

カシオハンディターミナル

# DT-900 FTPユーティリティ 解説書

Rev 1.00

カシオ計算機株式会社

# 目次

1 概要 .....	2
2 ファイル転送プロトコル .....	3
2 - 1 概要 .....	3
2 - 2 提供ファイル .....	3
2 - 3 FTP対応コマンド .....	3
2 - 4 関数の機能 .....	4
2 - 4 - 1 FTPマクロ終了コード .....	4
2 - 4 - 2 中断処理 .....	4
2 - 4 - 3 初期化処理と終了処理 .....	4
2 - 4 - 4 コマンドファイル実行機能 .....	4
2 - 4 - 5 FTPコマンド .....	4
2 - 4 - 6 データタイプ .....	4
2 - 4 - 7 関数詳細 .....	5
2 - 5 FTPコマンド .....	14
2 - 5 - 1 概要 .....	14
2 - 5 - 2 提供ファイル .....	14
2 - 5 - 3 書式 .....	14
2 - 5 - 4 戻り値 .....	14
2 - 5 - 5 コマンドスクリプトファイル .....	14
2 - 5 - 6 デバッグモードとVERBOSEモード .....	15
2 - 6 終了コード .....	16
2 - 6 - 1 ログファイル .....	16
2 - 6 - 2 クライアント側終了コードとメッセージ .....	16
2 - 6 - 3 サーバー側終了コードと意味 .....	17
2 - 6 - 4 ログファイル出力例 .....	18
3 サイトサーベイ関数 .....	19

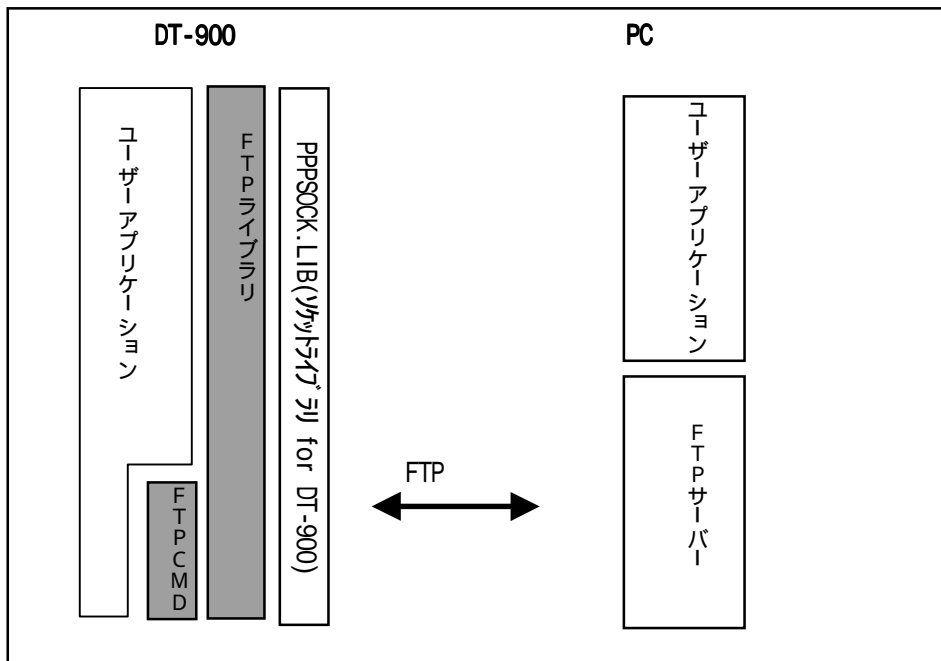
# 1 概要

ハンディターミナル『DT-900』PHS一体型モデルにおけるファイル転送プロトコル(FTP)ユーティリティとして、下記の機能を提供します。

## (1) ファイル転送機能(FTP)

PPPSOCK(ソケットライブラリ for CASIO DT-900)を使用し、FTP コマンドを解析/生成するクライアント側ライブラリを提供します。

本ライブラリを使用したFTPクライアントとして、FTPCMD関数があります。  
アプリケーションプログラムから簡単に実行する場合は、本関数を利用すると便利です。



