

# **DWDecode v1.02 Documentation**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> DWDecode v1.02 Documentation		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 5, 2022	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1 DWDecode v1.02 Documentation</b>	<b>1</b>
1.1 DWDecode v1.02 Documentation	1
1.2 Table Of Contents	2
1.3 Requirements	3
1.4 Basics	3
1.5 What It Handles	4
1.6 Limitations	5
1.7 How To Get Started	5
1.8 Filenoting of Input Files	6
1.9 Run time Options & Internal Defaults	7
1.10 Decoders and their Failure Codes	8
1.11 Files used for Determining Decoding Information	9
1.12 Variables used for Determining Decoding Information	9
1.13 Files used for Setting Runtime Options	9
1.14 Directories and Input and Output	10
1.15 Filtering of Input Files	11
1.16 Report Control Options	12
1.17 Delete Options	12
1.18 Debugging Options	13
1.19 Logging Options	14
1.20 Miscellaneous Runtime Options	14
1.21 Making DWDecode Smarter	15
1.22 Summary List	15
1.23 Index of Options	17
1.24 History	19
1.25 Bugs	20
1.26 Thanks	20

---

## Chapter 1

# DWDecode v1.02 Documentation

### 1.1 DWDecode v1.02 Documentation

DWDecode v1.02 Documentation

Copyright ©1996-1997 by Dick Whiting

Standard Disclaimer: I wrote it, it works for me, I don't guarantee that it will do anything productive for anyone else, etc. etc. ;-)

HOWEVER, if you do find a use for it: I homeschool my kids and they would love a postcard from where EVER you live. Instant Geography Lesson;)

Postcards:     Dick Whiting  
                  28590 S. Beaver creek Rd.  
                  Mulino, Oregon 97042  
                  USA

Email:             Dick Whiting <dwhiting@europa.com>

Table Of Contents

Requirements

Basics

How To Get Started

Filenoting of Input Files

Run time Options & Internal Defaults

History

Bugs

---

Thanks

## 1.2 Table Of Contents

MAIN	
DWDecode v1.02 Documentation	
1.	
Requirements	
2.	
Basics	
2.1.	
What It Handles	
2.2.	
Limitations	
3.	
How To Get Started	
4.	
Filenoting of Input Files	
5.	
Run time Options & Internal Defaults	
5.01.	
Decoders and their Failure Codes	
5.02.	
Files used for Determining Decoding Information	
5.03.	
Variables used for Determining Decoding Information	
5.04.	
Files used for Setting Runtime Options	
5.05.	
Directories and Input and Output	
5.06.	
Filtering of Input Files	
5.07.	
Report Control Options	
5.08.	
Delete Options	
5.09.	
Debugging Options	
5.10.	
Logging Options	
5.11.	
Miscellaneous Runtime Options	
5.12.	
Making DWDecode Smarter	
5.13.	
Summary List	
5.14.	
Index of Options	
6.	
History	
7.	
Bugs	
8.	
Thanks	

---

## 1.3 Requirements

You will need several programs/libraries from Aminet:

RexxArpLib by Willy Langeveld - util/rexx/ral3p3.lha

FlashFind by Frank Würkner - util/cli/FlashFind1.2.lha

A UUdecoder

e.g. UUout by Nicolas Dade - util/arc/uuOut114.lha

A Base64 decoder

e.g. Base64coders by Edmund Vermeulen - comm/mail/Base64coders.lha

A BinHex decoder

e.g. binhex374 by Nik Soggia - util/arc/binhex374.lha

And, of course, some Use Net files with encoded binaries;)

FILES INCLUDED IN THIS ARCHIVE:

DWdecode	- the REAL program.
DWdecode.Base64.Types	- means of identifying Base64 first lines
DWdecode.Binary.Types	- means of identifying filenames in subject
DWdecode.Doc	- text format of documentation
DWdecode.Encode.Types	- MIME encoding types
DWdecode.Guide	- Amiga guide documentation
DWdecode.Guide.info	- info file for guide
DWdecode.Prefs	- a starter preference file
DWdecode.Readme	- copy of the Aminet readme file
DWdecode.Section.Types	- describes Section lines of UUencoded files
DWdecode.Skip.Froms	- allows for filtering based on From: lines
DWdecode.Skip.Subjects	- allows for filtering based on Subject: lines
Samples/....	- subdirectory with example script(s)
arexxqsort.lha	- handles sorting of stem variables.
by Willy Langeveld	- originally found on Aminet

## 1.4 Basics

This program does MASS decoding of Use Net files. It decodes ALL binaries that it is able to recognize, which is most;) from the source directory and optionally from ALL subdirectories as well.

