

§ 8 . FLINK ライブラリ

8-1.機能

FLINK ライブラリは、DT900 の通信ユーティリティ関数で提供していた機能の中で、FLINK 通信機能と同等の機能を WindowsCE 上で実現するためのライブラリです。

なお、本体に内蔵している FLCE.EXE は、本 FLINK ライブラリを使用していません。

DT900 の通信ユーティリティ機能との比較

機能名		FLINK ライブラリ	DT900
マルチドロッププロトコル		×	
FLINK プロトコル	ファイル送受信基本機能		
	リモート操作関数		
	ファイルチェック機能関数	× (FCHKCE.EXE で代行可能)	
DT500 プロトコル		×	

8-2.動作環境

機種 DT-9700

OS Microsoft WindowsCE .NET 4.1

8-3.開発環境

Microsoft eMbedded C++ Version4.0 + SP1 or SP2

Microsoft Visual Studio .NET 2003

8-4.提供ファイル

finklib.lib インポートライブラリ

finklib.h ヘッダファイル

8-5.使用方法

・開発環境において、プログラムソース内にヘッダファイル「finklib.h」をインクルードし、インポートライブラリ「finklib.lib」を使用するライブラリとして指定してください。

8-6.ライブラリで使用される構造体の定義

FLK_RSPRM 構造体

シリアルポートの情報を格納する構造体です。

```
typedef struct{
    WORD    speed;    // 転送速度    FLK_B1200~FLK_B115K
    WORD    length;   // データ長    FLK_CHAR8
    WORD    parity;   // パリティ    FLK_PARI_NON   パリティなし
                //                FLK_PARI_ODD   奇数
                //                FLK_PARI_EVN   偶数
    WORD    stop_bit // ストップビット FLK_STOP1   1ビット
                //                FLK_STOP2   2ビット
}FLK_RSPRM;
```

FLK_RSPRM 構造体を使用する関数は以下の通りです。

FLKOpen

FLK_STATUS 構造体

FLINK ライブラリの状態を格納する構造体です。

```
typedef struct {
    WORD    status;           //コマンド実行ステータス
    WORD    function;        //コマンド種類
    TCHAR   *filename;      //送受信の名前
    DWORD   total_size;     //1回の転送の総バイト数
    DWORD   total_count;    //1回の転送の転送済みバイト数
    DWORD   file_size;      //転送中のファイルサイズ
    DWORD   file_count;     //転送中のファイルの転送済みバイト数
} FLK_STATUS;
```

FLK_STATUS 構造体を使用する関数は以下の通りです。

FLKReadStatus

FLK_ERRINFO 構造体

エラー情報を格納する構造体です。

```
typedef struct{
    BYTE    kind;           //エラー種別
    BYTE    command;       //コマンド種別
    BYTE    category;      //カテゴリ
    BYTE    detail;        //エラー詳細
    DWORD   biosStat;      //GetLastError()で取得したエラーコード
}FLK_ERRINFO;
```

FLK_ERRINFO 構造体を使用する関数は以下の通りです。

FLKReadErrStat

FLK_DATETIME 構造体

日付時刻を格納する構造体です。

```
typedef struct{
    BYTE    day;           // 日(1-31)
    BYTE    month;        // 月(1-12)
    WORD    year;         // 年(1980-2079)
    BYTE    sec;          // 秒(0-59)
    BYTE    min;          // 分(0-59)
    BYTE    hour;         // 時(0-23)
}FLK_DATETIME;
```

FLK_DATETIME 構造体を使用する関数は以下の通りです。

FLKMakeDir

FLKGetFileInfo

FLKSetFileInfo

FLKDateTime

FLK_FINFO 構造体

ファイル情報を格納する構造体です。

```
typedef struct{
    TCHAR    name[256]    // 検索されたファイル名(フルパス)
    FLK_DATETIME datetime; // 日付時刻格納エリア
    DWORD    size        // サイズ
    BYTE     atr;        // 属性
                                FLK_A_NORMAL   ;通常ファイル
                                FLK_A_HIDDEN   ;不可視ファイル
                                FLK_A_RDONLY   ;読み出し専用ファイル
                                FLK_A_SYSTEM   ;システムファイル
                                FLK_A_ARCH    ;アーカイブ
}FLK_FINFO;
```

FLK_A_NORMAL は DT900 の _A_NROMAL との互換のために用意していますが、値は 0 に定義されているため、設定してもしなくても動作に変化はありません。

FLK_FINFO 構造体を使用する関数は以下の通りです。

FLKGetFileInfo

FLKSetFileInfo

FLK_DINFO 構造体

ディスク情報を格納する構造体です。

```
typedef struct{
    DWORD    size;        // ディスク容量
    DWORD    freex;      // ディスク空き容量
    BYTE     status      // ディスク状態
                                FLK_DINFO_NORMAL: ディスク有り(フォーマット済み)
                                FLK_DINFO_NOFMT: ディスク有り(未フォーマット)
                                FLK_DINFO_NODISK: ディスクなし
}FLK_DINFO;
```

FLK_DINFO 構造体を使用する関数は以下の通りです。

FLKGetDiskInfo

FLK_SYSINFO 構造体

システム情報を格納する構造体です。

```
typedef struct{
    WORD    id;           // セッション ID(PC との接続時は0)
    BYTE    ftpver;      // プロトコルバージョン
    BYTE    code[3];     // 機種コード
    BYTE    model;       // モデル情報
}FLK_SYSINFO;
```

FLK_SYSINFO 構造体を使用する関数は以下の通りです。

FLKGetSysInfo

8-7.ライブラリー一覧

NO.	関数名	(DT900 の関数名)	機能
01	FLKOpen	cu_open	回線オープン
02	FLKFileSend	cu_fileSend	ファイル送信
03	FLKFileAdd	cu_fileAdd	ファイル追加
04	FLKFileRecv	cu_fileRecv	ファイル受信
05	FLKClose	cu_close	回線クローズ
06	FLKReadErrStat	cu_readErrStat	エラー情報の取得
07	FLKReadStatus	- - -	進捗情報のリード
08	FLKIdle	cu_idle	アイドル遷移
09	FLKCmdRecv	cu_cmdRecv	コマンド受信待ち
10	FLKFileDelete	cu_fileDelete	ファイル削除
11	FLKFileMove	cu_fileMove	ファイル移動
12	FLKMakeDir	cu_makeDir	ディレクトリ作成
13	FLKGetFileInfo	cu_getFileInfo	ファイル情報の取得
14	FLKSetFileInfo	cu_setFileInfo	ファイル情報の設定
15	FLKGetDiskInfo	cu_getDiskInfo	ディスク情報の取得
16	FLKDateTime	cu_dateTime	日付時刻の取得および設定
17	FLKGetSysInfo	cu_getSysInfo	システム情報の取得
18	FLKMsgSend	cu_msgSend	画面表示メッセージの送信
19	FLKBeep	cu_beep	ブザー鳴動
	- - -	cu_setIoboxInfo	IOBOX 情報設定は未サポート
	- - -	cu_fchklog_Create	FCHK リストファイルの作成は未サポート
	- - -	cu_fchklog_Check	FCHK リストファイルのチェックは未サポート

関数名の規則（DT900 の通信ユーティリティとの比較）

以下の規則で、DT900 の通信ユーティリティから名前を変更しました。

	DT900	本ライブラリ
関数名	cu_で始まる	FLK で始まる
	cu_の後の最初の文字は小文字	FLK の後の最初の文字は大文字
構造体名	CU_で始まる	FLK_で始まる
定数名	CU_で始まる	FLK_で始まる
関数の戻り値	E_OK/E_NG/E_PRM	FLK_OK/FLK_NG/FLK_PRM
パラメータ	B/H/W	BYTE/WORD/DWORD

8-8.関数リファレンス

次頁より、関数リファレンスを説明します。

FLKOpen

- 機能** 回線オープンします。
通信ポートの初期化およびセッションの確立を行います。
相手局のシステム情報の取得を行います。
- 書式** HANDLE FLKOpen(TCHAR *device, char *cString, FLK_RSPRM *rsPrm, WORD mode, HWND hWnd, UINT message)
- パラメータ**
- TCHAR device** 使用デバイスを指定します。
"COM1" シリアル(COM1)
"IRDA" 赤外線
"LAN" LAN
- char * cString** 通信パラメータを設定します
FLK_DEVICE_COM1 指定時 :無視します
FLK_DEVICE_IRDA 指定時 :無視します
FLK_DEVICE_LAN 指定時 :ホストPCのIPアドレスを指定します
- FLK_RSPRM *rsPrm** 通信パラメータを設定します
FLK_DEVICE_COM1 指定時 :COMポート通信パラメータを指定します
FLK_DEVICE_IRDA 指定時 :無視します
FLK_DEVICE_LAN 指定時 :無視します
- WORD mode** 局モード
FLK_MODE_HT :HTモード
FLK_MODE_PC :PCモード
- HWND hWnd** メッセージを通知するウィンドウの、ウィンドウハンドルを指定します。
NULLを指定された場合はメッセージ通知しません。
- UINT message** アプリケーションへ通知するメッセージ番号を指定します。
- 戻り値**
- FLK_PRM パラメータエラー
FLK_NG パラメータ以外の異常終了
上記以外 FLINK ライブラリのハンドル(正常終了)

