

**Citadel**

**COLLABORATORS**

	<i>TITLE :</i> Citadel		
<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>Citadel</b>	<b>1</b>
1.1	Citadel.Guide . . . . .	1
1.2	Table Of Contents . . . . .	1
1.3	MAIN . . . . .	5
1.4	Citadel History . . . . .	5
1.5	What is Citadel-68K . . . . .	6
1.6	Support Systems . . . . .	6
1.7	Hardware required . . . . .	7
1.8	Requirements . . . . .	7
1.9	Citadel Error . . . . .	8
1.10	Limits . . . . .	11
1.11	CONFIG . . . . .	11
1.12	SYSOP_PARAMETERS . . . . .	12
1.13	#LOCK-PORT . . . . .	14
1.14	#QWKWORKAREA . . . . .	14
1.15	#QWKFILEAREA . . . . .	15
1.16	#QWKMAXROOM . . . . .	15
1.17	#QWKMAXPACKET . . . . .	15
1.18	#QWKNAME . . . . .	15
1.19	#QWKLOCATION . . . . .	15
1.20	#SYSOP-ARCHIVE . . . . .	15
1.21	#SYSPASSWORD . . . . .	15
1.22	#SYSOPNAME . . . . .	16
1.23	#MIRRORMSG . . . . .	16
1.24	#SHARED-ROOMS . . . . .	16
1.25	#NET_AREA_SIZE . . . . .	16
1.26	#MAX_NET_FILE . . . . .	16
1.27	#EDITOR . . . . .	16
1.28	#CLOCK . . . . .	16
1.29	#SYSBAUD . . . . .	16

---

---

1.30	#SERIAL_7WIRE	16
1.31	#DIRECTTOCHIP	16
1.32	#SERIALDEVICE	17
1.33	#UNITNUMBER	17
1.34	USER_PARAMETERS	17
1.35	unlogged users	18
1.36	PRIVILEGES	19
1.37	user characteristics	19
1.38	#BASEROOM	19
1.39	#MAINFLOOR	20
1.40	areas	20
1.41	#HELPAREA	21
1.42	#LOGAREA	21
1.43	#MSGAREA	21
1.44	#MSG2AREA	22
1.45	#FLOORARE	22
1.46	#AUDITAREA	22
1.47	#HOMEAREA	23
1.48	CIT_MESSAGES.SYS	23
1.49	CALLOG.SYS	24
1.50	FILELOG.SYS	25
1.51	DOORUSE.SYS	25
1.52	#HOLDAREA	26
1.53	#EDIT-AREA	26
1.54	#EDITOR	26
1.55	#NETAREA	26
1.56	#NET_RECEPT_AREA	27
1.57	#NET_AREA_SIZE	27
1.58	#MAX_NET_FILE	27
1.59	#DOMAINAREA	27
1.60	basic system parameters	27
1.61	#CRYPTSEED	28
1.62	Safe Configuration Parameters	29
1.63	#NODEID	29
1.64	#NODENAME	30
1.65	#NODETITLE	31
1.66	banners	31
1.67	The Amiga Zone	31
1.68	#LOGSIZE	31

---

---

1.69	#MAXROOMS	32
1.70	#MAIL-SLOTS	32
1.71	#MESSAGEK	33
1.72	#SCAN-NET-MESSAGES	34
1.73	Utilities	34
1.74	Installation	34
1.75	C86Net	34
1.76	BBS List	34
1.77	Citadel	34
1.78	Files	34
1.79	debug.sys	34
1.80	crash.sys	35
1.81	#ROOMAREA	35
1.82	ALIASES.SYS	37
1.83	badnames.sys	37
1.84	badpasswords.sys	37
1.85	badpeople.sys	37
1.86	badwords.sys	38
1.87	citadoor.sys	38
1.88	ctdlarch.s	38
1.89	ctldir.sys	39
1.90	ctdlfwd.sys	39
1.91	ctdlinfo.sys	39
1.92	ctdlmodr.sys	39
1.93	ctdlprot.sys	39
1.94	protocols	40
1.95	ctdlroom.sys	41
1.96	DExxx.SYS	41
1.97	results.sys	41
1.98	DISCARD	42
1.99	CTDLCNFG.SYS	42

---

# Chapter 1

# Citadel

## 1.1 Citadel.Guide

Table Of Contents

MAIN

Citadel History

What is Citadel-68K

CONFIG

SYSOP\_PARAMETERS

USER\_PARAMETERS

Utilities

Installation

C86Net

BBS List

Citadel

Files

#ROOMAREA

## 1.2 Table Of Contents

MAIN

Citadel.Guide

1.

MAIN

2.

---

---

- Citadel History
  - 3.
    - What is Citadel-68K
      - 3.1.
    - Support Systems
      - 3.2.
    - Hardware required
      - 3.3.
    - Requirements
      - 3.4.
    - Citadel Error
      - 3.5.
  - Limits
    - 4.
  - CONFIG
    - 5.
      - SYSOP\_PARAMETERS
        - 5.1.
      - #LOCK-PORT
        - 5.2.
      - #QKWORKAREA
        - 5.3.
      - #QKFILEAREA
        - 5.4.
      - #QKMAXROOM
        - 5.5.
      - #QKMAXPACKET
        - 5.6.
      - #QKNAME
        - 5.7.
      - #QKLOCATION
        - 5.8.
      - #SYSOP-ARCHIVE
        - 5.9.
      - #SYSPASSWORD
        - 5.10.
      - #SYSOPNAME
        - 5.11.
      - #MIRRORMSG
        - 5.12.
      - #SHARED-ROOMS
        - 5.13.
      - #NET\_AREA\_SIZE
        - 5.14.
      - #MAX\_NET\_FILE
        - 5.15.
      - #EDITOR
        - 5.16.
      - #CLOCK
        - 5.17.
      - #SYSBAUD
        - 5.18.
      - #SERIAL\_7WIRE
        - 5.19.
      - #DIRECTTOCHIP
        - 5.20.
      - #SERIALDEVICE

---

```
5.21.
#UNITNUMBER
6.
USER_PARAMETERS
6.1.
unlogged users
6.2.
PRIVILEGES
6.3.
user characteristics
6.4.
#BASEROOM
6.5.
#MAINFLOOR
6.6.
areas
6.7.
#HELPAREA
6.8.
#LOGAREA
6.9.
#MSGAREA
6.10.
#MSG2AREA
6.11.
#FLOORARE
6.12.
#AUDITAREA
6.13.
#HOMEAREA
6.14.
CIT_MESSAGES.SYS
6.15.
CALLOG.SYS
6.16.
FILELOG.SYS
6.17.
DOORUSE.SYS
6.18.
#HOLDAREA
6.19.
#EDIT-AREA
6.20.
#EDITOR
6.21.
#NETAREA
6.22.
#NET_RECEPT_AREA
6.23.
#NET_AREA_SIZE
6.24.
#MAX_NET_FILE
6.25.
#DOMAINAREA
6.26.
basic system parameters
6.27.
```

---

---

#CRYPTSEED  
6.28.  
Safe Configuration Parameters  
6.29.  
#NODEID  
6.30.  
#NODENAME  
6.31.  
#NODETITLE  
6.32.  
banners  
6.33.  
The Amiga Zone  
6.34.  
#LOGSIZE  
6.35.  
#MAXROOMS  
6.36.  
#MAIL-SLOTS  
6.37.  
#MESSAGEK  
6.38.  
#SCAN-NET-MESSAGES  
7.  
Utilities  
8.  
Installation  
9.  
C86Net  
10.  
BBS List  
11.  
Citadel  
12.  
Files  
12.1.  
debug.sys  
12.2.  
crash.sys  
13.  
#ROOMAREA  
13.1.  
ALIASES.SYS  
13.2.  
badnames.sys  
13.3.  
badpasswords.sys  
13.4.  
badpeople.sys  
13.5.  
badwords.sys  
13.6.  
citadoor.sys  
13.7.  
ctdlarch.s  
13.8.  
ctlddir.sys

---

```
13.9.
ctdlfwd.sys
13.10.
ctdlinfo.sys
13.11.
ctdlmodr.sys
13.12.
ctdlprot.sys
13.13.
protocols
13.14.
ctdlroom.sys
13.15.
DExxx.SYS
13.16.
results.sys
13.17.
DISCARD
13.18.
CTDLCNFG.SYS
```

## 1.3 MAIN

The documentation contained is a collection of information based on the original Citadel 86 documentation by Hue, JR. The mistakes are mine.