## 2 ファイル転送プロトコル

### 2 - 1 概要

DT-900 において、ファイル転送プロトコル(FTP:File Transfer Protocol)をライブラリとして提供します。アプリケーションから FTP 機能呼び出す場合は、FTP コマンド単位に定義されたマクロを使用します。使用できる FTP コマンドは、DT-900 での利用に必要なコマンドのみです。

### 2 - 2 提供ファイル

・ `ftp.h` : マクロ定義等のヘッダーファイル  
・ `ftplib.lib` : ライブラリファイル

### 2 - 3 FTP 対応コマンド

DT-900 において実装するコマンドは、以下の通りです。

No	FTP コマンド	マクロ名	機能
1	<code>append</code>	FTP_APPEND	ローカルファイルをリモートコンピュータにコピーします。 (リモートファイルが既存の場合追加書き込み、存在しなければ新規作成を行う)
2	<code>ascii</code>	FTP_ASCII	ファイル転送の種類を ASCII に設定します。(初期値: ASCII)
3	<code>binary</code>	FTP_BINARY	ファイル転送の種類を BINARY に設定します。(初期値: ASCII)
4	<code>bye</code>	FTP_BYE	リモートコンピュータとの FTP セッションを終了して ftp を終了します。
5	<code>debug</code>	FTP_DEBUG	ログファイルにクライアント出力パケットを記録します。 FTP_VERBOSE により、ログファイル出力指定されている必要があります。 (初期値: オフ)
6	<code>get</code>	FTP_GET	リモート ファイルをローカル コンピュータにコピーします。
7	<code>mget</code>	FTP_MGET	複数のリモートファイルをローカルコンピュータにコピーします。 ワイルドカードによるファイル指定が可能。... 注1)
8	<code>mput</code>	FTP_MPUT	複数のローカルファイルをリモートコンピュータにコピーします。 ワイルドカードによるファイル指定のみ対応。... 注1)
9	<code>open</code>	FTP_OPEN	指定した FTP サーバーに接続します。
10	<code>put</code>	FTP_PUT	ローカルファイルをリモートコンピュータにコピーします。
11	<code>quit</code>	FTP_QUIT	リモート コンピュータとの FTP セッションを終了して ftp を終了します。 (bye と同じ)
12	<code>recv</code>	FTP_RECV	リモートファイルをローカルコンピュータにコピーします。(get と同じ)
13	<code>rename</code>	FTP_RENAME	リモートファイルの名前を変更します。
14	<code>send</code>	FTP_SEND	ローカルファイルをリモートコンピュータにコピーします。(put と同じ)
15	<code>user</code>	FTP_USER	リモートコンピュータに対してユーザーを指定します。
16	<code>verbose</code>	FTP_VERBOSE	ログファイルを作成します。 ファイル転送が完了すると、転送効率に関する統計情報も出力されます。 (初期値: オフ)

注1) ワイルドカードによるファイル指定のみ可能です。(可変引数でのファイル指定は不可)

注意事項1) FTP マクロ使用前後には、初期化処理、終了処理を行う必要があります。

注意事項2) FTP マクロでは、ファイル転送等を行う場合、open, user コマンドを実行する必要があります。

## 2 - 4 関数の機能

### 2 - 4 - 1 FTPマクロ終了コード

FTP マクロでは、基本的に、正常 / 異常終了のみを終了コードとし、詳細はログファイルに記録します。  
(但し、FTP\_VERBOSE によるログファイル指定時のみ )

FTP マクロでエラーが発生した場合の詳細情報取得用関数としては、下記の関数が用意されています。

```
int FTP_GETLASTERROR( void )
```

### 2 - 4 - 2 中断処理

FTP ファイル転送関連マクロ (FTP\_GET/FTP\_PUT/FTP\_MGET/FTP\_MPUT)では、**[機能]** + **[取消]** 及び、**[拡張]** + **C** を押下することにより、ファイル転送を中断します。  
アプリケーションプログラムでキーの機能を変更した場合は、中断できないことがあります。