What It Handles

Limitations

## 1.5 What It Handles

It uses information from MIME content lines, section lines, begin lines and (if it must) the subject line to determine what kind of encoding is used and which parts go together. It reports on files that have missing parts and on ones that it is unable to obtain complete information for.

It handles either single directories or can handle COMPLETE directory trees.

It handles files with SINGLE & MULTIPLE UUencoded OR Base64 binaries within them and Multipart UUencoded or Base64 files.

It handles files with APPLE/DOUBLE format. My logic uses the LAST file of a given name within a SINGLE input file. This forces the APPLE portion to be considered a duplicate and not processed.

It allows you to filenote the decoded binaries with a variety of substitution values as well as literals.

It is quite 'smart' and generally can figure out most files.

It is highly configurable and using external files can 'learn'.

It should work with almost ANY Base64 and UU decoder.  
See WARNING below about pathname removal.

It allows for filtering based on From: and/or Subject: lines.

It allows for deleting input files after decoding, deleting duplicates, deleting input files based on From: &/or Subject: lines, deleting files that have ONLY text in them, and deleting files based on their age.

Reporting can be tailored by setting/unsetting a variety of switches on the command line or in the prefs file.

Probably other things ... but this IS enough.

WARNING!!:

I strip ALL pathing off of encoded filenames prior to invoking the Decoders. I thought about making this an option, but decided not to. It would be TOO easy for someone to overlay one of your system files.

For Base64 (using Base64Decode) the stripped filename is created in DEST directory.

UUout uses WHATEVER it finds on the begin line and I can't override it. For UUdecoding, assuming you use UUout, specify USEBASENAME as an option. This will prevent an encoded file of S:Startup-Sequence (for example) overlaying YOURS! I do not suggest using a decoder that does not have this option!

---

## 1.6 Limitations

- 1) If someone REALLY tries to mislead this program; he CAN. I do alot of matching logic to locate parts xx/yy and associate them with the correct part 1/yy, but nothing is certain.

EXAMPLE OF HOW TO CONFUSE DWDECODE:

- 1) Multipart file with part 1 subject totally different from others
- 2) Multipart file with wrong xx/yy information, or swapped xx/yy
- 3) BASE64 file that has bad numbers in the subject. SOME people use 1/2 to mean 'I sent TWO files and here is the FIRST one'. This WILL show on the error report and you can hand edit the subject.

Just as a minor point, these would be ALMOST impossible to figure out using your brain instead of a program. You would have to ASSUME too many things to even have a chance of being able to assemble the parts, or you could just try all of the combinations until one worked;)

- 2) Very few error conditions are returned from UUOUT or BASE64, so I am able to catch very few. Generally I have found that even if I CAN detect an error, there is really no way to fix it -- generally the input file is corrupt. The only REAL exception is that UUOUT demands an 'end ' statement and you can go in and add one or fix the existing one (note that 'end ' should have a blank after the 'd').
- 3) It DOES NOT handle:
  - a) Files that include both UUencoded AND Base64 encoded pieces within the SAME file. (I have NEVER seen one, but you never know ...)
  - b) MultiPart files that include portions of more than 1 encoded binary. (I know that wasn't clear, but ...) For example: You encoded 3 pictures and spread them across 6 files with no 1 file containing a complete piece. Again, I haven't seen this, but ... oops!, figures;), just received one!!
  - c) It will probably have problems with joining files with dozens of parts because of the length of the command that is constructed. This has never been a problem for me since I NEVER have gotten ALL the necessary parts when I needed more than 4;) Actually, I SELDOM end up with all 3 of a 3/3 file.
  - d) It does not handle multipart BinHex files. I can not tell that parts xx/yy are in BinHex format rather than Base64. If YOU know how to tell by 'looking' that a file is in BinHex rather than Base64 please email me. Thanks.