It is pretty much a complete reference manual and every attempt is being made to make this a complete manual with all details explained so that even the most novice of users can understand how to setup and run a bbs. The most important thing is to read this documentation and give it a try!

## 1.4 Citadel History

What is Citadel? Citadel is a Freeware project. The source, executables and all the documentation are available for no cost to you. If you paid for this, someone is ripping you off.

Citadel was written in mid-December 1981 by CrT. Miraculously, it ran three days unattended over New Year's, collecting some remarkably favorable reactions. During the months that it ran at 633-3282 (ODD-DATA), Citadel became one of the more popular BBs in town, and there was some disappointment when a hardware failure forced the system down in February of 1982. But in January CrT had published the source code in BDS C, putting it in the public domain.

David Mitchell brought up the next incarnation of the Citadel program in April of 1982, running on hardware provided by Richard Knox. Called the Island Communication System, it is located on Bainbridge Island in Puget Sound. ICS has about 30 regular users

---

and about 120 log entries. Newcomers find it easy to learn, and often leave messages praising it. Some of the system's daily users are in Boston.

Citadel is descended from DandD.pas, an adventure game editor/driver. It is arranged as a series of rooms, starting with the LOBBY. In each room the user can read existing messages and leave more. The system was brought up with only one room, the LOBBY. Additional rooms were created by the users, with room names appropriate to the topics covered.

Environment: Citadel has had a checkered past. It first ran on a 64K Heath H89 with Magnolia CP/M, Hayes Smartmodem (plus an acoustic on another port) and BDS C V1.32.

As time went on, Citadel was ported to the Amiga, Atari, and even the MAC. Citadel runs on many platforms and since the source is available will probably run on most future ones too.

## 1.5 What is Citadel-68K

Citadel-68K(also know as Amiga Citadel) is a single user ↔  
BBS  
program. It is a direct decendant of the 3.42 Citadel 86 by Hue, JR in Minnesota.

Citadel comes in two flavors, a 68000 based version that will run on any Amiga and an optimized 68030 based. Citadel is able to run on any Amiga model and under any OS from 1.2 to the latest. The CTDL(main BBS executable) and CONFIG(BBS configuration tool) come in the two forms, the utilities come in the 68000 version only. The Amiga Citadel is a direct port from the IBM Citadel 86 by Hue Jr. It originally was done by Jay Johnston, who did not have the time to continue it so I, Tony Preston now maintain it. I don't have the time either, but I try...

Support Systems

Hardware required

Requirements

Citadel Error

Limits

## 1.6 Support Systems

Probably the hardest part of running a Citadel is ↔  
getting  
started. Citadel is not the most common of BBS programs even though it is free.

---

You should be able to read this document and setup your configuration file, run

```
CONFIG
```

and then startup the BBS with no problems. Since this rarely happens and having a helping hand from someone that has already done what you are trying to do can make things easier, you might want to try one of these three places for help and information.

The first is The Amiga Zone, my BBS(609-953-8159). It is available 24 hours and is where you will find the most support and help. I will often chat with people that call for help and always try to answer mail messages promptly. Since calling long distance may not be an option for you, check around in your area and see if you can find a local Citadel where you can take major advantage of the networking features built into Citadel! The

```
C86Net
```

```
contains
```

several support rooms where you can post your questions and usually get quick answers. These rooms are "Citadel 68K" and "Sysop Only". If your local sysop will let you have some Long Distance credit you can send me domain mail at "tony preston@The Amiga Zone.NJ". You will learn more about domain mail later. There are many Citadels active on the network so you might check the

```
BBS List
```

```
included in
```

this document to see if one is local to you. finally, you can send me mail via Internet. I will answer the mail quickly monday thru friday. Anything sent over the weekend will wait till monday. you can reach me at "apreston@isd.csc.com" or "tony-preston@cup.portal.com".

## 1.7 Hardware required

The minimum configuration for  
Citadel

```
is a 512K Amiga with 2
```

floppies. This will allow you to run the BBS, but probably not much more. There are some people that have run Citadel on such a small system. Most either expand their system or just quit running it. 1 MB of memory and a hard drive is really the practical limit. I have created and ran a test bbs on an A500 with 1 MB of memory and 2 floppies. I would recommend that you have 2 MBs and as a minimum a 20 MB HD for the BBS.

## 1.8 Requirements

Citadel will run on any Amiga Model. There are some minor problems with running CONFIG and fast memory on A3000s and A4000, but the work around is simply to run NOFastMem before running CONFIG. These may be fixed at any time, but since I do not have an A3000 or

---

A4000, I can't look for the problem.

Citadel does not need any external support software to run. It relies on the Operating System for 100% of the normal functions and is compatible with 1.2 through to the latest OS.

Citadel does not use alot of stack space, but will require that you have your stack set to 8K or larger. 8K is more than enough for even the largest and most complex Citadel. Citadel will make sure you have at least an 8K stack or it will quit with a 'Citadel Error'.

It is important to note is that you really should plan on a 24 hour BBS, with a dedicated phone line. A BBS that is available from 11pm to 6am is not going to be very popular. I would suggest that you do not even consider networking unless your BBS is on a regular schedule.

## 1.9 Citadel Error

Citadel is a complex system of functions. In any complex system, things go wrong. Citadel attempts to validate most things when it starts up.

Once you have the BBS up and running, you still may run into an occasional problem. The first thing to do is to collect sufficient information on what exactly is going on. Many times, if you look at the data you might be able to solve the problem yourself!

In general, if you get an error and this information does not tell you how to correct it, collect as much information as possible and report what happended either directly to me or in the Citadel 68k room. The first thing to look for is a file called  
debug.sys  
or  
crash.sys  
. These files should appear in either your audit area, the home area, or the location you started up Citadel. I usually will want the information in these files(even if it is just a cryptic one line message like "dependant variables mismatch", sometimes it tells me exactly where the problem is). The second thing I will tell you to do is turn on debug, Here is a general method I end up telling people:

- 1) go into the Sysop menu, turn on debug "D" option. You can also do this by typing ^D in the console window.

- 2) Shut down your Citadel, "X" option.

- 3) delete debug.sys in the audit area(or save it if it contains info I might need. At the least, edit the file and add some markers (like two lines of asterisks) at the end of the file.

4) Bring up Citadel and attempt to reproduce the problem. If you cannot do it locally, you might even ask a remote user to do it for you. leave debug on. Note: If you run CONFIG, debug is automatically turned off, repeat the above steps to turn it on again.

5) archive all the information(using something like lha) and arrange to get the information to me. I may call your BBS to download the file so make some arrangements in Citadel 68K so I know where it is.

The above may generate alot of output. Much of the output is cryptic and may not seem like anything understandable. It is mostly internal data and is useful to me.