### 2 - 4 - 3 初期化処理と終了処理

FTP マクロ内では初期化処理と終了処理を行わないため、FTP マクロ使用前に、アプリケーション側にて処理を行う必要があります。

初期化、終了処理用関数としては、以下の関数を使用します。

```
BOOL FTP_START( void )  
BOOL FTP_END( void )
```

### 2 - 4 - 4 コマンドファイル実行機能

指定された形式のコマンドファイルにしたがって実行する関数として、以下の関数が用意されています。  
コマンドファイルを書き換えるだけで、異なった処理が実行できるプログラミングが可能になります。

```
BOOL FTP_FILE( commandfilename )
```

### 2 - 4 - 5 FTPコマンド

指定された形式のコマンドファイルにしたがって実行する機能として、以下の関数が用意されています。

コマンド仕様は、「2 - 5 FTPコマンド」を参照してください。

### 2 - 4 - 6 データタイプ

パラメータのデータタイプは、以下の通りです。

```
typedef int          BOOL;  
typedef char _far*  LPSTR;  
typedef const char _far* LPCSTR;
```

**BOOL** FTP\_APPEND( *localfilename*, *remotefilename* )

---

LPCSTR	<i>localfilename</i> ;	ローカルファイル名
LPCSTR	<i>remotefilename</i> ;	リモートファイル名

**概要**

現在のファイル転送の種類を使って、ローカルファイルをリモートコンピュータにコピーします。  
(リモートファイルが既存の場合は追加書き込み、存在しなければ新規作成を行う。)

**詳細**

*localfilename* を省略することはできません。  
*remotefilename* に NULL を指定することにより、*localfilename* と同じファイル名で転送することができます。  
中断キー (機能 + 取消) は使用できません。

パラメータ	説明
<i>localfilename</i>	ローカルファイル名 省略はできません。
<i>remotefilename</i>	追加コピーするリモートファイル名 NULL を指定するとローカルファイル名と同じファイル名で コピーされます。

**戻り値**

0 : コマンド正常終了  
-1 : コマンド失敗

**BOOL** FTP\_ASCII( *void* )**概要**

ファイル転送の種類を ASCII に設定します。

**詳細**

ファイルの転送モードはデフォルトで ASCII に設定されています。  
バイナリファイルを転送するためには FTP\_BINARY マクロを実行する必要があります。

パラメータ	説明
なし	

**戻り値**

0 : コマンド正常終了  
-1 : コマンド失敗

## *BOOL* FTP\_BINARY( *void* )

---

### 概 要

ファイルの転送の種類を BINARY に設定します。

### 詳 細

ファイルの転送モードはデフォルトで ASCII に設定されています。  
バイナリファイルを転送するためにはこのマクロを実行する必要があります。

パラメータ	説明
なし	

### 戻り値

- 0 : コマンド正常終了
- 1 : コマンド失敗

## *BOOL* FTP\_BYE( *void* )

---

### 概 要

リモートコンピュータとの FTP セッションを終了して、FTP を終了します。

### 詳 細

リモートコンピュータとの FTP セッションの終了を要求し、コントロールコネクションを切断します。

パラメータ	説明
なし	

### 戻り値

- 0 : コマンド正常終了
- 1 : コマンド失敗

## BOOL FTP\_DEBUG( mode )

---

int                    mode;                    デバッグモード

### 概要

ログファイルにクライアント出力パケットを記録します。

### 詳細

FTP\_VERBOSE によりログファイルが指定されている必要があります。

デフォルトではこのモードは FALSE に設定されています。

出力されるコード、フォーマット等は、「2 - 4 終了コード」を参照してください。

パラメータ	説明
<i>mode</i>	debug モードのスイッチ
	TRUE : 出力
	FALSE : 未出力

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗

## BOOL FTP\_GET( remotefilename, localfilename )

---

LPCSTR                *remotefilename*;                リモートファイル名  
LPCSTR                *localfilename*;                ローカルファイル名