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	オープン処理が正常終了した
FLK_COMMAND_ERROR	-	オープン処理が異常終了した

インクルード #include <flinklib.h>

FLKFileSend

機能 指定された複数ファイルを一括して送信します。
転送先ディレクトリが存在しない場合は自動的に生成します。

書式 `DWORD FLKFileSend(HANDLE hPort, WORD mode, TCHAR *fName, TCHAR *dir, WORD protect)`

パラメータ

HANDLE hPort オープン時に取得したハンドル

WORD mode 転送モード(通常転送か再起呼び出し転送かを指定します)

FLK_TRANS_NORMAL :通常転送

FLK_TRANS_RECURSIVE :再起呼び出し転送

TCHAR *fName 送信元ファイル名(フルパスで指定、ワイルドカード可能)

TCHAR *Dir 送信先フォルダ名(フルパスで指定)

WORD protect 強制上書きフラグ(受信側に同一ファイルが書き込み禁止属性で存在した場合、属性変更して書き込むかを指定します)

FLK_PROTECT_VALID :強制書き込みしない

FLK_PROTECT_INVALID :強制書き込みする

戻り値

FLK_OK 正常終了

FLK_PRM パラメータエラー

FLK_NG パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	指定した全ファイルの送信が完了した
FLK_COMMAND_ERROR	-	ファイル送信がエラーにより終了した
FLK_STATUS_CHANGE1	-	1つのファイルの送信が始まった FLKReadStatus()関数で送信ファイル名を取得できます。
FLK_STATUS_CHANGE2	-	ファイル転送バイト数に変化があった FLKReadStatus()関数で転送バイト数を取得できます。

インクルード `#include <flinklib.h>`

FLKFileAdd

機能 指定したファイルを相手局の既存ファイルに追加します。
送信元、追加先ファイル名とも複数ファイルの指定および、ワイルドカードの指定はできません。
追加先ファイル名が相手局側に存在しない場合、新規にファイルを作成します。

書式 `DWORD FLKFileAdd(HANDLE hPort, TCHAR *sfName, TCHAR *rfName)`

パラメータ

<code>HANDLE hPort</code>	オープン時に取得したハンドル
<code>TCHAR *sfName</code>	送信元ファイル名(フルパスで指定)
<code>TCHAR *rfName</code>	追加先ファイル名(フルパスで指定)

戻り値

<code>FLK_OK</code>	正常終了
<code>FLK_PRM</code>	パラメータエラー
<code>FLK_NG</code>	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	指定したファイルの追加が完了した
<code>FLK_COMMAND_ERROR</code>	-	ファイル追加がエラーにより終了した
<code>FLK_STATUS_CHANGE1</code>	-	ファイル追加送信が始まった FLKReadStatus()関数で送信ファイル名を取得できます。
<code>FLK_STATUS_CHANGE2</code>	-	ファイル転送バイト数に変化があった FLKReadStatus()関数で転送バイト数を取得できます。

インクルード `#include <flinklib.h>`

FLKFileRecv

機能 指定された複数ファイルを一括して受信します。
受信先ディレクトリが存在しない場合は自動的に作成します。

書式 `DWORD FLKFileRecv(HANDLE hPort, WORD mode, TCHAR *fName, TCHAR *dir, WORD protect)`

パラメータ

HANDLE hPort	オープン時に取得したハンドル
WORD mode	転送モード(通常転送か再起呼び出し転送かを指定します) FLK_TRANS_NORMAL :通常転送 FLK_TRANS_RECURSIVE :再起呼び出し転送
TCHAR *fName	受信ファイル名(フルパスで指定、ワイルドカード可能)
TCHAR *Dir	受信フォルダ名(フルパスで指定)
WORD protect	強制上書きフラグ(受信側に同一ファイルが書き込み禁止属性で存在した場合、属性変更して書き込むかを指定します) FLK_PROTECT_VALID :強制書き込みしない FLK_PROTECT_INVALID :強制書き込みする

戻り値

FLK_OK	正常終了
FLK_PRM	パラメータエラー
FLK_NG	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	ファイルの受信が完了した
FLK_COMMAND_ERROR	-	ファイル受信がエラーにより終了した
FLK_STATUS_CHANGE1	-	1ファイルの受信が始まった FLKReadStatus()関数でファイル名を取得できます。
FLK_STATUS_CHANGE2	-	ファイル転送バイト数に変化があった FLKReadStatus()関数で転送バイト数を取得できます。

インクルード `#include <flinklib.h>`

FLKClose

機能 クローズします。

終了指示コマンドを相手に送信することにより、セッションを開放します。

その際、送信権モード時に限り、相手局に対して終了指示コマンドを送信します。

ただし、すでにエラーが発生していた場合は送信されません。

書式 DWORD FLKClose(HANDLE hPort, WORD endKind)

パラメータ HANDLE hPort オープン時に取得したハンドル

WORD endKind 相手局への終了指示(送信権局モードのみ有効)

FLK_CLOSE_NORMAL :正常終了

FLK_CLOSE_RESET :リセット指示

FLK_CLOSE_FORMAT_A :Aドライブフォーマット指示

FLK_CLOSE_FORMAT_B :Bドライブフォーマット指示

FLK_CLOSE_FORMAT_C :Cドライブフォーマット指示

FLK_CLOSE_PWROFF :電源 OFF 指示

これらのパラメータは相手局へ通知するだけです。

FLKIdle 又は FLKCmdRecv 関数が正常終了したとき、FLKReadErrStat 関数で相手局からどのコードを指定されたかを知ることができます。

戻り値 FLK_OK 正常終了

FLK_PRM パラメータエラー

FLK_NG パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	クローズ処理が正常終了した。
FLK_COMMAND_ERROR	-	クローズ処理が異常終了した。

インクルード #include <flinklib.h>

FLKReadStatus

機能 FLINK ライブラリ内の通信状態を取得します。

書式 `DWORD FLKReadStatus(HANDLE hPort, FLK_STATUS *flkstatus)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル

`FLK_STATUS *flkstatus` 通信状態格納エリア

FLK_STATUS のメンバーには以下の内容が設定されます。

status 最後に実行したコマンドが実行中か、終了したかの状態がセットされます。

FLK_STATUS_RUNNING 実行中

FLK_STATUS_END 正常終了

FLK_STATUS_ERROR エラー発生により中断

function と FileName には以下の状態がセットされます。

ただし、関数呼び出し時に、FileName メンバに NULL が指定されていた場合には、FileName にはなにも設定されません。

function	FileName	内容
FLK_SND_EXEC	送信ファイル名	ファイル送信中
FLK_RCV_EXEC	受信ファイル名	ファイル受信
FLK_ADDSND_EXEC	送信ファイル名	ファイル追加送信中
FLK_ADDRVC_EXEC	送信元ファイル名	ファイル追加受信
FLK_DELETE_EXEC	削除ファイル名	相手局のファイル削除要求中
FLK_DELETE_RECV	削除ファイル名	自局のファイル削除中
FLK_MOVE_EXEC	移動元のファイル名	相手局のファイルの移動要求中
FLK_MOVE_RECV	移動元のファイル名	自局のファイル移動中
FLK_MAKEDIR_RECV	作成ディレクトリ名	ディレクトリ作成中
FLK_DATETIME_EXEC	- - - -	相手局の日付時刻変更要求中
FLK_DATETIME_RECV	- - - -	日付時刻変更要求受信
FLK_DATETIME_SEND	- - - -	相手局の日付時刻送信
FLK_MESSAGE_RECV	要求された表示メッセージ	メッセージ表示要求受信
FLK_BUZZER_RECV	- - - -	ブザー鳴動要求受信
FLK_GETFILEINFO_RECV	処理中のファイル名	ファイル情報取得要求受信
FLK_SETFILEINFO_RECV	処理中のファイル名	ファイル情報設定要求中
FLK_GETDISKINFO_RECV	ドライブ名	ディスク情報受信

total_size 一回のコマンド単位で転送する総バイト数がセットされる。

total_count 一回のコマンド単位で転送され送バイト数の内、転送済みバイト数

file_size 1ファイルの転送バイト数(ファイルサイズ)

file_count 1ファイルの転送済みバイト数

戻り値 `FLK_OK` 正常終了

`FLK_PRM` パラメータエラー

`FLK_NG` パラメータ以外の異常終了

メッセージ

メッセージはありません。

インクルード `#include <flinklib.h>`

FLKReadErrStat

機能 エラー情報を取得します。
また、相手局からの終了指示コマンド受信時、カテゴリコード、エラー詳細コードを取得します。
取得後エラー情報はクリアされます。

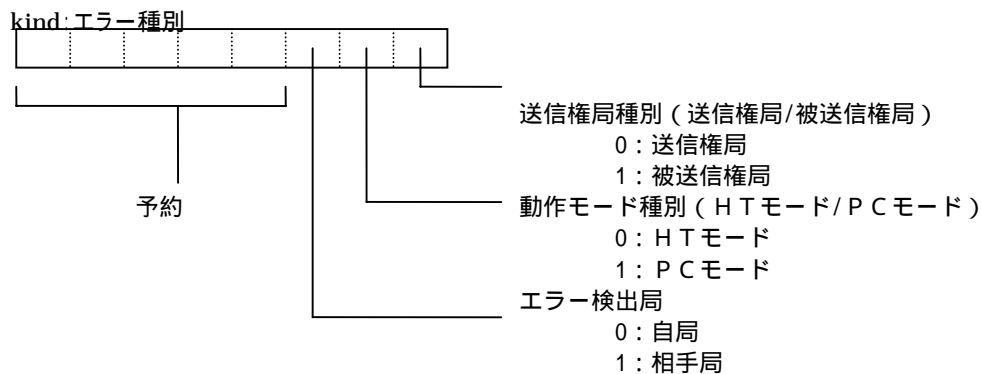
書式 `DWORD FLKReadErrStat(HANDLE hPort, FLK_ERRINFO *errinfo)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`FLK_ERRINFO *errinfo` エラー情報設定エリア