From time to time, other errors may appear when you do something that you really should not do(like power down the modem and then power it up). These will generate errors like:

Error: [1]IOError = nnnn

Error: [2]IOError = nnnn

Reason: nnnn is a result code returned from a serial port i/o, usually a dropped carrier(small timing window for a race condition could cause this). The error is handled for 99% of the cases in a way that will cause Citadel to recover and reset. [1] is the case where I check to see what is in the serial port buffer, and [2] is when the actual read is done.

Error: Startup Error Code nn

Reason: something went wrong during system initialization. The reasons are:

1 - unable to open intuition.library, you must be 1.2 or greater to run Citadel.

2 - unable to open graphics.library, same as 1. This error also used to mean that the req.library was not in the libs: directory. This is no longer a requirement. Citadel does not depend on the req.library(versions 3.42.P15 or later anyway).

3 - Insufficient Stack space, Citadel versions 3.42.E19 and earlier required a large stack, much larger than needed (50K). Versions 3.42.P35 and later will require a 8K stack or less(I am still adjusting the values down). Citadel still requires the larger limit just to be cautious.

11 - Console Open Error. Catch all for console window errors If you are using #WBSCREEN, try without it.

25 - Open Serial Port Failed, Well, Citadel

---

could not get to the serial port (maybe something else has it open. Note: You will not get this error if you run Citadel while it is already running since it opens the serial port in a shared mode.

31 - Could not create a Port for timer communications, Low memory? Trashed system tables? Try a re-boot. This is one of those, "you should never get here". If you bug me with this type of problem, you had better have a full system configuration and a lot of details.

32 - could not create an I/O request. See 31.

33 - Open timer.device failed. See 31.

Note: In the serial port open errors, and in most cases with debug turned on, you will get a text error message of the form:

```
1:   Date - Dos Error:nnnn
2:   (some text as to what happened)
3:   (some text as to what happened) <-- you may get
only one line.
4:   Reason: <error text>
5:   Current Directory
```

Line 1: is the internal error code (less than 100), or the Dos error code.

Lines 2-3: will either be a command (like in the external protocols) and a text line, or just one line of text. External commands will display the text and command, most errors do not have an external command.

4: is the reason the error occurred (from the Exec routine Fault).

5: is the current directory. This is important if you are trying to setup a door for example and in the wrong directory.

If the problem is reproducible, do it several times and record all possible information, especially your system configuration! If it happens just once and you can not reproduce it, then still record what you can, check things like memory in use, what is running.

Note: If you have a problem that seems to happen often, realize that I rarely have a crash. Please check to see that something else is not causing the problem. Remove commodities, other programs and see if you can cause the problem without

---

that super-duper-whiz-bang mouse accelerator/screen blanker! It probably ain't Citadel! If you are running on a 512K system, you may just be running out of memory. While every attempt has been made to exit in a friendly manner, no guarentees...

## 1.10 Limits

limits... In practicality, there are some real physical limits ←  
 you  
 will have. Each of these limits can be difficult to adjust later so  
 some planning is important at this point. You must set a limit on  
 the number of users (  
     #LOGSIZE  
   ), rooms (  
     #MAXROOMS  
   ), and messages  
 (  
     #MESSAGEK  
   ). These parameters will directly determine the amount  
 of memory used while the BBS is running and the disk space needed to  
 support the message base and userlog.

## 1.11 CONFIG

To setup the BBS, you must first configure your system. ←  
 This  
 means taking the example configuration and tayloring it to your  
 liking. The example is based directly on  
 The Amiga Zone  
 . The  
 first thing you must do is chose a name for your BBS (  
     #NODENAME  
   ),  
 a default banner (see  
     banners  
     and  
     #NODETITLE  
   ), enter in your  
 Identification (  
     #NODEID  
   ). It is this basic information that users  
 and other Citadels will know your bbs by. Once you have an idea of  
 what the theme of your BBS is, you can apply it to the initial room  
 (  
     #BASEROOM  
   ), and floor (  
     #MAINFLOOR  
   ). These will be the initial  
 place that people are located at when they login to your BBS. Now  
 comes the real work. You must decide some  
 basic system parameters

that will define how much disk space your system will use. ←

This is

important since the smaller the message base is, the quicker messages will scroll off. Citadel has a fixed message base so that once you configure your system, the disk space usage is constant. You will never run out of message space, the BBS will reuse the existing space deleting the oldest messages to make room for the new ones.

Next we have the

USER\_PARAMETERS

which define the default

PRIVILEGES

for your users. These determine how open your system is (can a user create their own account or do you do it?). Whether people are able to use doors, or post messages to the network. If you restrict people, then they will have to ask you for the privilege (and you will have to give it to those you choose). If you make them the default, people will get them automatically, you then do not have to do anything. I setup my system to be as automatic as possible. People can create their own account and do not need to be validated. The example setup is configured that way. The most important user is the SYSOP, You. There are some parameters that make your life easier. the

sysop\_parameters

will take care of

those. Now we get to Network parameters which you can skip initially, but will eventually want to look into. Get your BBS up and running first before you worry about that.

The basic BBS has several

areas

you will want to setup. Most

people will setup a logical assignment that is the root of the BBS disk area (called

#HOMEAREA

) and make everything a subdirectory

off of that. Citadel is pretty flexible, if you are running from an A500 with 2 floppies, it will run, even if the size will be small!

CONFIG is simple to run. Once you have created the

CTDLCNFG.SYS

file, you just run CONFIG in the same directory. It will read each line, and report any errors. If there are errors, it will stop after the last line is read. If no errors, it will finish up its processing, possibly ask you some questions and create all the required files.

## 1.12 SYSOP\_PARAMETERS

There are alot of parameters to setup. Don't be ←  
overwhelmed!

Each has a simple description and parameters. Some are ok as the

default.

#LOCK-PORT

#QKWORKAREA

#QKFILEAREA

#QKMAXROOM

#QKMAXPACKET

#QKNAME

#QKLOCATION

#SYSOP-ARCHIVE

#SYSPASSWORD

#SYSOPNAME

#MIRRORMSG

#SHARED-ROOMS

#NET\_AREA\_SIZE

#MAX\_NET\_FILE

#EDITOR

#CLOCK

#SYSBAUD

#SERIAL\_7WIRE

#DIRECTTOCHIP

#SERIALDEVICE

#UNITNUMBER

#LOCK-PORT

#QKWORKAREA

#QKFILEAREA

#QKMAXROOM

#QKMAXPACKET

#QKNAME

```
#QWKLOCATION
#SYSOP-ARCHIVE
#SYSPASSWORD
#SYSOPNAME
#MIRRORMSG
#SHARED-ROOMS
#NET_AREA_SIZE
#MAX_NET_FILE
#EDITOR
#CLOCK
#SYSBAUD
#SERIAL_7WIRE
#DIRECTTOCHIP
#SERIALDEVICE
#UNITNUMBER
```

### 1.13 #LOCK-PORT

This parameter tells Citadel that you wish to lock the COMPUTER to MODEM speed to a particular value. It also causes CTS/RTS hardware handshaking to be used so that the BBS communicates with the modem at a single speed and the modem handles all the Modem to Modem speed negotiations. This parameter takes a single numeric parameter which must be 1 to 8. The values correspond to the speeds:

0 - 300	1 - 1200	2 - 2400	3 - 4800	
4 - 9600	5 - 14400	6 - 19200	7 - 38400	8 - 57600

### 1.14 #QWKWORKAREA

This parameter takes a quoted string as an argument. It defines the working area for the QWK data. If you wish to support QWK packet downloads (you must have external archivers available) then this needs to be defined to be a directory where the QWK processing can put temporary files.