### 概要

現在のファイル転送の種類を使って、リモートファイルをローカルコンピュータにコピーします。

### 詳細

*remotefilename* を省略することはできません。

*localfilename* に NULL を指定することにより、*remotefilename* と同じファイル名で転送することができます。

中断キー（機能 + 取消）が使用できます。

FTP\_MGET マクロはこのマクロを参照します。

パラメータ	説明
<i>remotefilename</i>	取得するリモートファイル名
<i>localfilename</i>	ローカルファイル名
名	NULL を指定することにより、 <i>remotefilename</i> と同じファイル名 でコピーされます。

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗



## BOOL FTP\_GETLASTERROR( void )

---

### 概要

最後に発生したエラーコード (FTP サーバより取得した応答コード) を返します。

### 詳細

各FTP マクロがエラー状態で終了したときに、このマクロを実行することでより詳細な情報を得ることができます。

パラメータ	説明
なし	

### 戻り値

最後に発生したエラーコードです。  
FTPサーバ側のエラーの場合は、FTPサーバ側から取得した応答コードです。  
( 2 - 4 終了コード 参照 )

## BOOL FTP\_MGET( remotefilename, localdrive )

---

LPCSTR	<i>remotefilename;</i>	リモートファイル名
LPCSTR	<i>localdrive;</i>	ローカルドライブ名

### 概要

現在のファイル転送の種類を使って、複数のリモートファイルをローカルコンピュータにコピーします。

### 詳細

ワイルドカードによるファイル指定が可能です。  
中断キー (機能 + 取消) が使用できます。  
FTP サーバ上に存在しないファイル (ワイルドカード可) を指定した場合、FTP サーバはファイル一覧のバッファにエラーメッセージを返し、FTP\_GET はそのメッセージをコピーしようとして失敗します。  
(ファイル名とエラーメッセージとの区別がつかないためです。)

パラメータ	説明
<i>remotefilename</i>	取得するリモートファイル名
<i>localdrive</i>	ファイルを格納するドライブ名

### 戻り値

0 : コマンド正常終了  
-1 : コマンド失敗

## BOOL FTP\_MPUT( *localfilename*, *localdrive* )

---

LPCSTR	<i>localfilename</i> ;	ローカルファイル名
LPCSTR	<i>localdrive</i> ;	ローカルドライブ名

### 概要

現在のファイル転送の種類を使って、複数のローカルファイルをリモートコンピュータにコピーします。

### 詳細

ワイルドカードによるファイル指定可能です。

中断キー（機能 + 取消）が使用できます。

パラメータ	説明
<i>localfilename</i>	転送を行うローカルファイル名
<i>localdrive</i>	ファイルの存在するドライブ名

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗

---

## BOOL FTP\_OPEN( *hostname* )

LPCSTR	<i>hostname</i> ;	ホスト名
--------	-------------------	------

### 概要

指定した FTP サーバに接続します。

### 詳細

ホスト名は IP アドレスで指定します。（例：FTP\_OPEN("20.65.234.1")）

指定した IP アドレスの FTP サーバとの接続に失敗したときのタイムアウトは数分間かかります。

パラメータ	説明
<i>hostname</i>	接続するホスト名の IP アドレス

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗

## *BOOL* FTP\_PUT( *localfilename*, *remotefilename* )

---

LPCSTR	<i>localfilename</i> ;	ローカルファイル名
LPCSTR	<i>remotefilename</i> ;	リモートファイル名

### 概要

現在のファイル転送の種類を使って、ローカルファイルをリモートコンピュータにコピーします。

### 詳細

*localfilename* を省略することはできません。

*remotefilename* に NULL を指定することにより、*localfilename* と同じファイル名で転送することができます。

中断キー（機能 + 取消）が使用できます。

FTP\_MPUT マクロはこのマクロを参照します。

パラメータ	説明
<i>localfilename</i>	ローカルファイル名
<i>remotefilename</i>	コピーするリモートファイル名 NULL を指定することにより、 <i>localfilename</i> と同じファイル名でコピーします。