## 1.7 How To Get Started

Unload the archive to somewhere (make sure the 'S' protection bit ←  
is set for  
the program - DWdecode to make it easier to invoke)

---



Get any required programs from Aminet and install according to their documentation. If you put them ALL in the SAME directory as DWdecode, then you do not HAVE to modify the prefs file to get started.

Modify DWdecode.prefs to use YOUR decoding programs, etc.

See

Run time Options & Internal Defaults

for a complete description of

all the possible options, variables, etc. and how to specify them.

Also look at the included files so that you will know how to add to them if necessary.

Run from a CLI or directory utility specifying a source directory and a destination directory.

IF YOU INSTALL DIFFERENT PROGRAMS or LOCATE THEM ELSEWHERE:

Change these in YOUR prefs file as necessary (these are MY values):

```
Base64cmd='AmitTCP:bin/Base64Decode %sf %df USEMINLEN > nil:'
```

```
B64cmdFail=20 #Return Code of 'DISK FULL' for Base64Decode
```

```
BinHexCmd='Utils:BHD data=%DF %SF'
```

```
BinCmdFail=20 #Return Code of 'DISK FULL' for BHD
```

```
ScanProg='Utils:FlashFind'
```

```
UUcmd='AmitTCP:bin/UUout %sf BUFSIZE=150 IGNORETERMINATION USEBASENAME > nil:'
```

```
UUcmdFail=10 #Return Code of 'DISK FULL' for UUOUT
```

## 1.8 Filenoting of Input Files

I filenote all of the files in the SOURCE= directory(s). The filenote is in the following format:

```
xyy SUBJECT-LINE-FOUND-IN-THIS-FILE
```

```
  where x='S' means the file was skipped due to a match on SKIPSUBJECT
```

```
        x='F' means the file was skipped due to a match on SKIPFROM
```

```
        x='O' means the file is older than the value for KEEP_DAYS
```

```
        x=' ' for all others.
```

```
  where yy=' ' for text files
```

```
        yy='UU' for UU encoded files
```

```
        yy='64' for Base64 encoded files
```

```
        yy='BH' for BinHex encoded files
```

This makes it fairly easy to check if files have been properly identified. I suggest using Dopus (or another), sorting the files by comment and examining them. This makes it easy to ensure that all 'text' files are really text. I look for high byte counts and then check them. Usually I find that someone has quoted the first 400 lines of an encoded file;) Doing this also makes it

easy to look at the subject lines and see if parts were properly put together.

Needless to say, if you specify some of the DELETExxxx options, there may be nothing to examine;)

## 1.9 Run time Options & Internal Defaults

Command line options override the prefs file and the prefs file options will override the internal defaults. ←

All commands can be specified using the ALIAS or the fullname.

Commands have a 'TYPE' indicated.

'B' is Boolean (YES/NO, TRUE/FALSE, ON/OFF, 1/0)

'N' is Numeric (any positive integer)

'F' is a file. The program checks for its existence

'S' is a character string. Any alphanumeric string is accepted

All options with a type of 'B' may be turned ON by specifying just the command or its alias.

---

FORMAT of documentation: (options are grouped according to function)

Command-Option Alias Type Default-Value

---

A complete alphabetical listing can be gotten by invoking DWDecode as:

DWDecode -h OR DWDecode ? OR DWDecode HELP

---

Decoders and their Failure Codes

Files used for Determining Decoding Information

Variables used for Determining Decoding Information

Files used for Setting Runtime Options

Directories and Input and Output

Filtering of Input Files

---

Report Control Options

Delete Options

Debugging Options

Logging Options

Miscellaneous Runtime Options

Making DWDecode Smarter

Summary List

Index of Options

## 1.10 Decoders and their Failure Codes

```
Base64Cmd    BC  F  Base64Decode %SF %DF USEMINLEN > nil:
B64cmdFail  BF  N  20

BinHexCmd    BHC F  Xbin -v %SF
BincmdFail   BHF N  0

UUcmd        UC  F  UUout %SF BUFSIZE=150 IGNORETERMINATION USEBASENAME > nil:
UUcmdFail    UF  N  10
```