---

## 1.15 #QWKFILEAREA

This parameter takes a quoted string as an argument. It defines the directory where the QWK processing saves the USER related data. You must define this if you wish to support QWK packet downloads (external archivers must be available).

## 1.16 #QWKMAXROOM

This parameter defines what the maximum number of rooms a user can scan at one time. Users can set their own personal value from one to this number. This parameter takes a single integer argument.

## 1.17 #QWKMAXPACKET

This parameter defines what the maximum messages a user can scan at one time. Users can set their own personal value from one to this number. This parameter takes a single integer argument.

## 1.18 #QWKNAME

This parameter defines a single quoted string that is passed to the QWK packet to define the name of the packet file.

## 1.19 #QWKLOCATION

This parameter defines a single quoted string passed in the QWK packet as the location of your BBS.

## 1.20 #SYSOP-ARCHIVE

This parameter defines a file where all sysop mail is archived. If this is defined, each mail message to you will be written to this file.

## 1.21 #SYSPASSWORD

This parameter defines a filename that has your sysop password. This password will allow you (or anyone you give the password to) to become a REMOTE Sysop. A Remote Sysop can do anything you can do from the console so use this wisely.

---

## 1.22 #SYSOPNAME

This is you... This parameter tells the BBS that you are the Sysop. You will have to create you account first, then add this to the CTDLNCFG.SYS and run CONFG again.

## 1.23 #MIRRORMSG

This parameter tells the BBS that you wish to have a mirrored message file. Basically, if you have the memory, copy your CTDLMSG.SYS to RAM:, and then start up the BBS, this parameter will allow the BBS to write to both this mirrored message file and the regular one. You are responsible for coping te current file to the mirrored one before the BBS starts up. In addition to this statement you need to include a  
#MSG2AREA  
to tell the BBS where the secondary message file is. This parameter takes a single integer value, 0 for off, 1 for on. If you were using this feature, then put "#MSG2AREA 1" in the CTDLNCFG.SYS file.

## 1.24 #SHARED-ROOMS

This parameter defines the maximum number of rooms a single node can share with you. Each entry takes up 6 bytes so the space requirements are minimal. The DATACHNG utility will allow you to modify this value(make it larger) so plan accordingly.

## 1.25 #NET\_AREA\_SIZE

## 1.26 #MAX\_NET\_FILE

## 1.27 #EDITOR

## 1.28 #CLOCK

## 1.29 #SYSBAUD

## 1.30 #SERIAL\_7WIRE

## 1.31 #DIRECTTOCHIP

---

## 1.32 #SERIALDEVICE

## 1.33 #UNITNUMBER

## 1.34 USER\_PARAMETERS

User parameters is a catch all for most of the parameters related to user. Since the BBS is about users, nearly everything could be put into this category. There are three sets of parameters. The first is the unlogged users parameters. This is all the parameters relating to a user that has not logged in yet. The second is the PRIVILEGES, the values given to a new user when their account is created. The last is the user characteristics.

Each of these parameters must be setup and will define the way your BBS operates.

```

unlogged users

PRIVILEGES

user characteristics

#BASEROOM

#MAINFLOOR

areas

#HELPAREA

#LOGAREA

#MSGAREA

#MSG2AREA

#FLOORARE

#AUDITAREA

#HOMEAREA

CIT_MESSAGES.SYS

CALLLOG.SYS

FILELOG.SYS

```

---

```
DOORUSE.SYS

#HOLDAREA

#EDIT-AREA

#EDITOR

#NETAREA

#NET_RECEPT_AREA

#NET_AREA_SIZE

#MAX_NET_FILE

#DOMAINAREA

basic system parameters

#CRYPTSEED

Safe Configuration Parameters

#NODEID

#NODENAME

#NODETITLE

banners

The Amiga Zone

#LOGSIZE

#MAXROOMS

#MAIL-SLOTS

#MESSAGEK

#SCAN-NET-MESSAGES
```

### 1.35 unlogged users

When a user first calls the BBS, they will get a set of default parameters that will define how the BBS operates until they login or create an account. If you do not allow them to create an account on their own, they will have to send you mail and you will have to do this manually (called account validation). Citadel allows you to operate either way. For unlogged users, the parameters are:

```
#UNLOGGED-WIDTH - The default width of a line
```

---

```
#LOGINOK          - Open/Close system control
#ENTEROK          - Can users enter messages while not logged in?
#READOK           - Can users read messages while not logged in?
#ANON-MAIL-LENGTH - Limit on anonymous mail length to prevent RUGGIES
#LOGIN-ATTEMPTS   - Limit on how many times a user can make a mistake
```

## 1.36 PRIVILEGES

This section defines the user privileges, defaults and all related parameters. These parameters will save you some time and effort. If you have doors and want everyone to be able to play, it does not make sense to have to give everyone the privilege. Instead use these parameters to set the defaults.

```
#DOORPRIVS       - Allow new users to have access to doors
#ROOMOK           - Allow users to be able to create new
rooms.
#ALLMAIL          - Control who can use mail
FILE-PRIV-DEFAULT - Allow users to have file up/down load
access
```

## 1.37 user characteristics

### 1.38 #BASEROOM

Citadels always have a minimum of three rooms. There is the Aide room, Mail room, and the initial room a caller starts out in called the base room.

Historically, the initial room was always called The Lobby. Most Citadels today have this configuration parameter which allows you to name that initial room.

This parameter is a string value obeying formatting directives and goes through the Citadel formatter, and you must limit yourself to 19 characters or less for this value. And one more note -- Citadel will append the '>' to this name when it prints the room prompt for this room, you don't have to put it in yourself. If you wished to emulate the old CP/M Citadel, you'd set baseRoom thus:

```
#BASEROOM "Lobby"
```

There is no default for this parameter.

---

## 1.39 #MAINFLOOR

MainFloor is analogous to #BASEROOM. Most Citadels have a base floor, just as it has an Aide> room, etc. This parameter allows you to name this base floor. This parameter is a string value which cannot be longer than 19 characters, and specifies the name of your base floor. So, if you want to name your base floor MAIN FLOOR, you'd have

```
#MAINFLOOR "MAIN FLOOR"
```

There is no default value for this parameter.

## 1.40 areas

The BBS is organized into what is called areas. These ←  
are  
directories that either Citadel creates files in, or uses to receive temporary files from a network session, or user action. There are parameters for each of the major areas.

```
#HOMEAREA
```

- The root location of the BBS.

```
#HELPAREA
```

- Help files(.HLP), menus, and banners

```
#LOGAREA
```

- User data files

```
#ROOMAREA
```

- Room related files

```
#MSGAREA
```

- Message base

```
#MSG2AREA
```

- Optional secondary Message base to speed up the BBS

```
#FLOORAREA
```

- Floor related files

```
#AUDITAREA
```

- User, Door, and File activity

```
#HOLDAREA
```

- Hold area for user messages

---

```
#EDIT-AREA
    - Editor area for a sysop editor(console only)

#NETAREA
    - Network files area

#NET_RECEPT_AREA
    - Receiving area for files sent to you

#DOMAINAREA
    - Domain data files area
```

The

```
CONFIG
```

program will require that you define each area and will create the directory if it does not exist.

## 1.41 #HELPAREA

This parameter specifies where all of your Help files will be located. These files are \*.HLP, \*.BLB, and \*.MNU. Normally, you should create this directory and place the help files in the directory before bringing up Citadel-86, since help files are usually online at all times.

```
#HELPAREA "cit:helps"
```

The help files, menus and default bulletins are in the cithelps.lha file in the Citadel Documentation room. You will have to do some customization of these files for your system. If you find an error or re-write the contents of a file, try to return that file so that others will benefit from your work.