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗

## *BOOL* FTP\_RENAME( *filename*, *renamefile* )

---

LPCSTR	<i>filename</i>	リモートファイル名
LPCSTR	<i>renamefile</i>	変更後のファイル名

### 概要

リモートファイルの名前を変更します。

### 詳細

名称変更により、リモートディレクトリ間移動が伴うような指定はできません。

*filename*、*renamefile* とともに省略できません。

パラメータ	説明
<i>filename</i>	名称変更を行うリモートファイル名
<i>renamefile</i>	変更後のファイル名

### 戻り値

0 : コマンド正常終了

-1 : コマンド失敗

## *BOOL FTP\_USER( username, password )*

---

LPCSTR	<i>username</i> ;	ユーザー名
LPCSTR	<i>password</i> ;	パスワード

### 概要

リモートコンピュータに対して、ユーザを指定します。

### 詳細

コントロールコネクションが接続されていれば、何度でもこのマクロを実行することができます。  
(途中から違うユーザ名にログインし直すことができます。)

パラメータ	説明
<i>username</i>	ユーザー名
<i>password</i>	パスワード

### 戻り値

0 : コマンド正常終了  
-1 : コマンド失敗

## *BOOL FTP\_VERBOSE( mode, logfile name )*

---

BOOL	<i>mode</i>	verbose のスイッチ
char *	<i>logfile name</i>	FTP 状態を記録するファイル名

### 概要

ログファイルのモード(on/off)とログファイル名を指定します。

### 詳細

デフォルト状態では、ログファイルを作成しません。(verbose は FALSE です。)  
指定ファイルにログを追加する(上書きしない)ので、ディスク容量に注意が必要です。  
出力されるコード、フォーマット等は、「2 - 4 終了コード」を参照してください。

パラメータ	説明
<i>mode</i>	verbose のスイッチ
	TRUE : 出力
	FALSE : 未出力
<i>logfile name</i>	FTP 状態を記録するファイル名

### 戻り値

0 : コマンド正常終了  
-1 : コマンド失敗

## BOOL FTP\_FILE( *commandfilename* )

LPCSTR

*commandfilename*;

FTP コマンドを記述したファイル名

### 概要

指定されたコマンドファイル内のコマンドを実行します。  
他の関数と同様に、初期化处理、終了処理は必要です。

### 詳細

1 行 128 文字未満で設定してください。  
各パラメータの間はスペースとタブを認識します。  
途中で空行があっても構いません。(読み飛ばします)  
コマンド名は、大文字、小文字どちらでも構いません。  
以下のコマンドは使用できません。(コマンドエラーになります)

getlasterror, file

パラメータ

説明

*commandfilename*

FTP コマンドを記述したファイル名

コマンドファイルは、各 FTP コマンドとそれに対応するマクロの引数を 1 行 1 コマンドで記述します。  
使用できるコマンドは「2 - 1 機能概要 (3) FTP 対応コマンド」を、引数は各関数の詳細を参照して下さい。  
実行結果のエラーは、個別には取得できません。

### 戻り値

- 0 : 全てのコマンドが正常終了
- 1 : 1 つ以上のコマンドが異常終了

コマンドファイルの例

```
verbose true logfile.txt
debug true

open serverIP
user taro passwd
binary
put hello.exe
get sakura.bmp
bye

verbose false
(eof)
```

## *BOOL* FTP\_START( *void* )

---

### **概要**

初期化処理を行います。  
処理の最初に必ず実行して下さい。

### **詳細**

各FTP マクロを実行する前に必ず実行して下さい。

パラメータ	説明
なし	

### **戻り値**

- 0 : コマンド正常終了
- 1 : 初期化失敗 (ドライバなし等)
- 2 : ポート初期化失敗
- 3 : 同期取り失敗 (マスターなし等)

