  
 DataFlyer XDS External IDE interface With a Rodime 3.5" 100M IDE HD  
 Sony CDU-590 Double Speed SCSI CD-ROM  
 Hayes Accura 288 V.FC + FAX  
 Hewlett Packard DeskJet 540c  
 Amiga 2002 Monitor (A wee bit nicer than the old 1080, but not much)  
 Amiga 1010 low-density external floppy.

Yes Kick3.1 is in ROM, and I paid \$75 for it. (my wallet's still cringing)  
 (YES, the IDE HD and PCMCIA work)  
 In case you're wondering, this rom is advertised as the A500/2000/2500 rom, however, if you plan to 3.1 your A600, make sure the rom is version 40.63!)

---

### 1.30 Thanks to...

Thanks to:

-----

Kay Drangmeister

For bugreports, and sourcecode for readargs, and testing it on an '060.

Don Becker

For ideas for better bitmap checking, to be used in a future version of und64.

Jouko Valta

Author of X64, for sending me some .x64 files to test with UNd64.

Amiga Technologies

For resurrecting the Amiga!

Everyone else who sent me ideas, email, and/or bugreports.

And...Everyone who spread the program.

If you have ever sent me mail, for bugreports/ideas,  
and your name is not mentioned here, and you would like it mentioned here,  
email

me

.

---