## 1.42 #LOGAREA

This parameter specifies the location of your CTDLOG. ←  
 SYS  
 file (this file is sized by your  
 #LOGSIZE  
 parameter).

```
#LOGAREA "cit:users"      -- put it in a general system dir
```

## 1.43 #MSGAREA

This parameter specifies the location of CTDLMMSG.SYS. It is also the location of the special Citadel message file CIT\_MESSAGES.SYS. Citadel will create the message file when you run CONFIG, the other file is supplied with the executable archive.

```
#MSGAREA "cit:messages"           -- give msgs there own place
in the sun
```

## 1.44 #MSG2AREA

This parameter specifies the location of a second CTDLMMSG.SYS. Citadel will create the message file when you run CONFIG. Before starting up the BBS, you will need to copy CTDLMMSG.SYS into this area if you have the #MIRRORMSG statement in the CTDLCNFG.SYS file.

```
#MSG2AREA "cit:messages"         -- give msgs there own
place in the sun
```

## 1.45 #FLOORARE

This parameter specifies the location of CTDLFLR.SYS.

```
#FLOORAREA "cit:floors"
```

## 1.46 #AUDITAREA

This parameter is a string value parameter specifying a directory which will hold the audit files. If this parameter is not present in your CTDLCNFG.SYS file, then the audit files will not be created or updated by Citadel.

The audit files are usually text files of information on how

```

the BBS is running. For example there is a file (
    CALLLOG.SYS
)
which shows information on the callers. Another file keeps track of
door usage (
    DOORUSE.SYS
), and another one the file up/download
information (
    FILELOG.SYS
).

#AUDITAREA "c:audit"          -- This can only be a subdirectory

```

## 1.47 #HOMEAREA

This parameter defines the base directory the BBS will use for its operation.

This is the directory that the BBS will operate out of. In the examples,

this directory is assigned to the logical CIT: to make things simpler.

## 1.48 CIT\_MESSAGES.SYS

This file contains most of the Citadel BBS messages. The BBS references the text via the Message code. This allows the SYSOP the maximum flexibility in configuring their BBS. You can use the standard messages, or customize them to your heart's content.

The Message file is formatted into one line per message. The first 8 columns may be a "#" for a comment line, or a message code. THE "#" in column 1 will cause the rest of the line to be ignored. Column 9 is blank, for readability, and columns 10 to 79 are the message text. If the message text starts with an "@", the message text is taken to be a filename and that file will be read instead as the message text. This will allow you to have more than one line in a single message. The message codes end in either EX for expert user messages, or NO for novice user message. If no EX version exists, the BBS will automatically use the NO version. If neither one exists, the BBS will display "\*\*\*ERROR CODE nnnnnnnn" where nnnnnnnn is the missing message. If these occur, just create the appropriate message and add it to the file. If you find any message codes in the original file missing, then notify the Amiga Zone.

One of the reasons for the message formatting is to get system dependant information from the BBS by using special variable names. These names are listed below:

Variable	Description
^variant	Name of this Citadel Variant such as "Citadel 68K"
^version	Major Version Id of Citadel

```

^sysvers      Minor Version Id of Citadel
^baseroom     The baseroom of your BBS
^sysop        The name of the Sysop
^nodetitle    The BBS Node Title
^nodename     The BBS Node Name
^nodedomain   The Domain the BBS is considered part of
^nodeid       The BBS Node Id
^mainfloor    The Floor that contains the BaseRoom
^curruser     The name of the Current User.
^ulprotocols  A list of the Protocols usable for uploading
^dlprotocols  A list of the Protocols usable for downloading
^doorlist     A list of the Doors available in the current room
^lastuser     The last caller's name
^privileges   A list of the privileges you currently have.
^callcount    The number of calls this Citadel has recieved.
^ial          Special Integer Argument #1 (see below)
^sa1          Special String Argument #1
^ia2          Special Integer Argument #2
^sa2          Special String Argument #2
^ia3          Special Integer Argument #3
^sa3          Special String Argument #3
^currtime     The current time
^currdate     the current date
^s            A single space
^n            A newline followed by a space

```

The Special Arguments are pieces of data that are used in some of the existing messages. Currently the 3rd one is not used (but may be). Most of the messages do not use them, but those that do should probably continue to use them. You can remove the special variable from the messages that currently do use them, but adding them to a message that does not will get you a zero for an interger argument and nothing for a string argument.

It is best to keep the original message file around to check to see what was available for the code.

## 1.49 CALLLOG.SYS

CALLLOG.SYS contains three types of notes. The first ← type lists when the system has come up and down.

The second type records who has called, listing login and logout times, one line per person, in the following format:

```
<person> : <login time> - <logout time> <baud rate>
```

Occasionally such a line will have an extra character appended onto it, and they have the following significance.

```
'+' The user logged in as new.
```

```
'-' The user used .TS to logout.
```

'T' The user timed out on the system.

'E' The user hit the error limit on the system and was kicked off.

'B' The system kicked the user off for too many offenses against  
BADWORDS.SYS  
.

'C' The user tried to chat with you.

The third type of message in CALLLOG.SYS are notes regarding network sessions, both normal and anytime-net. These record on a single line the start and end times of the net sessions. This particular message can be disabled by using the #CLEAN-CALLLOG parameter.

## 1.50 FILELOG.SYS

FILELOG.SYS format is somewhat different. Generically, it looks like this:

```
<user name> @ <baud rate>

file1 (n bytes) <roomname> <U or D> <start to end> <length>
<protocol>

[FAILED]

file2 (n bytes) <roomname> <U or D> <start to end> <length>
<protocol>

[FAILED]
```

This format keeps the number of user names down. "n bytes" is the size of the file. "roomname" is the room involved. "U or D" refers to whether the named file was Uploaded or Downloaded. "start to end" refers to start time and end time of the file session, while length is the amount of time involved. Protocol will be one of the three XMODEM, YMODEM, or WXMODEM, or an external one you have setup. "FAILED" will only appear on the line if the transfer failed.

## 1.51 DOORUSE.SYS

DOORUSE.SYS simply lists who used what doors for what amount of time at what time of the day. It appears in the #AUDITAREA  
.

## 1.52 #HOLDAREA

Citadel has an optional capability to save a user's messages, put them on hold so to speak. This can be because the user lost carrier while entering a message, or told the BBS to Hold the message for later. The reason this is optional, is that if you do not specify this area, a user will not be able to use this option and any message held will be lost when the user terminates the session. A held file takes about 8K bytes of space on the disk. It is possible that every user could have a held message at one time, each is uniquely identified so in figuring disk space, this should be remembered.

```
#HOLDAREA "hold"
```

## 1.53 #EDIT-AREA

The optional edit area goes along with the sysop ←  
editor  
directive

```
#EDITOR  
which is used to define a directory where the  
BBS will put the temporary message file and run the sysop  
editor(this is for the console user only). This is like any BBS  
area.
```

```
#EDIT-AREA "RAM:"
```

## 1.54 #EDITOR

This is the command that is used to run the optional Console user editor. When a user is logged into the console, this command is used to invoke the external program to edit the message text(will be written to tempmsg.sys in this area). This command should specify any options needed to make the editor run and have the BBS pause while the editor is running(some editors will release the task as soon as they startup which will make Citadel think your done editing).

```
#EDIT-AREA "TTX WAIT"
```

## 1.55 #NETAREA

This command tells the BBS where to put the files that are related to the network process. It is like any other BBS area.

```
#NETEAREA "NET"
```

---

## 1.56 #NET\_RECEPT\_AREA

This is a special BBS directory that is used to store all files sent to your system by another system during a network session. When a system uses the Send File faculty(not the same as requesting a file over the network).

```
#NET_RECEPT_AREA "Recieving"
```

Files sent to your BBS using the utility AFF will appear in this area. In addition, the parameters

```
#NET_AREA_SIZE  
and
```

```
#MAX_NET_FILE
```

will be used to limit the amount of files and the largest file in this area.