## *void* FTP\_END( *void* )

---

### **概要**

終了処理を行います。  
FTP\_START 関数が異常終了した場合は、本関数を実行する必要はありません。

### **詳細**

FTP マクロ実行終了後に本関数を実行して下さい。

パラメータ	説明
なし	

### **戻り値**

なし

## 2 - 5 FTPコマンド

### 2 - 5 - 1 概要

FTP関数(FTP\_CMD)は、DT-900上で実行可能なFTPクライアントです。また、FTPの各コマンドは、DT-900アプリケーションプログラムでの利用を想定したサブセットで、コマンドスクリプトファイルの内容に従って実行されます。

### 2 - 5 - 2 提供ファイル

- ・ ftpcmd.h : マクロ定義等のヘッダーファイル
- ・ ftpcmd.lib : ライブラリファイル

### 2 - 5 - 3 書式

```
int FTP_CMD( commandfilename )
```

パラメータ	説明
commandfilename	FTP コマンドを記述したファイル名

### 2 - 5 - 4 戻り値

実行結果として、以下の正常 / 異常終了コードを返します。

- 0 : 正常終了
- 1 : 無線の初期化に失敗した
- 2 : ポートの初期化に失敗した
- 3 : 同期取りに失敗した
- 4 : コマンドファイルが見つからない
- 5 : 異常終了あり  
(コマンドスクリプトファイル中のコマンドで、1つ以上のコマンドが異常終了したとき)

### 2 - 5 - 5 コマンドスクリプトファイル

FTPの各コマンドの実行手順は、スクリプトファイルに記述します。コマンドスクリプトファイルは、下記書式に従い、テキストエディタ等で作成して下さい。

<書式>

```
FTP コマンド [パラメータ1] [パラメータ2]  
.  
.  
.
```

## < コマンド一覧 >

FTP コマンド	パラメータ 1	パラメータ 2	動作	備考
append	localfilename	remotefilename	サーバへファイルを追加コピー	*2
ascii	-	-	ファイル転送のモードを ASCII に変更	
binary	-	-	ファイル転送のモードを BINARY に変更	
bye	-	-	FTP セッション終了	
debug	true/false	-	DEBUG モード ON/OFF	
get	remotefilename	localfilename	サーバよりファイルを上書きコピー	*2
mget	remotefilename	-	サーバよりファイルを上書きコピー	*1
mput	remotefilename	-	サーバへファイルを上書きコピー	*1
open	remotehostname	-	FTP セッションを接続 (IP で指定)	
put	localfilename	remotefilename	サーバへファイルを上書きコピー	*2
quit	-	-	(bye と同じ)	
recv	remotefilename	localfilename	(get と同じ)	*2
rename	filename	renamefile	ファイル名変更	
send	localfilename	remotefilename	(put と同じ)	*2
user	username	password	ユーザ名 / パスワード設定	*3
verbose	true/false	logfile	verbose モード ON/OFF	*4

\*1 ワイルドカードによる指定が可能です。

\*2 パラメータ 2 は省略可能です。(省略した場合、パラメータ 2 にはパラメータ 1 と同じ情報が設定される)

\*3 “パスワードなし” の場合、パラメータ 2 は不要です。

\*4 パラメータ 1 を false に設定する場合、パラメータ 2 は不要です。

## < コマンドスクリプトファイル例 >

```

VERBOSE true logfile.txt      ... 1
debug true

open serverIP                 ... 2
user taro passwd              ... 3
binary
put hello.exe hello1.exe abc  ... 4
get sakura.bmp
bye

                               ... 5

verbose false
 eof)

```

- 1 FTP コマンドは大文字 / 小文字、どちらでも構いません。
- 2 FTP サーバーの IP アドレスを文字列で指定します。
- 3 各パラメータの間は、スペースかタブで認識します。
- 4 余計なパラメータは無視されます。
- 5 改行のみの行 (空行) を入れても構いません。

## 2 - 5 - 6 デバッグモードと VERBOSE モード

デバッグモード指定、または VERBOSE モード指定を行うと、実行結果をログファイルに出力します。出力フォーマットは、「2 - 6 終了コード」を参照して下さい。



## 2 - 6 終了コード

### 2 - 6 - 1 ログファイル

ログファイルを生成する場合は、下の書式にて記録します。

( 1 ) クライアント側正常 / 異常終了時

“ FTP : コマンド : エラーコード メッセージ ”