---

These options control which decoders are used and what is considered a FATAL error such as 'Disk Full'. If you use different decoders you will have to construct your own command line string for them. They accept the following substitution variables:

%SF is replaced with the complete name (including path) of the input file  
 %DF is replaced with the complete name (including path) of the output file  
 %SD is replaced with the name of the input directory  
 %DD is replaced with the name of the output directory

Likewise, you will HAVE to determine the appropriate xxxcmdFail value. The easiest way is just to direct decoding to a floppy disk and see what value is returned when the disk fills up and you select CANCEL on the requestor.

I do a change directory to the output prior to invoking the decoders, so if yours does not require an output filename AND used the current directory you will not need to specify it.

If you do NOT place them in the same directory as DWDecode, they you must specify the path information also. Such as:

```
Base64Cmd='AmiTCP:bin/Base64Decode %SF %DF USEMINLEN > nil:'
```

---

## 1.11 Files used for Determining Decoding Information

Base64TypesFile	B64F	F	DWdecode.Base64.Types
BinTypesFile	BTF	F	DWdecode.Binary.Types
EncTypesFile	ETF	F	DWdecode.Encode.Types
SectionTypesFile	STF	F	DWdecode.Section.Types

---

These FILES contain information for determining how a file is encoded, where parts begin and end, and what words in a subject line MIGHT be a valid part name. See the individual file for further documentation. These files are required and are part of the DWDecode distribution.

## 1.12 Variables used for Determining Decoding Information

MaxHeader	MH	N	25
MinEncoded	ME	N	4

---

After the output of the SCAN program has been checked, what are usually text files are searched looking for lines that might be UUencoded or B64 encoded. Usually, any found will be part xx/yy of files.

The MAXHEADER values says to only read nn lines looking for encoded lines.

The MINENCODED value says that there MUST be at least nn lines that seem to be the SAME kind of encoding and that no other lines may appear within this group of nn lines. The default values seem to work well. If you see encoded files being indicated as text files, you can modify these values.

## 1.13 Files used for Setting Runtime Options

PrefsFile	P	F	DWdecode.Prefs
-----------	---	---	----------------

---

This file is used to set your personal preferences. You can have as many different preference files as you wish. Just specify:

```
DWDecode P='pref-file-this-time'
```

---

to use a file with a different name. This is the VERY FIRST option that is checked from the command line and is used (if present) to load the current runs options.

## 1.14 Directories and Input and Output

Source	S	F	(NO default value)
Recursive	REC	B	YES
Dest	D	F	(NO default value)
FileNote	FN	S	%LLQ2 : %SUBJECT
Replace	REP	B	NO

---

SOURCE= and DEST= are used as the input for the USENET news files and as the output directory for decoded files. The RECURSIVE option controls whether SOURCE= is treated as a single directory or if it is considered as the highest level of a directory tree. RECURSIVE=YES will result in decoding ALL of the files in SOURCE= plus ALL of the files in ALL of the subdirectories of SOURCE=

FILENOTE= controls the format of the filenote attached to decoded output files. Any character string is valid. However, substitution variables MUST be separated from other characters by at least 1 blank.

The string may contain the following substitution variables:

For the following examples assume input file is from directory:

```
INTERNET:UUNews/alt/binaries/pictures/animals/cute/fuzzy
```

%LLQn is replaced by the last n levels of the INPUT directory name.  
if n is NOT specified the default is n=1

i.e. specifying FN='%LLQ2 : keep' would result in filenotes of  
'cute fuzzy : keep'

%HLQn is replaced by the first n levels of the INPUT directory name.  
if n is NOT specified the default is the top level only (n=1)

i.e. specifying FN='%HLQ2 : keep' would result in filenotes of  
'INTERNET UUNews : keep'

%QUALxy is replaced with a portion of the INPUT directory name starting with the xth level for y levels. If xy are not specified then the COMPLETE directory name is used. If y is missing, then all levels starting with the xth are used.

i.e. specifying FN='%QUAL53 : keep' would result in  
'pictures animals cute : keep'

---

%DATE is replaced with the date from the Date: HEADER in the input file.