構造体

エラー情報を格納する構造体です。

```
typedef struct{
    BYTE    kind;           //エラー種別
    BYTE    command;       //コマンド種別
    BYTE    category;      //カテゴリ
    BYTE    detail;        //エラー詳細
    DWORD   biosStat;      //GetLastError()で取得したエラーコード
}FLK_ERRINFO;
```



戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー

メッセージ

メッセージはありません。

インクルード `#include <flinklib.h>`

エラー情報の取得 コマンド種別・エラー状態 一覧

コマンド種別		
値	シンボル	意味
0 0 h	FLK_CMD_NON	該当コマンドなし
0 1 h	FLK_CMD_FSEND_TINFO	ファイル転送情報コマンド
0 2 h	FLK_CMD_FSEND_FINFO	ファイル情報コマンド
0 3 h	FLK_CMD_FRECV_TREQ	ファイル受信要求コマンド
0 4 h	FLK_CMD_FADD	ファイル追加コマンド
0 5 h	FLK_CMD_FDATA	ファイルデータコマンド
0 6 h	FLK_CMD_FDEL	ファイル削除コマンド
0 7 h	FLK_CMD_FMOV	ファイル移動コマンド
0 8 h	FLK_CMD_MAKEDIR	ディレクトリ作成コマンド
0 9 h	FLK_CMD_TIME_SET	日付時刻設定コマンド
0 A h	FLK_CMD_TIME_GET	日付時刻取得コマンド
0 B h	FLK_CMD_DISP	メッセージ表示コマンド
0 C h	FLK_CMD_BEEP	ブザー鳴動コマンド
0 D h	FLK_CMD_FINFO_GET	ファイル情報取得コマンド
0 E h	FLK_CMD_FINFO_SET	ファイル情報設定コマンド
0 F h	FLK_CMD_DINFO_GET	ディスク情報取得コマンド
1 0 h	FLK_CMD_SYS_GET	システム情報取得コマンド
1 1 h	FLK_CMD_IDLE	I D L E通知コマンド
1 2 h	FLK_CMD_END	終了指示コマンド
カテゴリコード・エラー詳細コード カテゴリとエラー詳細コードの組み合わせによりエラー状態を表す。		
値	意味	
カテゴリ	詳細	
正常終了状態		
0 0 h	0 0 h	正常終了
D C h ~ F 5 h	0 0 h	フォーマット指示コマンド (A ~ Z)
F 6 h	0 0 h	電源 OFF 終了通知
F 7 h	0 0 h	リセット指定終了通知
F 8 h	0 0 h	中断キーによる終了通知
F 9 h ~ F F h	-	予約領域
プロトコルエラー		
0 1 h	0 0 h	受信フレームファンクションコード 未定義エラー
	0 1 h	受信フレームサブ ファンクションコード 未定義エラー
	0 3 h	受信フレームチェックサムエラー
	0 4 h	シーケンスエラー
	0 5 h	シーケンス番号エラー
	0 7 h	受信フレーム内情報パ ラメータエラー
	0 8 h	受信タイムアウト
	1 0 h	コマンド レング スエラー
ファイルエラー [プロトコル論理]		
0 4 h	0 0 h	リードオンリファイルアクセスエラー

ユーティリティエラー		
1 0 h	0 0 h	回線オープンエラー ・回線がオープンされていない ・オープン時にエラーが発生していないか確認
	0 1 h	使用関数フェーズエラー ・関数の使い方に誤りがある。 ・動作モード / 送信権局モードを確認
	0 2 h	使用関数パラメータエラー ・関数パラメータに誤りがある ・指定パラメータを確認
	0 3 h	指定ファイル未検出エラー ・定されたファイルが存在しない ・指定ファイルを確認
	0 4 h	相手局未検出 ・セッション確立待ちタイムアウト ・通信設定、回線経路を確認
	0 5 h	システム日付設定エラー ・指定日付を確認
	0 6 h	システム時刻設定エラー ・指定時刻を確認
	0 9 h	致命的エラー
	0 A h	通信中回線断エラー ・通信中に回線が切断された。回線経路を確認
	0 B h	ドライブ容量不足 ・指定ドライブの容量が足りない
	0 C h	コマンド実行中エラー ・コマンド実行中に次のコマンドを実行した。
ファイルエラー [ファイル関数]		
1 1 h	0 0 h	クリエートエラー
	0 1 h	オープンエラー
	0 2 h	リードエラー
	0 3 h	ライトエラー
	0 4 h	シークエラー
	0 5 h	ファイル削除エラー
	0 6 h	ディレクトリ削除エラー
	0 7 h	ファイル名変更移動エラー
	0 8 h	タイムスタンプ設定エラー
	0 9 h	タイムスタンプ取得エラー
	0 A h	ファイル属性設定エラー
	0 B h	ファイル属性取得エラー
	0 C h	ディレクトリ作成エラー
	0 D h	ファイルサイズ変更エラー
システム異常エラー		
0 F h	0 x h	F T P 部内部エラー
	1 x h	通信ユーティリティ内部エラー

FLKIdle

機能 アイドル通知送信後、相手局からのコマンド受信待ち状態となります。
HT モード時のみ使用可能です。
終了指示コマンドを受信するか、エラーが発生するまで、相手局から受信したコマンドを順次実行します。

書式 `DWORD FLKIdle(HANDLE hPort, TCHAR *script)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`TCHAR *script` スクリプトファイル名(ファイル名のみ。拡張子を含め 13 文字以内)
未設定時は NULL を指定します。

戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー
`FLK_NG` パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	相手局から終了指示コマンドを受信した。
<code>FLK_COMMAND_ERROR</code>	-	エラーが発生により、コマンドが終了した。
<code>FLK_STATUS_CHANGE1</code>	-	コマンドに変化があったときに通知されます。 コマンドの内容は、 <code>FLKReadStatus</code> 関数で取得できます。
<code>FLK_STATUS_CHANGE2</code>	-	ファイル転送バイト数に変化があった <code>FLKReadStatus()</code> 関数で転送バイト数を取得できます。

インクルード `#include <flinklib.h>`

FLKCmdRecv

機能 HT からのコマンド受信待ち状態となります。

PC モード時のみ使用可能です。

アイドル通知コマンド、終了指示コマンドを受信するか、エラーが発生するまで、相手局から受信したコマンドを順次実行します。

書式 `DWORD FLKCmdRecv(HANDLE hPort, WORD *endKind, TCHAR *script)`

パラメータ

HANDLE hPort オープン時に取得したハンドル

WORD *endKind 終了種別フラグ設定エリア (正常終了時のみ有効)

CU_RECV_END : 終了指示受信

CU_RECV_IDLE : IDLE 通知受信

TCHAR *script スクリプトファイル名 (ファイル名のみ。拡張子を含め 13 文字以内)

相手局からスクリプトが指定された場合には、script で指定されたエリアへスクリプト名が設定されます。スクリプトが指定されない場合は、NULL を設定します。

戻り値

FLK_OK 正常終了

FLK_PRM パラメータエラー

FLK_NG パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	相手局から終了指示又は IDLE 通知コマンドを受信した。
FLK_COMMAND_ERROR	-	エラー発生によりコマンドが終了した。
FLK_STATUS_CHANGE1	-	コマンドに変化があったときに通知されます。コマンドの内容は、FLKReadStatus 関数で取得できます。
FLK_STATUS_CHANGE2	-	ファイル転送バイト数に変化があった FLKReadStatus 関数で転送バイト数を取得できます。

インクルード `#include <flinklib.h>`

FLKFileDelete

機能 ファイルを削除します。

書式 `DWORD FLKFileDelete(HANDLE hPort, TCHAR *fName)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`TCHAR *fName` 削除するファイル名(フルパスで指定。ワイルドカード可能)

戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー
`FLK_NG` パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	ファイル削除が正常終了した。
FLK_COMMAND_ERROR	-	ファイル削除がエラー終了した。

インクルード `#include <flinklib.h>`

FLKFileMove

機能 ファイルを移動します。同一フォルダを指定している場合は、ファイル名を変更します。

書式 `DWORD FLKFileMove(HANDLE hPort, TCHAR *sfName, TCHAR *dfName)`

パラメータ

<code>HANDLE hPort</code>	オープン時に取得したハンドル
<code>TCHAR *sfName</code>	移動元ファイル名(フルパスで指定)
<code>TCHAR *dfName</code>	異動先ファイル名(フルパスで指定)