( 2 ) サーバー側正常 / 異常終了時

“ FTP : コマンド : FTP 応答コード メッセージ ”

サーバーから送られてきた応答コード / メッセージをそのまま記録します。

( 3 ) DEBUG モード時

“ FTP コマンド文字列 パラメータ ”

クライアント側(DT-900)が、サーバー側FTP サービスに対して送信したパケット内容を出力します。

### 2 - 6 - 2 クライアント側終了コードとメッセージ

終了コード	メッセージ	意味
0	Command completed.	コマンド正常終了
1	FTP connection initialize error.	FTP コネクション初期化エラー
2	FTP connect error.	FTP 接続エラー
3	FTP termination error.	FTP 切断エラー
4	Send timeout error.	送信タイムアウト
5	Recv timeout error.	受信タイムアウト
6	Resume error.	レジュームエラー
7	Memory allocation error.	メモリアロケーションエラー
8	File open error.	ファイルオープンエラー
9	File read error.	ファイル読み込みエラー
10	File write error.	ファイル書込みエラー
11	File no exist error.	ファイル不在エラー
12	Command error.	コマンドエラー
13	Interrupted by user.	ユーザによる中断
14	Buffer over error.	バッファオーバーエラー
15	Parameter error.	パラメータエラー

## 2 - 6 - 3 サーバー側終了コードと意味

応答コード	意 味
110	リスタートマーカ応答。テキストは正常で、特定の実装に依存されません。
120	サービスは、nnn 分で準備できます。
125	データ接続は既にオープンしました。転送は開始しています。
150	ファイルステータスは良好です。データ接続をオープンしています。
200	コマンドはOKです。
202	コマンドは実装されていません。このサイトでは不必要です。
211	システムステータス、またはシステムヘルプ応答。
212	ディレクトリステータス。
213	ファイルステータス。
214	ヘルプメッセージ
215	NAME システムタイプ
220	新規ユーザ用サービスが準備できています。
221	サービスはコントロール接続をクローズします。 適当であれば、ログアウトします。
225	データ接続はオープンしています。進行中の転送はありません。
226	データ接続をクローズします。要求されたファイル処理は完了しました。
227	パッシブモードに入ります。
230	ユーザがログインしました。先に進みます。
250	要求されたファイル処理が正常終了しました。
257	“ PATHNEME ” が作成されました。
331	ユーザ名はOKです。パスワードが必要です。
332	ログイン用のアカウントが必要です。
350	要求されたファイル処理は、追加情報を持っています。
421	サービスは利用できません。コントロール接続をクローズします。
425	データ接続をオープンできません。
426	接続がクローズしました。転送が打ち切られました。
450	要求されたファイル処理が実行されませんでした。 ファイルが使用不可能です。
451	要求された行動が打ち切られました。ローカルの処理エラーです。
452	要求された行動は実行されませんでした。システムの記憶領域が不十分です。
500	シンタックスエラー。コマンドが認識されませんでした。
501	パラメータ或いは引数のシンタックスエラー。
502	コマンドが実装されていません。
503	コマンドのシーケンスが間違っています。
504	このコマンドパラメータは実装されていません。
530	ログインしていません。
532	ファイル格納用のアカウントが必要です。
550	要求されたコマンドが実行されませんでした。ファイルが使用不可能です。
551	要求された行動が打ち切られました。ページタイプが不明です。
552	要求されたファイル処理が打ち切られました。記憶領域が超過しました。
553	要求された行動が実行されませんでした。ファイル名が不当です。