%FROM is replaced with the value from the From: HEADER in the input file.

%SUBJECT is replaced with the Subject: HEADER in the input file.

%SOURCE is replaced with the FULL source directory name. If RECURSIVE=YES this will be the actual subdirectory name.

REPLACE=YES says to replace existing files in the output directory with new ones from the input directory. This may result in the same file being created multiple times each time overlaying the previously created one. With all of the cross posting that occurs, this can be a considerable nuisance. However, there ARE times that you may wish to turn this on.

## 1.15 Filtering of Input Files

KeepDays	KD	N	30
MarkOld	MO	B	YES
SkipFrom	SF	B	YES
SkipFromFile	SFF	F	DWdecode.Skip.Froms
SkipSubject	SS	B	YES
SkipSubjectFile	SSF	F	DWdecode.Skip.Subjects

---

SKIPFROM=YES uses the values in the SKIPFROMFILE to bypass processing of certain input files. Files matching a SKIPFROMFILE value will have an 'F' in their filenote comment string. Use Dopus or another directory to check these.

If DELETEDFROM=YES then these files will be deleted PRIOR to any decoding.

The SKIPSUBJECT=YES and SKIPSUBJECTFILE= options also work together in the same way. The input files that match will have an 'S' in their comment.

These will disappear without ever being seen if DELETESUBJECT=YES.

KEEPDAYS=nn specifies how many days (based on the Date: header) to keep an input file. If MARKOLD=YES then these files will go through the normal decoding logic and then have 'O' (old) placed in their filenote comments.

If DELETEOLD=YES, then they will be deleted instead of being marked.

---

## 1.16 Report Control Options

ShowDecode	SHDE	B	YES
ShowDirs	SHDI	B	YES
ShowFiles	SHF	B	YES
ShowParts	SHP	B	YES
ShowStats	SHS	B	YES
DumpOpts	DO	B	NO

---

These options control which reports appear on the console or log (see LOG=)

SHOWDIRS switches on/off the complete list of ALL directories processed  
 SHOWFILES switches on/off the reporting by file  
 SHOWPARTS switches on/off the reporting by partname  
 SHOWDECODE switches on/off the reporting of the decoding process  
 SHOWSTATS switches on/off the reporting of various stats for each directory

DUMPOPTS switches on/off the reporting of the runtime options

## 1.17 Delete Options

SUGGESTION: Don't turn these options on until you are REALLY secure with my program. Test SKIP options thoroughly BEFORE making them into DELETES.

---

DeleteAfter	DELA	B	NO
DeleteDups	DELD	B	NO
DeleteFrom	DELF	B	NO
DeleteOld	DELO	B	NO
DeleteSubject	DELS	B	NO
DeleteText	DELT	B	NO
XDELETE	XD	B	NO

---

These options control which (if any) input files are deleted after decoding and/or filtering.

DELETEDAFTER=YES deletes all files used in a 'successful' decode.  
 DELETEDUPS=YES works in combination with REPLACE=  
     If REPLACE=NO then DELETEDUPS will delete files that specify  
     an output file that ALREADY exists.  
     If REPLACE=YES then DELETEDUPS will delete files that name  
     SAME output file as one decoded in the current run.

DELETETEXT=YES deletes any files that DWDecode 'thinks' are just text.

---

DELETEFROM=YES deletes any files identified during the SKIPFROM logic.

DELETESUBJECT=YES deletes files identified by SKIPSUBJECT logic.

DELETEOLD=YES deletes any files identified as older than keepdays, but this is done AFTER attempting to decode them.

XDELETE=YES turns ON ALL DELETE options. Very dangerous, but it's there.