## 1.57 #NET\_AREA\_SIZE

This parameter controls the total amount of space you wish to allow files coming into your system via the net(Send File Command). This is the limit on files being sent to your without you asking. If this area fills up to this size, additional files will be rejected.

## 1.58 #MAX\_NET\_FILE

This parameter controls the size of the largest file your will allow to be sent to you during a network session. Files larger than this size will be rejected.

## 1.59 #DOMAINAREA

This parameter specifies the area where Citadel will put the domain related temporary files. The files in this area are dynamic. Citadel will create them as needed and maintain them totally. You will not need to worry about these files unless there is a problem with domain mail and you are the server for your domain. This is a fairly advance topic and not covered in this document. Eventually, there will be a separate document for these types of issues.

## 1.60 basic system parameters

---

The basic system parameters define how many rooms (

```

    #MAXROOMS
    ), messages per room(#MSG-SLOTS), private mail
messages per user(
    #MAIL-SLOTS
    ), Size of the message
base(
    #MESSAGEK
    ) and if you will want it encrypted (
    #CRYPTSEED
    ).

```

You want to give some careful thought to these parameters since any choices made now will be a bit painful to modify later. There are utilities that will allow parameters to be modified, but only to increase them. To decrease them requires that you start over by deleting the appropriate files and reconfiguring.

I recommend that you keep

```

    #CRYPTSEED
    at zero although any
value can be used. It makes debug easier for me if I grab your
files plus that value will speed up all the processing. The message
slots and size of the message base is a little cryptic. If you have
100 slots, then Citadel will remember the last 100 messages in each
room. Mail has a separate value, but it is the same idea. With 100
rooms, you have 10,000 active messages possible in the system. With
messages sizing from 500 bytes to 7500 bytes, you could have a
message base of 5,000,000 to 75,000,000! Typically the average
message is alot closer to the 500 bytes size. The 600K size here
gives me a file that is 1217 blocks in size(614400 bytes). This
would actually fit on a floppy if you wanted(although it would
pretty much fill the floppy). You can make these pretty much any
value you want, the larger the value the more the disk space
needed. A reasonable approximation for messagek is:

```

```

( MSG-SLOTS + MAIL-SLOTS ) * 2.75K

```

```

( 120 + 99 ) * 3K = 602.25

```

You can use more..... For the larger sized system, use 7.5K and get 1604K... The practical limit is 4095K

## 1.61 #CRYPTSEED

CRYPTSEED is a value used in encrypting the data files. Choose a value when you install the system, but not thereafter -- or you won't be able to read the existing files any more. If you use a value of zero, none of the data files will be encrypted. This has little value since you as SYSOP can access anybody's account and read any message, there is no privacy. I recommend using zero. You should not allow any of the system files to be downloaded and this

is the only protection you have if you do. It is better to keep the users out of the data files. Using zero has an additional benefit that your system will be slightly faster. If you use a non zero encryption seed, all the data files will be encoded. An example is:

```
#CRYPTSEED 1234
```

It is important that you do not change this value. If for some reason you should lose your

```
CTDLCNFG.SYS
```

```
file, run the VERIFY
```

utility and it will display not only this value, but the value of all the important data from this file. Without this data item, you will not be able to reconfigure your BBS. This is important since if the bbs should crash, or your system should go down while the bbs is running, you have to run the CONFIG utility to recreate the data file CTDLTABL.SYS. Without that file, the BBS will not run. There is only one parameter on the command line. If it does not match "onlyParams" or "FirstInit" then CONFIG will assume you are re-initializing the BBS. "FirstInit" assumes that you want to create the BBS from scratch initializing all the files as if creating a new BBS. This means that if you already have a BBS up and running, all the data files will be re-created and initialized as empty(i.e all existing users deleted, all messages gone). You can use this the first time and CONFIG will not ask you any questions about creating this file or that one... Once you have a running BBS and you need to modify certain parameters(see 'Safe Configuration Parameters')

## 1.62 Safe Configuration Parameters

These parameters control characteristics of the BBS and ↔ not file sizes. You can modify these at any time by changing the value in the

```
CTDLCNFG.SYS
```

```
file and then running "CONFIG ONLYPARAMS". To
```

do this, change the file, bring the BBS down, then run CONFIG and then restart the BBS.

## 1.63 #NODEID

As mentioned, this parameter is a network parameter that ↔ has traditionally always been set, even for non-network Citadels. If you have no plans to ever be on a C86Net, Then this is not real important.

If you do plan to join the

C86Net  
 , (we'll go over this in more detail in the section on Networking), then you do have to set this parameter correctly. The format of this parameter is

```
"<Country code><Area Code><Phone number>"
```

all of which applies to the phone your system resides on. Country code is a two letter sequence indicating what country you live in (US is the United States, CA is Canada. Other country codes may be found in COUNTRY.DOC). Area code is the area code of your system (yes, we are aware there is a clear bias towards US-style telephony). And Phone number is, of course, the phone number your system is on. You can put punctuation (such as parenthesis and dashes), but please be conservative with them. This string value does not obey formatting directives. Here's a fairly generic example:

```
#NODEID "US (609) 953 8159" -- Some system somewhere..:)
```

Other systems will use your node id to call you for networking. It will be how other systems identify your system's messages.

## 1.64 #NODENAME

nodeName is, in reality, purely a network parameter, and if you have no plans to ever join a C86Net, then there is no need to fill in this parameter. However, it has always been traditional, even before there was a net for any Citadel system anywhere, to fill in this and the

```
#NODEID
```

parameter. nodeName is a string value which does NOT accept formatting directives (i.e., formatting directives will be ignored). It can be no longer than 19 letters long. It should be a short, mnemonic name for your system. An EXAMPLE of a reasonable value:

```
#NODENAME "ODD-DATA" -- The original Citadel
```

If you ever do join a C86Net, messages from your system appearing on another Citadel-86 node will look something like this

```
82Nov23 from Cynbe ru Taren @ODD-DATA
```

except ODD-DATA would be replaced with your value for #NODENAME.

---

## 1.65 #NODETITLE

The first parameter you should find is called `nodeTitle`. It is a string value obeying formatting directives, and is subject to formatting considerations. `nodeTitle` is the title of your installation printed when carrier is detected on your system. More precisely, `nodeTitle` will show up in the following place on your system:

```
Welcome to <#NODETITLE>
```

However, `nodeTitle` may not necessarily be printed at this point. After successfully bringing your system up, please consult the section on Help Files for more information on banner options.  
EXAMPLE:

```
#NODETITLE "Test System\n Truly a Heaven in Reverse" The  
#NODETITLE is printed out on .Read Status commands, also. There is  
no formal limit on the length of this parameter.
```

## 1.66 banners

## 1.67 The Amiga Zone

The Amiga Zone is the primary support BBS. The number is (609) 953-8159. There are other Citadels that will help the budding Sysop out, but this is the place you will find the latest and greatest version of Citadel, CONFIG, and the Utilities. In addition to calling direct, you should think about networking the Citadel 68K room. This is the place where comments, bug reports, and other issues are discussed. The Amiga Zone will feed the room to any Citadel that wishes to network, however, the Amiga Zone will not call out for a network session unless the system is local. You will have to pay for the calls. This does not amount to much if you call a few times a week. Fortunately, there are about 200 Citadels in the USA and Canada, you might find a local system to network with, or one that costs less than the Amiga Zone to network with. If you wish, I will answer questions at my internet address "apreston@isd.csc.com" or "tony-preston@cup.portal.com".