戻り値

<code>FLK_OK</code>	正常終了
<code>FLK_PRM</code>	パラメータエラー
<code>FLK_NG</code>	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	ファイル削除が正常終了した。
<code>FLK_COMMAND_ERROR</code>	-	ファイル削除がエラー終了した。

インクルード `#include <flinklib.h>`

FLKMakeDir

機能 ディレクトリを作成します。

書式 `DWORD FLKMakeDir(HANDLE hPort, TCHAR *mDir, FLK_DATETIME *datetime, BYTE atr)`

パラメータ

HANDLE hPort	オープン時に取得したハンドル
TCHAR *mDir	作成ディレクトリ名
FLK_DATETIME *datetime	日付格納エリア
BYTE atr	属性(OR 指定による複数指定可)
	FLK_A_NORMAL : 通常ファイル
	FLK_A_HIDDEN : 不可視ファイル
	FLK_A_RDONLY : 読み出し専用ファイル
	FLK_A_SYSTEM : システムファイル
	FLK_A_SUBDIR : ディレクトリ
	FLK_A_ARCH : アーカイブ

FLK_A_NORMAL の値は 0 なので、設定する必要はありません。また、FLK_A_SUBDIR はライブラリ内で自動的に OR されます。

日付時刻を指定しない場合には、datetime 構造体のメンバーの、year に 0xFFFF、day,month,sec,min,hour に 0xFF を設定してください。

戻り値

FLK_OK	正常終了
FLK_PRM	パラメータエラー
FLK_NG	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	ディレクトリ作成が正常終了した。
FLK_COMMAND_ERROR	-	ディレクトリ作成がエラー終了した。

インクルード `#include <flinklib.h>`

FLKGetFileInfo

機能 相手局側の指定ファイル情報(ファイルサイズ・タイムスタンプ・属性)の取得を行います。
 検索ファイル名と一致するファイルの情報がファイル情報エリアに設定されます。
 ワイルドカード指定時は1回目に「最初の取得」、2回目以降に「次情報取得」を指定します。
 ワイルドカード指定時は、この関数を連続的に呼ぶ必要があります。
 他の通信関数を使用すると、次情報取得は行えなえません。

書式 `DWORD FLKGetFileInfo(HANDLE hPort, WORD mode, TCHAR *fName, FLK_FINFO *fInfo)`

パラメータ

HANDLE hPort オープン時に取得したハンドル

WORD mode 最初 / 次フラグ
 FLK_GET_FIRST: 最初の取得 (1ファイル指定又はワイルドカード指定時の1回目)
 FLK_GET_NEXT: 次情報取得 (ワイルドカード指定時の2回目以降)

TCHAR *fName 検索ファイル名エリア (ワイルドカード指定可。複数指定不可。
 「次情報取得」では参照しません。)

FLK_FINFO *finfo ファイル情報エリア (検索したファイルの情報が設定されます)
 該当ファイルが存在しない場合には、各パラメータに 0x00 が設定されます。

戻り値

FLK_OK 正常終了

FLK_PRM パラメータエラー

FLK_NG パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	正常終了した。
FLK_COMMAND_ERROR	-	エラー終了した。

インクルード `#include <flinklib.h>`

FLKSetFileInfo

機能 相手局側の指定ファイル情報(ファイルサイズ・タイムスタンプ・属性)の更新を行います。
ファイル情報エリアの内容をファイル名エリアと一致するファイルに設定します。

書式 `DWORD FLKSetFileInfo(HANDLE hPort, FLK_FINFO *fInfo)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`FLK_FINFO *fInfo` ファイル情報エリア

戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー
`FLK_NG` パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	正常終了した。
<code>FLK_COMMAND_ERROR</code>	-	エラー終了した。

インクルード `#include <flinklib.h>`

FLKGetDiskInfo

機能 相手局側の指定ドライブ情報の取得を行います。
指定ドライブの情報がドライブ情報エリアへ設定されます。

書式 `DWORD FLKGetDiskInfo(HANDLE hPort, BYTE drive, FLK_DINFO *dinfo)`

パラメータ

HANDLE hPort	オープン時に取得したハンドル
BYTE drive	ドライブ名エリア、'A' ~ 'Z'のいずれか
FLKDINFO *dinfo	ドライブ情報エリアアドレス(検索したドライブの情報が設定されます) コマンド正常終了メッセージ受信時に、dinfo に値が設定されます。

戻り値

FLK_OK	正常終了
FLK_PRM	パラメータエラー
FLK_NG	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	正常終了した。
FLK_COMMAND_ERROR	-	エラー終了した。

インクルード `#include <flinklib.h>`

FLKDateTime

機能 日時を取得 / 設定します。
 取得の場合は、日付時刻エリアへ相手局のシステム日付が設定されます。
 設定の場合は、日付時刻エリアの値を相手局へ送信します。

書式 `DWORD FLKDateTime(HANDLE hPort, WORD mode, FLK_DATETIME *dateTime)`

パラメータ

<code>HANDLE hPort</code>	オープン時に取得したハンドル
<code>WORD mode</code>	取得 / 設定フラグ
	<code>FLK_GET_MODE</code> 取得
	<code>FLK_SET_MODE</code> 設定
<code>FLK_DATETIME dateTime</code>	設定日付時刻格納エリア

戻り値

<code>FLK_OK</code>	正常終了
<code>FLK_PRM</code>	パラメータエラー
<code>FLK_NG</code>	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	正常終了した。
<code>FLK_COMMAND_ERROR</code>	-	エラー終了した。

インクルード `#include <flinklib.h>`

FLKGetSysInfo

機能 相手局側のシステム情報を取得します。
 相手局が PC の場合は、接続セッション番号も返します。(相手が HT の場合は 0)
 これらの情報はオープン時に取得しているため、通信は行わずに情報のみ返します。

書式 `DWORD FLKGetSysInfo(HANDLE hPort, FLK_SYSINFO sysinfo)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`FLK_SYSINFO *sysinfo` 取得システム情報エリア (検索された情報が設定されます)

戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー
`FLK_NG` パラメータ以外の異常終了

メッセージ

メッセージはありません。

インクルード `#include <flinklib.h>`

FLKMsgSend

機能 相手局に表示するメッセージを送信します。

書式 `DWORD FLKMsgSend(HANDLE hPort, TCHAR *msg)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル
`TCHAR *msg` 表示メッセージ格納エリア

戻り値 `FLK_OK` 正常終了
`FLK_PRM` パラメータエラー
`FLK_NG` パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
FLK_COMMAND_END	-	正常終了した。
FLK_COMMAND_ERROR	-	エラー終了した。

インクルード `#include <flinklib.h>`

FLKBeep

機能 相手局側にブザーを鳴動を要求します。

書式 `DWORD FLKBeep(HANDLE hPort)`

パラメータ `HANDLE hPort` オープン時に取得したハンドル

戻り値

<code>FLK_OK</code>	正常終了
<code>FLK_PRM</code>	パラメータエラー
<code>FLK_NG</code>	パラメータ以外の異常終了

メッセージ

WPARAM	LPARAM	内容
<code>FLK_COMMAND_END</code>	-	正常終了した。
<code>FLK_COMMAND_ERROR</code>	-	エラー終了した。

インクルード `#include <flinklib.h>`

8-9.通信仕様

8-9-1. 通信インターフェイス

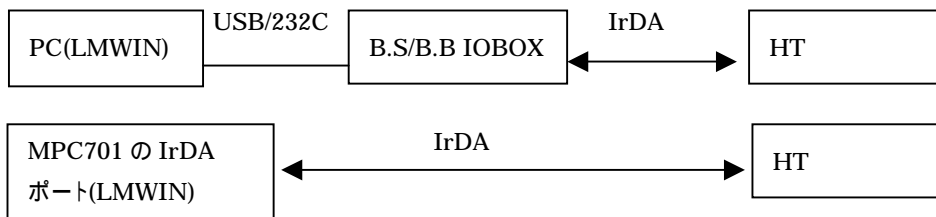
FLINK ライブラリは以下の通信ポートをサポートします。

機種 \ 通信ポート	赤外線 (IrDA)	シリアル	LAN
DT-9700			

8-9-2. 通信構成

本ライブラリでは、以下の構成でのファイル転送機能を提供します。

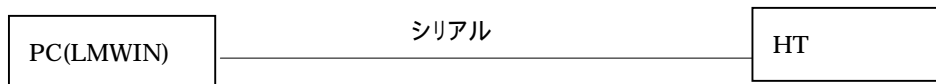
(1) B.S/B.B IOBOX、及び FIVA(MPC701)を使用した IrDA 通信



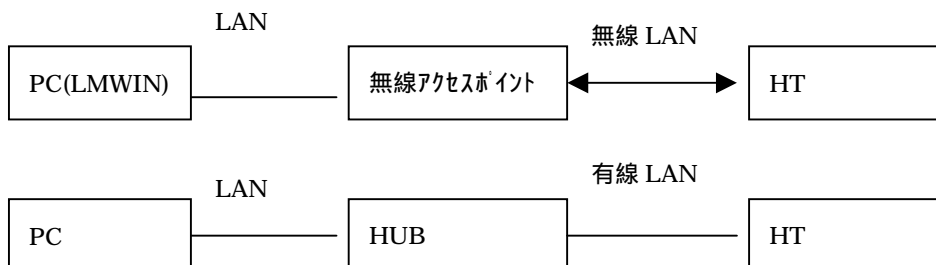
(2) IrDA による本体間通信



(3) シリアルケーブルによるホスト PC 間通信



(4) 無線 LAN / 有線 LAN を使用したホスト PC 間通信



8-10.通信パラメータ

8-10-1. 共通パラメータ

以下のパラメータはレジストリにより設定を行います。

	機能	設定範囲	レジストリ設定無し時の デフォルト値
セッション確立タイムアウト	回線オープン時、セッション確立までの待ち時間	0 - 3600 秒	1800 秒
受信待ちタイムアウト	コマンド/レスポンス受信待ち時間	0 - 600 秒	30 秒
セッション終了待ちタイムアウト	終了指示コマンド送信側が、相手局セッション終了を確認するまでの待ち時間	0 - 600 秒	3 秒