## 1.18 Debugging Options

These are probably useless to most of you, but since I left them in, I thought I should document them--a very little;)

---

Debug	DB	B	NO
DumpBase64Types	D64T	B	NO
DumpBinTypes	DBT	B	NO
DumpDates	DD	B	NO
DumpEncTypes	DET	B	NO
DumpFiles	DF	B	NO
DumpFroms	DFR	B	NO
DumpMisc	DM	B	NO
DumpOpts	DO	B	NO
DumpPhase2	DP2	B	NO
DumpSectionTypes	DST	B	NO
DumpSkipFroms	DSF	B	NO
DumpSkipSubjects	DSS	B	NO
DumpSubjects	DS	B	NO
TrapAllBegs	TAB	B	NO
TrapAllEncs	TAE	B	NO
TrapAllSecs	TAS	B	NO

---

Debug	- turns on ALL debugging options
DumpBase64Types	- writes the Base64TypesFile to the console/log
DumpBinTypes	- writes the BinTypesFile to the console/log
DumpDates	- writes ALL Date: header lines to the console/log
DumpEncTypes	- writes the EncTypesFile to the console/log
DumpFiles	- writes the Files. array to the console/log
DumpFroms	- writes ALL From: header lines to the console/log
DumpMisc	- writes the MiscInfo. array to the console/log
DumpOpts	- writes all of the runtime options to the console/log
DumpPhase2	- writes various arrays to the console/log
DumpSectionTypes	- writes the SectionTypesFile to the console/log
DumpSkipFroms	- writes the SkipFromFile to the console/log
DumpSkipSubjects	- writes the SkipSubjectsFile to the console/log
DumpSubjects	- writes ALL Subject: header lines to the console/log

---



TrapAllBegs - writes ALL lines containing 'begin' to the console/log  
 TrapAllEncs - writes ALL 'Content-Transfer-Encoding:' lines  
 TrapAllSecs - writes ALL lines containing 'section' to console/log

## 1.19 Logging Options

Log	L	B	YES
LogFile	LF	S	DWdecode.Log

---

LOG=YES says to write reports to a file instead of the console  
 LOGFILE is used to specify the name of the log file.

## 1.20 Miscellaneous Runtime Options

DWassignIN	DWI	S	DW:
DWtemp	DWT	F	T:
Quiet	Q	B	NO
RunBack	RB	S	run > nil:
ScanOnly	SO	B	NO
ScanProg	SP	F	FlashFind

---

DWassignIN - within the program I use this assign for the input directory. you would only need to change it if there was a conflict with an existing assign - it is removed at the end of program.

DWtemp - this is used as the target of temporary files. Might need to be changed if T: can't handle the amount of bytes necessary?

QUIET - if set to YES will suppress almost all console messages. I recommend leaving it set to NO so that you have something to watch as the decode process progresses.

RunBack - this runs QuickSort. You could change it to use c:asynchrn or some other program -- why I'm not really sure.

ScanOnly - if set to YES will have the program do everything EXCEPT the actual decoding and deleting of files. If you are nervous about running this or just curious about the output - try this. It was mostly useful when I was doing the coding.

ScanProg - originally I had intended allowing for different Scan/search

---

programs, but the AmigaDos Search command was toooooo slow, and others wrote strange characters. FlashFind was fast AND it was easy to parse the output. So this option MUST be FlashFind with its path (if you put in a different directory)

## 1.21 Making DWDecode Smarter

TrapNewEncs	TNE	B	YES
TrapNewEncsFile	TNEF	S	DWdecode.Trap.Encodes
TrapNewSecs	TNS	B	YES
TrapNewSecsFile	TNSF	S	DWdecode.Trap.Sections

---

TrapNewEncs - enables trapping of NEW 'Content-Transfer-Encoding:' lines. This enables you to add them as you discover them.

TrapNewEncsFile is the file where the NEW encoding lines are logged.

TrapNewSecs - enables trapping of NEW section lines. These are lines that start with the word section and do not match ones in the STF

TrapNewSecsFile is the file where the NEW section lines are logged

SUGGESTION: Check these files occasionally and add the necessary info to the corresponding file. Also, I would appreciate it if you would send any new section lines or encoding lines to me (email) and I will add them for future updates.