## 1.68 #LOGSIZE

This numerical parameter gives you the ability to decide how many accounts will be available on your system. If you run a system in which more accounts are used than there are accounts reserved, then new accounts are generated by killing old accounts. Accounts are recycled by finding the account who's last use is the oldest of all the accounts in the system, under the assumption such an account is no longer active.

All space is reserved immediately for these accounts. The size of this file can be estimated from the formula (this includes a possible held file for every USER).

```
# of bytes = LOGSIZE * (82 + MAXROOMS + (6 * MAIL-SLOTS) + 8092)
```

so if you are operating in a restricted environment, plan accordingly. If you need to, you can expand the size of the log through the use of the DATACHNG utility, but the log cannot be shrunk. This is a numerical value. Here is an example:

```
#LOGSIZE 200

For a system with 100 rooms(
    #MAXROOMS
    ), and 100 mail
slots(
    #MAIL-SLOTS
    ) this would be just over 150K bytes for 200
users. It should be noted that the larger the logsize, the longer
the
    CONFIG
    utility will take to re-configure the system. Each
entry is checked and updated when this is done.
```

## 1.69 #MAXROOMS

This numerical parameter specifies the maximum number of rooms your system will support. Since the baseRoom, Aide>, and Mail> room are necessary, the smallest value you can give is 3. The largest number is 65536. If you wanted to have a 64 room system, you'd have

```
#MAXROOMS 64
```

You can use the following formula to estimate the number of bytes a room file will take up on your SYSTEM:

```
# of bytes = MAXROOMS * (50 + (6 * MSG-SLOTS))
```

For a system with 100 rooms and 200 message slots (#MSG-SLOTS), you would need approximately 125 Kbytes of disk space. It should be noted that the larger this is, the longer the

```
CONFIG
    takes since each room is updated.
```

## 1.70 #MAIL-SLOTS

This is a numerical parameter specifying how many messages per

log record you wish to reserve for Mail. The Mail> room is the only room in the system whose data is not kept in CTDLROOM.SYS. Instead, the file CTDLLOG.SYS contains the "Mail>" room, reserving for each account enough room for MAIL-SLOTS Mail messages. Therefore, this parameter gives you the ability to decide the maximum number of Mail> messages per user they can access. Please remember if a user gets more messages in Mail> than there are MAIL-SLOTS between two successive logins, then they will lose the earlier messages sent to them. Another consideration is many users like to review old Mail when engaged in an in-depth private conversation. Therefore, setting MAIL-SLOTS to a low value may not be the attractive alternative it first seems. However, it should go without saying a high MAIL-SLOTS value may eat up more room than necessary on your drives. The section on

```
#LOGSIZE
```

will give an exact formula for how much space your log will take up.

## 1.71 #MESSAGEK

MESSAGEK defines how much disk space you wish to allocate for messages on your installation. Because messages can vary in size, there is no way to define how many messages you can have in your system, or how fast they turnover. All the messages in your system will reside in CTDLMSG.SYS, and thus the number of messages in your system at any given moment will depend totally on the length of the messages being entered into the system by your users. The turnover rate of your messages will depend on how busy your system is.

For example, if you reserve 600K for messages, you would have an approximate 1200 messages(messages can get as large a 7500 characters so if you have verbose users, this could be as low as 80 messages if they were all to the limit, a good conservative estimate is 512 characters which gives 1200 messages). If you get 25 callers a day and each posted 4 messages, you would have a turnover of about 12 days. If you networking and get 25 messages a day in 4 rooms, plus these callers, you have a 6 day turnover. The higher the volume, the quicker the turnover. When the messages turnover, older message space gets reused which means older messages are deleted. Shared rooms can have a very high volume.

The sysop of an installation should also keep in mind that very large systems, with many new messages, can be intimidating to new users, so large message spaces should be approached with caution. Remember, there is a utility(Expand) for expanding the message base, but not for shrinking it. The only method available to shrink the message base is to delete the existing one and then reset this value to a smaller amount. You will lose all the messages(including mail) if you do this.

This is a numerical value which you specify in 'K', which is actually 1024 bytes/K. So, for example, to specify a 250K message file

```
#MESSAGEK 250          -- 250K message base
```

The above parameter will require 250 Kbytes of disk space.

## 1.72 #SCAN-NET-MESSAGES

This parameter tells Citadel that the messages recieved ←  
over  
the network should be scanned against the file  
BADWORDS.SYS  
and  
any matches should cause the offending message to be discarded.

## 1.73 Utilities

## 1.74 Installation

## 1.75 C86Net

## 1.76 BBS List

## 1.77 Citadel

## 1.78 Files

This section details the various files that exist in ←  
the  
Citadel BBS system. Most of these files are maintained by the BBS  
software and you only need to know a general idea of why they are  
there and how big they will be. Some have particular formats and  
must be maintained by the Sysop. The files are:

debug.sys  
- System debug information

crash.sys  
- System failure message

debug.sys

crash.sys

## 1.79 debug.sys

---

This file is only generated if the #AuditArea is defined. It will be generated only if the debug sysop option is turned on or there is a serious error or problem and the system needs to report information. Most of the entries in this file are also displayed on the console. This is a log that should be examined for problems that could occur in your setup. Generally, if you have a problem and want someone to assist you, it would be a good idea to make this file available (in other words don't delete until your sure it wont be needed).

## 1.80 crash.sys

This file usually contains only a single error message. ←  
It is used to display information about a failure while the BBS was initializing and did not have the screen and windows open to report the problem. This file will occur in the current directory which might not be

```
#HOMEAREA
, since the BBS is going to stop itself
immediately.
```

## 1.81 #ROOMAREA

This parameter specifies the location many files.

```
#ROOMAREA "cit:room"      -- another general system dir
```

This directory contains many files which are very important to the basic operation of Citadel. This may seem overwhelming at first, and you need to know what these files are to understand how to fix problems that might occur later. Much of these files relate to the options you select on your Citadel with

```
CONFIG
```

```
. These
```

affect networking, user account creation, what external programs you can run and many other Citadel options. For the most part, you can start up a BBS without knowing anything about these files, but eventually if you run into problems, these items are a major help with most of them.

```
aliases.sys
```

```
badnames.sys
```

badpasswords.sys

badpeople.sys

badwords.sys

citadoor.sys

ctdlarch.sys

ctlddir.sys

ctdlfwd.sys

ctdlinfo.sys

ctdlmodr.sys

ctdlprot.sys

ctdlroom.sys

DExxx.SYS

results.sys

Some of these files are maintained by Citadel itself and you need not do anything with the files at all. The only reason they are mentioned here is to prevent confusion and to document their ultimate purpose in the life of your BBS.

ALIASES.SYS

badnames.sys

badpasswords.sys

badpeople.sys

badwords.sys

citadoor.sys

ctdlarch.s

ctlddir.sys

ctdlfwd.sys

ctdlinfo.sys

ctdlmodr.sys

ctdlprot.sys

protocols

ctdlroom.sys

```
DExxx.SYS

results.sys

DISCARD

CTDLCNFG.SYS
```

## 1.82 ALIASES.SYS

This file is used to alias the name of a BBS with another name. This serves the purpose of clarifying what a user thinks is the name of a BBS. For example, in the typical discussions on BBS issues, people refer to "C-86 Test System" using "Test System". This is common enough that a User might try to send mail to "Sysop@Test System" only to find that the BBS does not exist. When you have two names that seem equally applicable for some system, you can make an entry in the ALIASES.SYS file. The format is one per line and is:

```
<alias> <realname>
```

The <alias> and <realname> are quoted strings so "Amiga Zone" and "The Amiga Zone" would be good entries for an alias and realname. The two are separated by a single space.