注) メッセージの言語に関しては、サーバー側FTP サービスに依存します。

## 2 - 6 - 4 ログファイル出力例

### ログファイルの例1 (verbose=TRUE, debug=FALSE のとき)

```
FTP : verbose : 0 Command completed.
FTP : debug   : 0 Command completed.
FTP : open    : 220 phoenix Windows NT FTP Server (Version 3.51).
FTP : open    : 0 Command completed.
FTP : user    : 331 Password required for casio.
FTP : user    : 230 User casio logged in.
FTP : user    : 0 Command completed.
FTP : binary  : 200 Type set to I.
FTP : binary  : 0 Command completed.
FTP : get     : 200 PORT command successful.
FTP : get     : 150 Opening BINARY mode data connection for lanma256.bmp.
FTP : get     : 145314 bytes received in 9300 ms (15625 bytes/sec)
FTP : get     : 226 Transfer complete.
FTP : get     : 0 Command completed.
FTP : rename  : 350 File exists, ready for destination name
FTP : rename  : 250 RNT0 command successful.
FTP : rename  : 0 Command completed.
FTP : bye     : 221 Goodbye.
FTP : bye     : 0 Command completed.
.
.
.
```

### ログファイルの例2 (verbose=TRUE, debug=TRUE のとき)

```
FTP : verbose : 0 Command completed.
FTP : debug   : 0 Command completed.
FTP : open    : 220 phoenix Windows NT FTP Server (Version 3.51).
FTP : open    : 0 Command completed.
--->USER casio
FTP : user    : 331 Password required for casio.
--->PASS DT-900
FTP : user    : 230 User casio logged in.
FTP : user    : 0 Command completed.
--->TYPE I
FTP : binary  : 200 Type set to I.
FTP : binary  : 0 Command completed.
--->PORT 10,5,0,254,12,45
FTP : get     : 200 PORT command successful.
--->RETR lanma256.bmp
FTP : get     : 150 Opening BINARY mode data connection for lanma256.bmp.
FTP : get     : 145314 bytes received in 9300 ms (15625 bytes/sec)
FTP : get     : 226 Transfer complete.
FTP : get     : 0 Command completed.
--->RNFR lanma255.bmp
FTP : rename  : 350 File exists, ready for destination name
--->RNT0 lanma254.bmp
FTP : rename  : 250 RNT0 command successful.
FTP : rename  : 0 Command completed.
--->QUIT@
FTP : bye     : 221 Goodbye.
FTP : bye     : 0 Command completed.
.
.
.
```

### 3 サイトサーベイ関数

*int* AntennaDisp( *int* line, *int* column, *int* loop )

---

<i>int</i>	<i>line</i> ;	アンテナ記号表示行
<i>int</i>	<i>column</i> ;	アンテナ記号表示桁
<i>int</i>	<i>loop</i> ;	状態チェック回数

#### 概要

PHSの電波状態をチェックし、その結果を返します。  
アンテナ記号の表示も行います。

#### 詳細

PHSの電波状態を計る関数 ( phs\_ready : ソケットライブラリの関数 ) を呼び、状態を数値で返します。  
状態を表すアンテナの記号も表示します。

本関数を起動する前に、必ず FTP\_START を実行しておく必要があります。  
(ソケット関数では、Ninit を実行する必要があります)

アンテナ記号のデータは、外字ファイル<ANTENNA.DAT>で用意しているため、このファイルをAドライブのルートディレクトリにダウンロードしておく必要があります。

注) アンテナ記号は16ドットフォント用のデータとして作成しているため、16ドット表示の場合のみ使用可能です。

パラメータ loop は、 phs\_ready を呼び出す回数です。

電波状態は、この phs\_ready を1回実行することで認識は出来ませんが、正しい電波状態は、10秒程度 ( loop の値を100とする ) 繰り返して実行することで得られます。

状態が良 ( 戻り値 = 3 ) 以上で通信を行うようにして下さい。

パラメータ	説明
<i>line</i>	アンテナ記号を表示させる行
<i>column</i>	アンテナ記号を表示させる桁
<i>loop</i>	状態チェックを行う回数

#### 戻り値

- 0 : 圏外
- 1 : 不良 ( アンテナ無し )
- 2 : 不良 ( アンテナ一本 )
- 3 : 良 ( アンテナ二本 )
- 4 : 良 ( アンテナ三本 )
- 1 : 状態チェック不可 ( PHS\_START 未実行 )

#### 備考

本関数は、オブジェクトファイルとソースファイルで提供いたします。

最終ページ