レジストリの設定内容は、レジストリの項目を参照してください。

8-10-2. IrDA パラメータ

パラメータ	サポートする値
最大通信速度	設定不可 最大通信速度は、システムの設定(レジストリ)によります。
Windows98 モード	B.S/B.BIOBOX を使用して、Windows98 と通信する場合に設定するパラメータです。 FLCE.EXE の/8 パラメータに相当する機能です。 本パラメータは、レジストリにより設定します。

8-10-3. シリアル通信パラメータ

パラメータ	サポートする値
回線速度	選択可能 (1200/2400/4800/9600/19200/38400/57600/115200)
データ長	8 ビット固定
パリティビット	選択可能 (無し / 偶数 / 奇数)
ストップビット	選択可能 (1ビット / 2ビット)
フロー制御	RS/CS フロー制御のみ

8-10-4. LAN 通信パラメータ

パラメータ	サポートする値
接続先 IP アドレス	設定可能

8-11.動作モード

本ライブラリでは接続構成により以下のモードをサポートします。

これらのモードはオープン時に選択します。

(1) Tモード

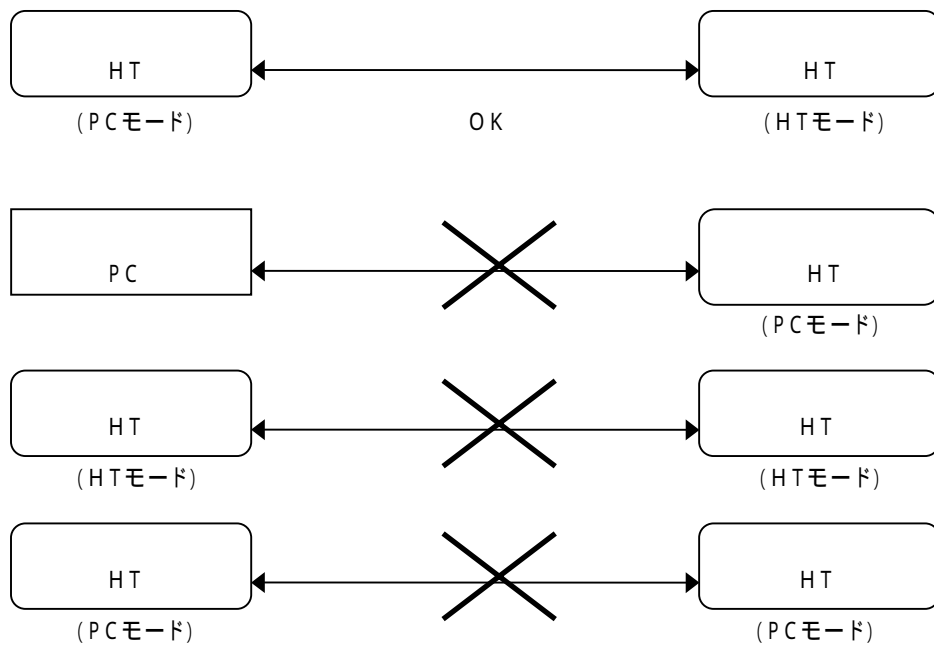
セッション(*1)確立後、コマンドを送信する権利(以後、送信権とする)を持つモードです。

PC - HT間通信時及びHT - HT間通信時(どちらか一方のHT)に選択します。

(2) PCモード

セッション確立後、HTからのコマンド待ちとなる、擬似PCモードです。

HT - HT間通信時(どちらか一方のHT)に選択します。



*1セッションとは回線オープン時に、相手局確認等のネゴシエーションをさします。

8-12.処理概要

以下に各関数内の処理概要及びエラー発生時の処理を示します

エラー発生時は、直ちに通信を終了します。この場合、送信権の有無に関らず、先にエラーを検出した側が

エラー情報(カテゴリコード・エラー詳細コード)を終了指示コマンドに設定し、相手局へ送信します。

相手局は、受信した終了指示コマンドのエラー情報により、異常終了を検出します。

関数	送信権局の処理	被送信権局の処理	エラー発生時の処理
ファイル送信	コマンド送信後、指定ファイルを順次送信。	指定ファイルを順次受信。	受信中ファイルの削除を行う。
ファイル受信	コマンド送信後、指定ファイルを順次受信。	指定ファイルを順次送信。	
ファイル追加	コマンド送信後、指定ファイルを送信。	転送ファイルをテンポラリファイル(FL.ADD)に受信後、ファイルを追加する。追加完了後、テンポラリファイルを削除。	テンポラリファイルを削除。
ファイル削除	コマンド送信。	指定ファイル/ディレクトリを削除。	削除したファイル/ディレクトリの復旧は行わない。
ファイル移動	コマンド送信。	指定ファイルを移動。	移動後の削除ファイルは復旧しない。
ディレクトリ作成	コマンド送信。	指定ディレクトリを作成。	作成したディレクトリは削除しない。
日付時刻の取得	コマンド送信後、日付時刻情報を受信する。	システム日付時刻情報を送信。	—————
日付時刻の設定	コマンド送信。	日付時刻をシステムに設定。	設定後の日付時刻は復旧しない。
ファイル情報の取得	コマンド送信後、ファイル情報を受信する。	指定ファイル情報を送信する。	—————
ファイル情報の設定	コマンド送信。	指定ファイルの情報を変更する。	設定後のファイル情報は復旧しない。
ディスク情報の取得	コマンド送信後、ディスク情報を受信する。	指定ディスク情報を送信する。	—————
システム情報の取得	セッション確立時に相手局情報を取得する為、通信は行わない	セッション確立時に相手局情報を取得する為、通信は行わない	—————
画面メッセージ表示	コマンド送信。	画面メッセージ表示要求を受信したことをアプリケーションに通知する。メッセージ表示は行わない。	—————
ブザー鳴動	コマンド送信。	ブザー鳴動要求を受信したことをアプリケーションに通知する。ブザー鳴動は行わない。	—————
IDLE通知	(HTモードのみ) コマンドを送信後、 コマンド受信待ちとなる。	(PCモードのみ) コマンド送信権を取得する。	—————

8-13.ファイル送受信基本機能

複数ファイルの送信及び受信を行う為の基本機能を提供するものである。

8-13-1. 基本数進関数

ファイル送受信及びリモート操作関数を使用する際に必要となる基本関数を以下に示します。

8-13-1-1. 通信ポートの初期化(FLKOpen 関数)

通信ポートの初期化を行います。

通信ポート毎のパラメータおよび設定値を以下に示します。

COM1	
通信速度	FLK_B1200(1200BPS) / FLK_B2400(2400BPS) / FLK_B4800(4800BPS) FLK_B9600(9600BPS) / FLK_B19K(19.2kbps) / FLK_B38K(38.4kbps) FLK_B57K(57.6kbps) / FLK_B115K(115.2kbps)
データ長	FLK_CHAR8(データ長 8 ビット)
パリティ	FLK_PRI_ODD(奇数) / FLK_PRI_EVEN(偶数) / FLK_PRI_NON(パリティなし)
ストップビット	FLK_STOP1(ストップビット1) / FLK_STOP2(ストップビット2)
IrDA	
設定値なし	
LAN	
接続先 IP アドレス	IP アドレスを指定します。

APO を禁止に設定します。

8-13-1-2. 通信ポートのクローズ(FLKClose 関数)

- ・通信終了及び回線ポートのクローズを行います。通信を終了するために相手局に終了指示コマンドを送信します。
但し、既に相手局より終了指示コマンドを受信していた場合は、終了指示コマンドの送信は行ないません。
- ・終了指示コマンドにはエラー情報(カテゴリコード・エラー詳細コード)を設定します。
- ・回線オープン時にセーブした APO 設定を復旧します。

8-13-1-3. エラー詳細情報の取得(FLKReadErrStat 関数)