## 1.22 Summary List

Command-Option	Alias	Internal Value
----------------	-------	----------------

---

B64cmdFail	BF	20
Base64Cmd	BC	Base64Decode %SF %DF USEMINLEN > nil:
Base64TypesFile	B64F	DWdecode.Base64.Types
BinCmdFail	BHF	20
BinHexCmd	BHC	BHD data=%DF %SF
BinTypesFile	BTf	DWdecode.Binary.Types
Debug	DB	NO
DeleteAfter	DELA	NO
DeleteDups	DELD	NO
DeleteFrom	DELF	NO
DeleteOld	DELO	NO

---

```

DeleteSubject      DELS      NO
DeleteText         DELT      NO
Dest               D          == none ==
DumpBase64Types   D64T      NO
DumpBinTypes       DBT       NO
DumpDates          DD        NO
DumpEncTypes       DET       NO
DumpFiles          DF        NO
DumpFroms          DFR       NO
DumpMisc           DM        NO
DumpOpts           DO        NO
DumpPhase2         DP2       NO
DumpSectionTypes  DST       NO
DumpSkipFroms     DSF       NO
DumpSkipSubjects  DSS       NO
DumpSubjects       DS        NO
DWassignIN        DWI       DW:
DWtemp             DWT       T:
EncTypesFile      ETF       DWdecode.Encode.Types
FileNote           FN        %LLQ2 : %SUBJECT
KeepDays           KD        30
Log                L         YES
LogFile            LF        DWdecode.Log
MarkOld            MO        YES
MaxHeader          MH        25
MinEncoded         ME        4
PrefsFile          P         DWdecode.Prefs
Quiet              Q         NO
Recursive          REC       YES
Replace            REP       NO
RunBack            RB        run > nil:
ScanOnly           SO        NO
ScanProg           SP        FlashFind
SectionTypesFile  STF       DWdecode.Section.Types
ShowDecode         SHDE      YES
ShowDirs           SHDI      YES
ShowFiles          SHF       YES
ShowParts          SHP       YES
ShowStats          SHS       YES
SkipFrom           SF        YES
SkipFromFile       SFF       DWdecode.Skip.Froms
SkipSubject        SS        YES
SkipSubjectFile    SSF       DWdecode.Skip.Subjects
Source             S         == none ==
TrapAllBegs        TAB       NO
TrapAllEncs        TAE       NO
TrapAllSecs        TAS       NO
TrapNewEncs        TNE       YES
TrapNewEncsFile    TNEF      DWdecode.Trap.Encodes
TrapNewSecs        TNS       YES
TrapNewSecsFile    TNSF      DWdecode.Trap.Sections
UUcmd              UC        UUout %SF BUFSIZE=150 IGNORETERMINATION USEBASENAME > ↔
  nil:
UUcmdFail          UF        10
XDELETE            XD        NO

```

## 1.23 Index of Options

- B64cmdFail
- Decoders and their Failure Codes
  - Base64Cmd
- Decoders and their Failure Codes
  - Base64TypesFile
- Files used for Determining Decoding Information
  - BincmdFail
- Decoders and their Failure Codes
  - BinHexCmd
- Decoders and their Failure Codes
  - BinTypesFile
- Files used for Determining Decoding Information
  - Debug
- Debugging Options
  - DeleteAfter
- Delete Options
  - DeleteDups
- Delete Options
  - DeleteFrom
- Delete Options
  - DeleteOld
- Delete Options
  - DeleteSubject
- Delete Options
  - DeleteText
- Delete Options
  - Dest
- Directories and Input and Output
  - DumpBase64Types
- Debugging Options
  - DumpBinTypes
- Debugging Options
  - DumpDates
- Debugging Options
  - DumpEncTypes
- Debugging Options
  - DumpFiles
- Debugging Options
  - DumpFroms
- Debugging Options
  - DumpMisc
- Debugging Options
  - DumpOpts
- Debugging Options
  - DumpOpts
- Report Control Options
  - DumpPhase2
- Debugging Options
  - DumpSectionTypes
- Debugging Options
  - DumpSkipFroms
- Debugging Options
  - DumpSkipSubjects
- Debugging Options