## 1.83 badnames.sys

This file is optional. The Sysop may create it if desired. The format is very simple. One name per line. Each name in the file will be checked against any new account name and the name will be rejected if a match occurs. This file is a list of invalid user names. If it is not present, Citadel will not complain and will accept any new name.

## 1.84 badpasswords.sys

This file is optional. The Sysop may create it if it is desired that the BBS should check each password to ensure that commonly used names and easily guessable passwords are not used. Each Password entered by users will be validated against this list and a match will be rejected.

## 1.85 badpeople.sys

---

This file is optional. The Sysop may create it if desired.

The format is a username and a room name separated by a comma. If this file exists, each network message will be checked against the list and any matches will cause the message to be discarded(see

badwords.sys

for a similar censor mechanism). It is important to note here that these messages are \*REMOVED\* from the network and not sent on to other systems that may not want them removed. At times, when a certain user gets out of control, a Sysop may want to use this option.

## 1.86 badwords.sys

This optional file may be created by the Sysop to control the contents of messages. Each message may be optionally scanned (if

```
#SCAN-NET-MESSAGES
  is in the
CTDLCNFG.SYS
  file) as it arrives
```

during a network session. Any messages which fail to meet your standards of decency will be discarded(placed in the file that is called

```
DISCARD
  ) and a message left for you in the AIDE room.
```

Usually, there is little need to actively censor Citadel Users. The format of the file is simply a list of words or partial words(frog in the list will reject froggy, froggie, and frog). The list of words starts on the 3rd line of the file and all lines from there to the end of the file. The first line is called the "icky" level. This level indicates how many times a user may use one of the "forbidden" words before the system will disconnect them. The second line may be blank if you dont want the rejected messages saved. If non-blank, It will be the name of the file that Citadel uses to save the text. Any user kicked off the BBS will get a "B" added to the CALLLOG.SYS entry.

## 1.87 citadoor.sys

This file is created by the CONFG

program. It contains the data needed by the BBS to run any door programs you have setup.

## 1.88 ctdlarch.s

This file is maintained by the BBS. It contains entries for rooms that are archived. You should not mess with this file, instead use the BBS to change how and when a room is archived. Room archiving is just an additional copy of all messages that appear in a room. The archive file may have optional formatting parameters in the name. %m will be replaced by the current month and %y by the last two digits of the year.

## 1.89 ctldir.sys

This file is maintained by the BBS. It contains entries that tell the BBS the name of the directory that is attached to the room. You should use the AIDE commands with the BBS to make any changes needed to this file.

## 1.90 ctldfwd.sys

This file is maintained by the BBS. It contains entries that tell the BBS where to forward mail to a particular user. This data is maintained by the individual user, you do not need to worry about it.

## 1.91 ctldinfo.sys

This file is maintained by the BBS. It contains entries for each room that are the text information on that room. You should use the BBS to change any room information and not directly in this file.

## 1.92 ctldmodr.sys

This file is maintained by the BBS. It contains entries to tell the BBS who is a moderator of a particular room. You should use the BBS to change any of this information.

## 1.93 ctldprot.sys

This file contains the commands needed to implement external protocols. The BBS will read this file only when it starts up. Each line in the file contains information about either an upload or download protocol. The BBS always has X and Y modem (even if they are really slow implementations of it) internally. There are two types of entries. The first is the "regular" external program entry that defines how to call on a program that will implement the

---

protocol. The protocol parameters are used with this type of protocol. Citadel will invoke the program and expect the program to take care of everything except the description(which will be prompted for afterwards). The second type is an XPR implementation. Both of these types have the same parameters for the first four parameter, it is the fifth parameter that varies. The format is:

```
<letter> <type> <name> <direction> <fifth parameter>
```

The <letter> is the protocol letter that will be used by the BBS when a user enters .R<letter>B for example. Most people use Z for Zmodem for example.

The <type> is 1 or M for "regular" external protocols. 1 means only single file transfers, M means batch transfers are supported. It is suggested that even for a protocol like Zmodem, you only allow uploads to be single files. This will prevent files from getting uploaded without descriptions. A <type> of X or Y is the corresponding types for the XPR type. X is the single and Y is the batch.

The <name> parameter is the name the BBS will display when you type the <letter>

The <direction> is U for upload and D for download.

The <fifth parameter> is an XPR library name if it is an XPR protocol. It should be spelled exactly like the name in the LIBS: directory. If it is not an XPR protocol, the rest of the line is the command used by the BBS to invoke the protocol. An example CTDLPROT.SYS file looks like:

```
Z X Zmodem U xprzmodem.library
```

```
Z Y Zmodem D xprzmodem.library
```

```
Q M Zmodem U xprd -mcit:xprd.log -s -c -n -q r %g
```

```
K M Kermit U xprd -mcit:xprd.log -s -c -n -q
-lxprkermit.library r %g
```

This would only allow downloading with an XPR Zmodem, but allow uploading with two types of Zmodem and a Kermit.

## 1.94 protocols

When you have an external protocol, the command may get rather complex. The BBS must insert the filename(s) on the command line. Citadel will scan the command and locate a "%g", if that is not found the end of the command line is used instead and the filename(s) being transferred will be inserted there.

## 1.95 ctdlroom.sys

This file is maintained by Citadel. You should not mess with it. It contains all the information needed to maintain a room. Use the utilities and Citadel to make any appropriate changes.

## 1.96 DExxx.SYS

These files define external commands that Citadel may use. There are three lines in the file, each defining what Citadel does to Test, Uncompress, and Compress files using the "xxx" archiver. The supported types are ARC, LHA, LZH, and ZIP. Line 1 is the test line, this is used when a user uploads a file of the recognized types. Citadel will test the archive to ensure a good upload. Line 2 is the Uncompress line, Citadel uses this line to allow the

## 1.97 results.sys

This file defines all the results codes your modem returns. ←  
Citadel needs this file to determine the speed, even if the modem is locked to one speed(see #LOCK-PORT). Citadel will use this to properly compute the estimated times for file downloads and for the speed of the modem. The codes are one per line and a sample file would look like:

```
#RESULT-300 CONNECT 300

#RESULT-1200 CONNECT 1200

#RESULT-2400 CONNECT 2400

#RESULT-2400 CONNECT 2400/ARQ

#RESULT-4800 CONNECT 4800

#RESULT-4800 CONNECT 7200

#RESULT-9600 CONNECT 9600

#RESULT-9600 CONNECT 12000

#RESULT-14400 CONNECT 14400

#RESULT-14400 CONNECT 38400

#RESULT-14400 CONNECT 57600

#NO-DIALTONE NO DIALTONE
```

```
#NO-DIALTONE NO CONNECT
```

```
#NO-CARRIER NO CARRIER
```

```
#OK OK
```

```
#NO-CARRIER ERROR
```

```
#NO-CARRIER VOICE
```

```
#BUSY BUSY
```

```
#RING RING
```

The format is `<code> <modem result code>`, the two parameters are separated by a space. Every possible result is not defined so this example has multiple uses of the same `<code>` for different connect speeds.

## 1.98 DISCARD

This file is the default file that will be used for ↔ messages that are duplicates, rejected because of decency( BADWORDS.SYS or BADPEOPLE.SYS ). Citadel saves the discards here so you can review them(just incase there is problem).

## 1.99 CTDLNCFG.SYS

This file is the basic configuration information for ↔ setting up the BBS. The text lines in this file are processed by CONFIG and the CTDLTABL.SYS file is created. This is the file you should edit to make adjustments to the BBS.