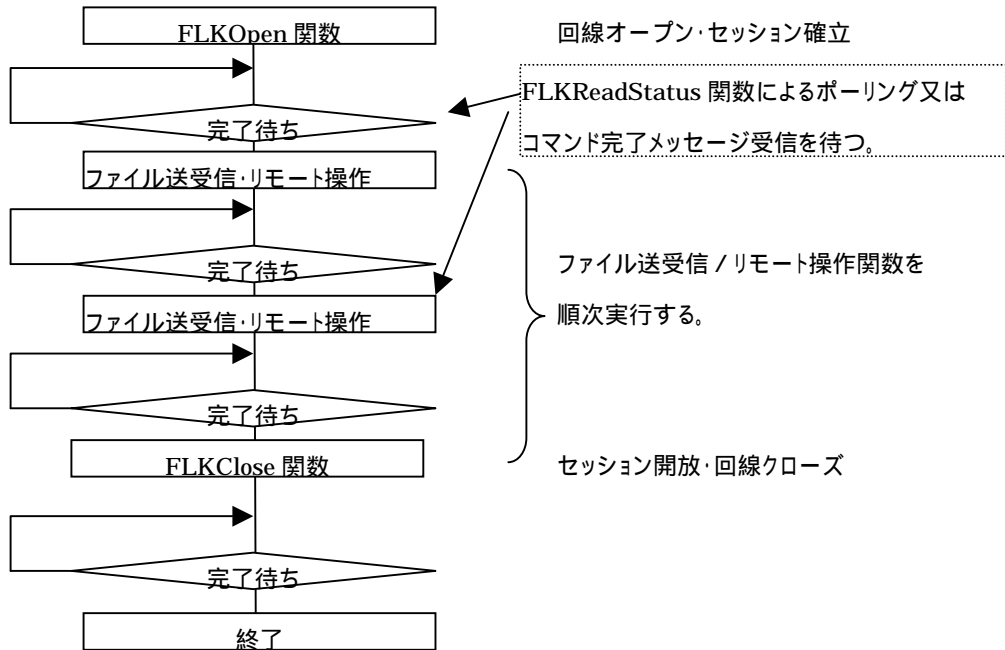
- ・エラー情報の取得を行います。
- ・発生エラーコード、相手局からの終了指示コマンドのエラー情報(カテゴリコード・エラー詳細コード)等の取得が可能です。

8-13-1-4. 通信状態の取得(FLKReadStatus 関数)

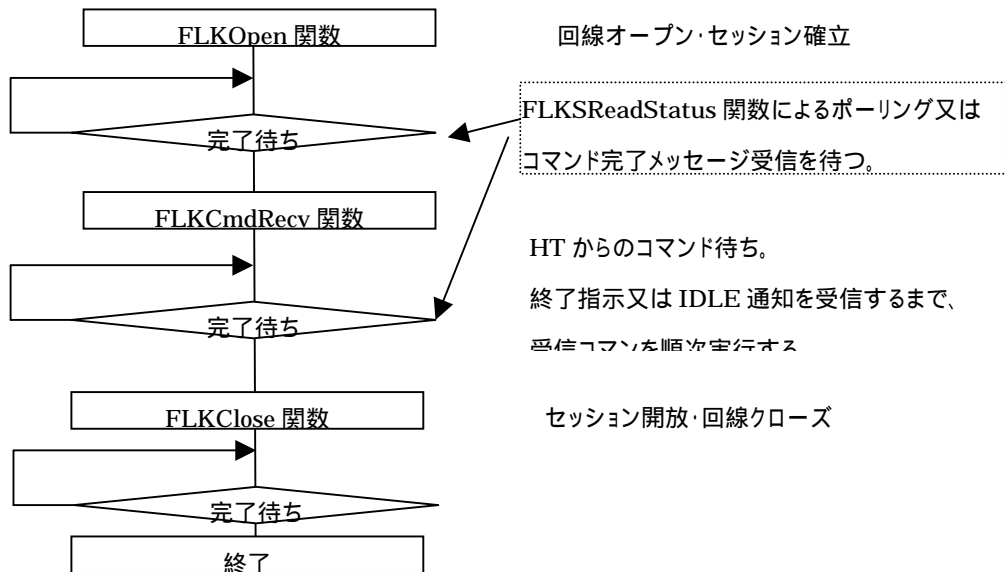
- ・通信の進捗状態(現在実行中の機能、ファイルの転送バイト数など)を取得できます。
- ・最後に実行したコマンドが終了したか、実行中かを知ることができます。

HTコマンド送信による通信

【HTモード基本フロー】

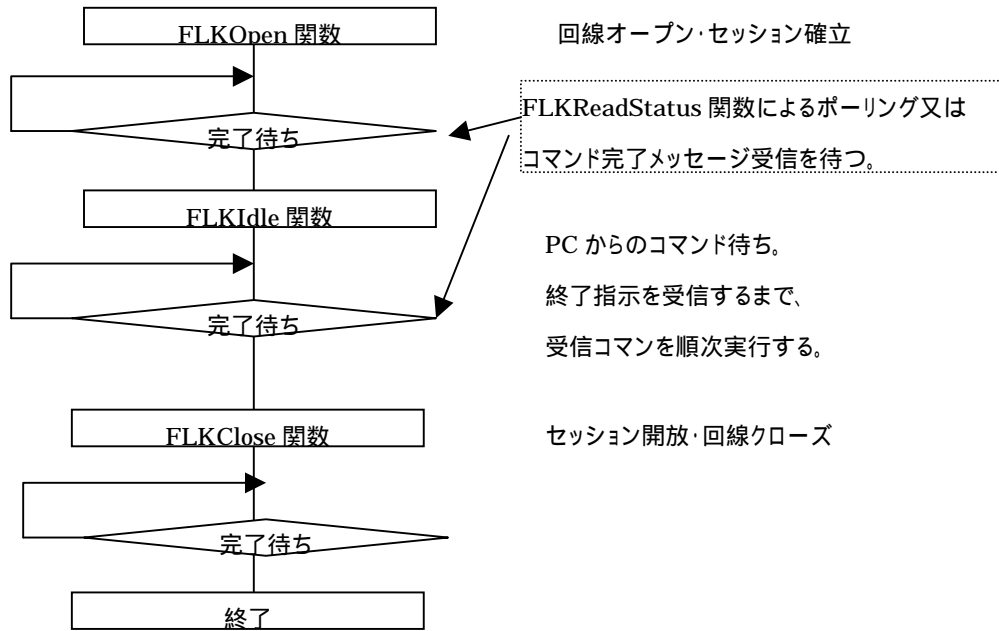


【PCモード基本フロー】

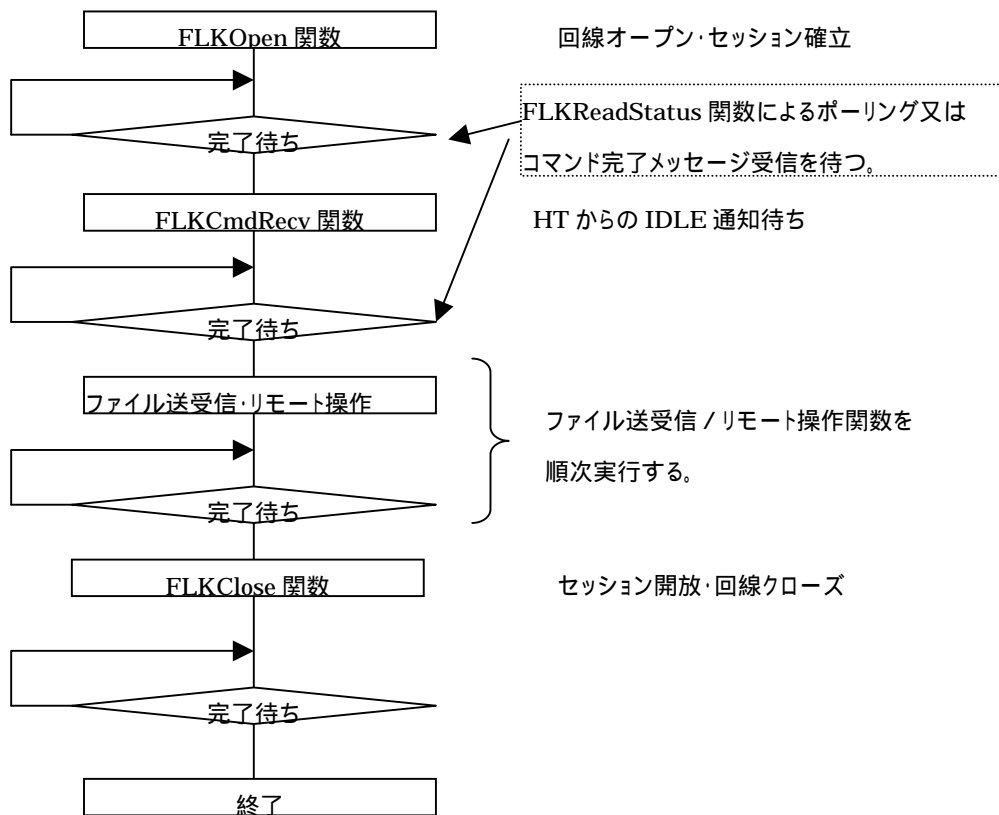


(2) PCコマンド送信による通信

【HT モード基本フロー】



【PCモード基本フロー】



8-13-2. ファイル送受信関数

相手局とのファイル転送(送信、追加、受信)を行うための関数です。

送信権局はファイル送信、追加及び受信コマンドを使用して、相手局とのファイル転送を実現します。

被送信権局は、IDLE 状態(HTモード)、PC モードコマンド待ち状態(PCモード)にて相手からのコマンドを受け付けます。

なお、ライブラリには、ファイル転送時の進捗表示を行う機能はありません。

アプリケーションで必要に応じて表示を行ってください。

8-13-2-1. ファイル送信(FLKFileSend 関数)