- DumpSubjects
- Debugging Options
  - DWassignIN
- Miscellaneous Runtime Options
  - DWtemp
- Miscellaneous Runtime Options
  - EncTypesFile
- Files used for Determining Decoding Information
  - FileNote
- Directories and Input and Output
  - KeepDays
- Filtering of Input Files
  - Log
- Logging Options
  - LogFile
- Logging Options
  - MarkOld
- Filtering of Input Files
  - MaxHeader
- Variables used for Determining Decoding Information
  - MinEncoded
- Variables used for Determining Decoding Information
  - PrefsFile
- Files used for Setting Runtime Options
  - Quiet
- Miscellaneous Runtime Options
  - Recursive
- Directories and Input and Output
  - Replace
- Directories and Input and Output
  - RunBack
- Miscellaneous Runtime Options
  - ScanOnly
- Miscellaneous Runtime Options
  - ScanProg
- Miscellaneous Runtime Options
  - SectionTypesFile
- Files used for Determining Decoding Information
  - ShowDecode
- Report Control Options
  - ShowDirs
- Report Control Options
  - ShowFiles
- Report Control Options
  - ShowParts
- Report Control Options
  - ShowStats
- Report Control Options
  - SkipFrom
- Filtering of Input Files
  - SkipFromFile
- Filtering of Input Files
  - SkipSubject
- Filtering of Input Files
  - SkipSubjectFile
- Filtering of Input Files
  - Source

---

- Directories and Input and Output
  - TrapAllBEGs
- Debugging Options
  - TrapAllEncs
- Debugging Options
  - TrapAllSecs
- Debugging Options
  - TrapNewEncs
- Making DWDecode Smarter
  - TrapNewEncsFile
- Making DWDecode Smarter
  - TrapNewSecs
- Making DWDecode Smarter
  - TrapNewSecsFile
- Making DWDecode Smarter
  - UUcmd
- Decoders and their Failure Codes
  - UUcmdFail
- Decoders and their Failure Codes
  - XDELETE
- Delete Options

## 1.24 History

### v1.02 Fixes:

Finally, I HOPE, have fixed adding libraries during init. Several people told me I hadn't... must have been some other program;)

### v1.01 Fixes:

Discovered that REXXTricks.Library uses the SAME function name as QuickSort, BUT in a different format.

Added code to test if REXXTricks is present and to call QSORT in that format if necessary instead of W.L's.

The SORTS are faster if REXXTricks.Library is NOT present, so try to do decoding prior to anything that might add it.

Can NOW place in directory named DWDECODE and have it WORK!! Really stupid error - thanks to Paul Copsey for finding this one.

Now correctly deletes parts greater than 1 of multipart input files when DeleteDups is specified.

If DWdecode MUST resort to using just Subject: line information will now consider the SAME partname appearing multiple times in the subject, IFF it is the only name present, as THE name to use.

Added support for BinHex files (only 1/1 type files).

New variables w/defaults:

BinHexCmd='BHD data=%DF %SF'

BinCmdFail=20

Removed unused variable DumpB64Files from preferences/help.

---

Now handles PC type lines with Ctrl-M's

Now filenotes output files containing parens in their names

Now writes FileNote errors to log/console as requested

Made maximum filename length on report = 30 characters.

Added message referring user to logfile for errors.

Added Guide format of 'documentation'.

v1.0 Initial Release

## 1.25 Bugs

As of the date of uploading this program, I have stopped being able to break it. This does not mean it is bug free. If you find any, please email me so that I can correct it for others. Also, when you find NEW Section lines or MIME encoding types, I would appreciate you emailing them on to me and I will add them to future updates, changes, etc.

Thanks.

## 1.26 Thanks

Thank you's to:

Willy Langeveld for REXXArpLib, VLT, QuickSort, et. al.

Frank Würkner for FlashFind

Nicolas Dade for UUout

Edmund Vermeulen for Base64coders

Nik Soggia for binhex374

Stephan Sürken for Text2Guide

(without it I would not have done a Guide format documentation)

Marcel Beck for YAM - the BEST way to contact me

Holger Kruse for MIAMI - the BEST way to hook YAM to the net;)

AND ...

---

all of the other shareware, freeware, etc. programmers for the AMIGA  
who make it possible for US to still use our favorite machine;)

Dick Whiting  
April 02, 1997

---