複数ファイルの送信を一括して行います。

送信先に指定ディレクトリが存在しない場合は、自動的に作成します。

送信ファイルに対して以下のオプションを選択することができます。

(a)リードオンリーファイル強制上書きオプション

送信ファイルが、既に受信側にリードオンリーファイルとして存在していた場合、強制的に上書きすることができます。

この指定が無い場合にリードオンリーファイルへの書き込みを行うと、エラーとなります。

(b)再帰呼び出し指定オプション

送信ファイルパス名で指定されたディレクトリ傘下の全てのファイルが転送対象となります。

指定ディレクトリ傘下にサブディレクトリが存在した場合はサブディレクトリ名を付加してファイルの送信を行います。

(例)

[送信ファイル名]	[送信先ディレクトリ名]
"¥SEND¥AAA.DAT"	C:¥RECV¥"

(送信側ディレクトリ構成)

¥---SEND¥---SUB1¥-----AAA.DAT	C:¥---RECV¥---SUB1¥-----AAA.DAT
-----SUB2¥-----BBB.DAT	-----AAA.DAT
-----AAA.DAT	
-----BBB.DAT	

(c)ワイルドカードの使用

送信ファイル名にはワイルドカード(*,?)を使用することができます。

8-13-2-2. ファイル追加関数(FLKFileAdd 関数)

HT側ファイルを相手局側ファイルにアペンドすることができます。

相手局側に指定されたアペンドファイルが存在しない場合は、新規作成となります。

複数ファイル及びワイルドカードの指定はできません。

8-13-2-3. ファイル追加関数 (FLKFileAdd 関数)

複数ファイルの受信を一括して行うことができます。

受信ファイルに対して以下のオプションを選択することができます。

(a) リードオンリーファイル強制ライトオプション

受信ファイルが、既に受信側にリードオンリーファイルとして存在していた場合でも、上書きすることができます。この指定が無い場合にリードオンリーファイルへの書き込みを行うと、エラーとなります。

(b) 再帰呼び出し指定オプション

受信ファイルパス名で指定されたディレクトリ傘下の全てのファイルが転送対象となります。

指定ディレクトリ傘下にサブディレクトリが存在した場合はサブディレクトリ名を付加してファイルの受信を行います。

(c) ワイルドカードの使用

受信ファイル名にはワイルドカード(*,?)を使用することができます。

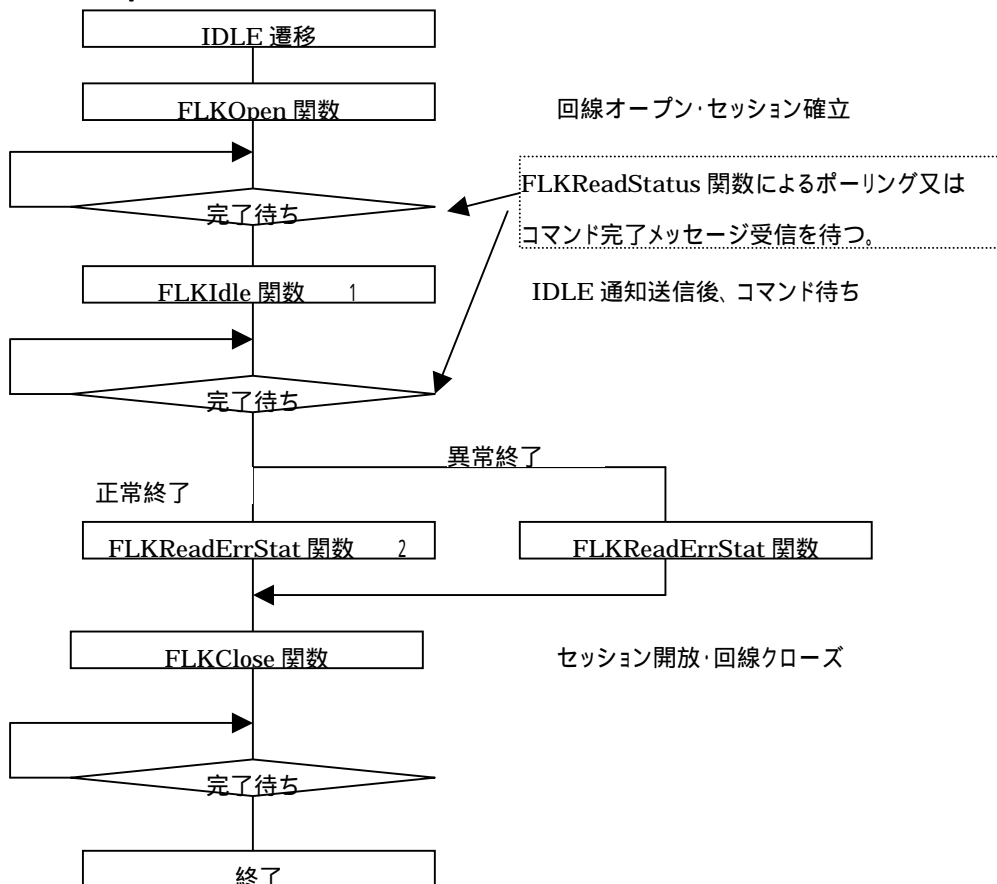
8-13-3. アイドル通知 (FLKIdle 関数)

相手局側に送信権を渡し、コマンド待ち状態となります。

終了指示コマンドを受信またはエラー発生まで、受信したコマンドを順次実行します。

また、PCのスクリプトファイルの実行を指示することができます

[IDLE 遷移基本フロー]



* 1…相手局からの終了指示コマンド受信又はエラー発生まで、受信コマンドを順次実行する。

* 2…必要に応じて相手局からの終了指示コマンド詳細情報(フォーマット指示、リセット指示等)の取得が可能。

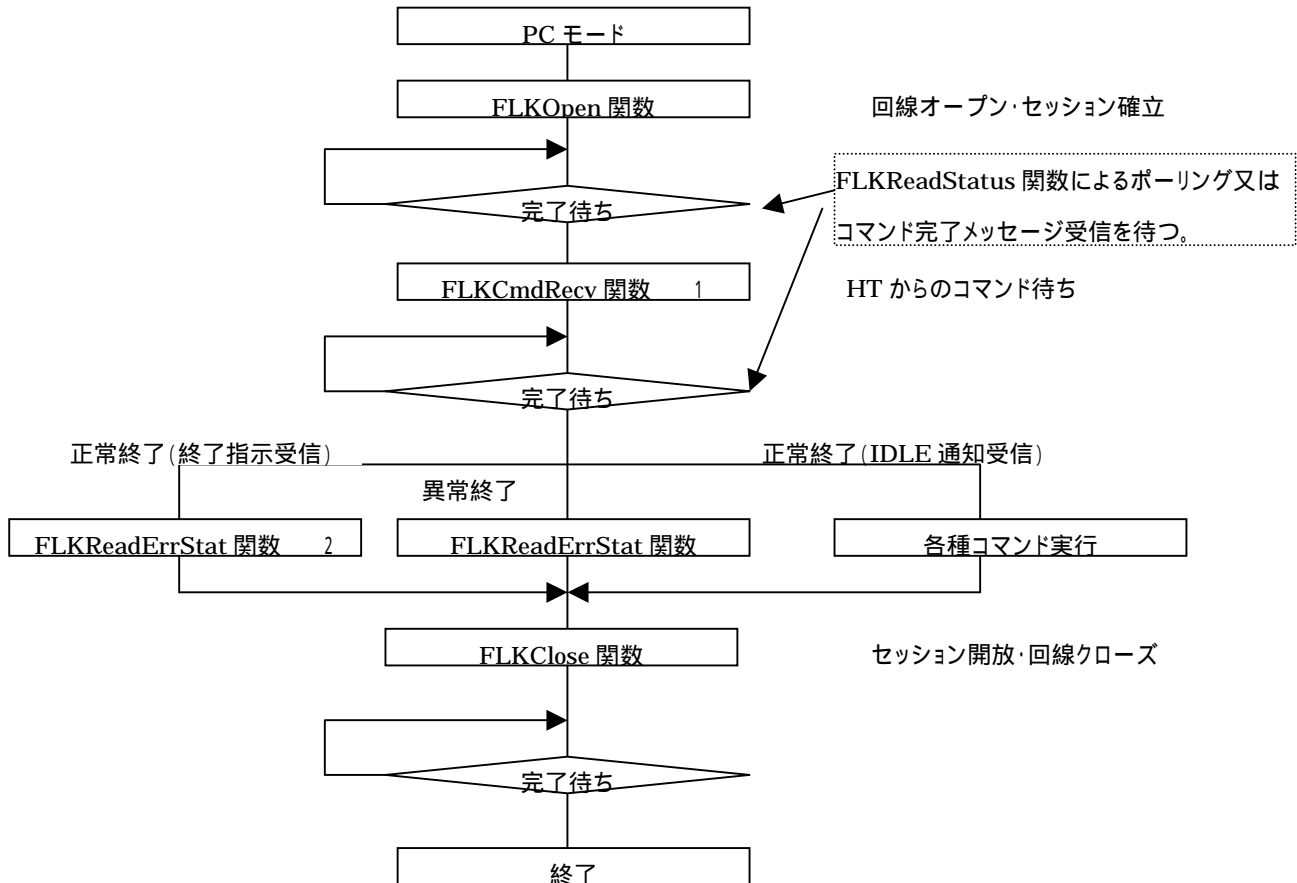
8-13-4. PCモードコマンド待ち(FLKCmdRecv 関数)

HTからのコマンド受信待ち状態となります。PCモードでのみ使用可能です。

セッション確立直後は、HT側に送信権があるため、PCモードではオープン直後にこの関数を用いてコマンド待ちとなる必要があります。

終了指示コマンドか、IDLE通知コマンドを受信またはエラー発生まで、受信したコマンドを順次実行します。

[PCモード時の基本フロー]



- 1……HTから終了指示コマンドまたはIDLE通知コマンドを受信するか、エラーが発生するまで、受信コマンドを順次実行する。
- 2……必要に応じて相手局からの終了指示コマンド詳細情報(フォーマット指示、リセット指示等)の取得が可能。

8-13-5.リモート操作関数

相手局側のファイル操作、環境情報の取得 / 設定を行います。

相手局のファイル / ディレクトリ情報の取得及び設定、相手局上でのファイル操作を行うための関数です。

8-13-5-1. ファイルディレクトリ削除 (FLKFileDelete 関数)

相手局側ファイル及びディレクトリの削除を行います。

複数ファイル指定、ワイルドカード使用が可能です。

指定ファイルが存在しない場合でもエラーになりません。

8-13-5-2. ファイル移動 (FLKFileMove 関数)

相手局側ファイルの同一ドライブ内での移動またはファイル名の変更を行います。

複数ファイル指定、ワイルドカード使用はできません。

移動先ディレクトリが存在しない場合は自動的に作成します。

移動元と移動先のドライブ名が異なる場合はエラーとなります。

8-13-5-3. ディレクトリ作成 (FLKMakeDir 関数)

相手局側ディレクトリ作成を行います。

複数ファイル指定、ワイルドカード使用はできません。

タイムスタンプ、属性の設定が可能です。

8-13-5-4. ファイル情報の取得 (FLKGetFileInfo 関数)

相手局のファイル情報(タイムスタンプ、サイズ、属性)の取得を行います。

ワイルドカード使用が可能です。

8-13-5-5. ファイル情報の設定 (FLKSetFileInfo 関数)

相手局のファイル情報(タイムスタンプ、サイズ、属性)の更新を行います。

8-13-6. 相手局環境情報取得 / 設定関数

相手局のシステム環境情報の取得及び設定を行うための関数です。

8-13-6-1. 日付時刻の取得 / 設定 (FLKDateTime 関数)

相手局のシステム日付時刻の取得 / 設定を行います。

8-13-6-2. ディスク情報の取得 (FLKGetDiskInfo 関数)

相手局側ディスク情報の取得を行います。

ディスク情報の項目は以下の通りです。

- ・ディスク総容量
- ・ディスク空き容量
- ・ディスク状態 (フォーマット済み / 未フォーマット / ディスク無し)

8-13-6-3. システム情報の取得 (FLKGetSysInfo 関数)

相手局側のシステム情報の取得を行います。

システム情報の項目は以下の通りです。

- ・セッションID (通信時のセッション番号)
- ・プロトコルバージョン (ファイル転送プロトコルのバージョン番号)
- ・相手局機種コード (HT / PC (AT互換機) / PC (98シリーズ))
- ・OSモデル情報 (HTモデル種別 / PCのOS種別)

*なお、上記情報は回線オープン時のセッション確立直後に相手局より取得します。

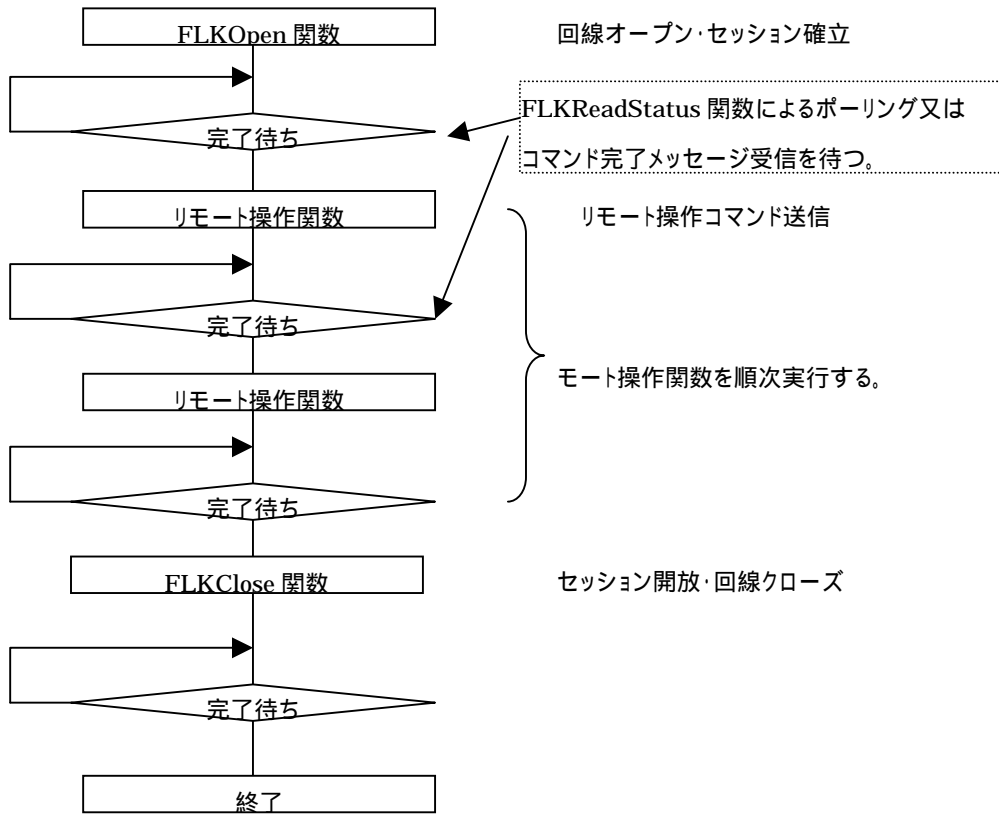
8-13-6-4. 画面表示メッセージの送信 (FLKMsgSend 関数)

相手局へ画面表示用のメッセージを送信します。

8-13-6-5. ブザー鳴動要求の送信 (FLKBeep 関数)

相手局へブザー鳴動コマンドを送信します。

[リモート操作関数の基本フロー]



8-13-7. 通信状態の通知

8-13-7-1. 通信状態の通知方法

本ライブラリの関数は、関数実行後すぐに呼び出し元に制御が戻ります。

実行した関数の機能が正常終了したかどうかは、ライブラリからのメッセージ通知、又は、FLKReadStatus 関数により状態を読み出して判断します。

8-13-7-2. 進捗表示

本ライブラリには、進捗状態を表示する機能はありません。

通信状態が変化した場合に、メッセージ通知するので、アプリケーション側で必要に応じて進捗情報を表示してください。

また、FLKReadStatus 関数を実行することで、現在の進捗状況をいつでも読み出すことができます。

8-13-7-3. メッセージの種類

以下のタイミングで、本ライブラリから指定された Window にメッセージを通知します。

メッセージ番号は、FLKOpen 関数のパラメータで指定します。

メッセージの WPARAM の値	種類	備考
FLK_COMMAND_END	実行したコマンドが正常終了した。	次のコマンドを実行できます。
FLK_COMMAND_ERROR	実行したコマンドがエラー終了した。	クローズを実行してください。
FLK_STATUS_CHANGE1	通信状態が変化した。	FLKReadStatus 関数で状態を読み出すことで、現在実行中の処理を知ることができます。
FLK_STATUS_CHANGE2	ファイル転送中に、ファイルの転送バイト数に変化があった。	FLKReadStatus 関数で状態を読み出すことで、転送中のファイルの進捗情報を知ることが出ます。

8-14. パス名の記述方法

パス名の記述は、指定したいパスが存在するマシンの OS のパス名の命名規則に従ってください。

パスを表現する場合、ドライブレターについて以下に従った記述をしてください。

1. HT 上のパス名はドライブレターを用いず、ルートディレクトリから記述してください。
(ホスト PC 用のアップダウンユーティリティから HT 上のファイルまたはディレクトリのパスを指定する場合も同様です)
2. もし通信相手から、ドライブレター付で指定されても、本ライブラリではドライブレターを無視します。
(ルートディレクトリからドライブレターなしで指定されたものと同等に扱われます)
3. HT からドライブレターの必要な OS の動作している通信相手 (PC 等) 上のファイルまたはディレクトリのパスを表現する場合は、ドライブレターをつけて記述してください。

ただし、上記 2. にかかわらず例外として、通信相手から本機のデバイスを指定して、フォーマット、ディスク情報取得を行う場合、ドライブレターに以下の意味をもたせます。ただし、これらの設定はレジストリに記述することで変更可能です。

デフォルト設定	内部 RAM	C:
	CF カード	D:
	SD カード	E:
	FlashDisk	F:

8-15. レジストリ

レジストリに値を書き込むことにより以下の設定を変更することができます。

レジストリは、オープン時に読み込みます。

一部のレジストリは、FLCE.EXE と同じ物を使用します。

設定項目

- ・セッション確立タイムアウト レジストリに設定がない場合 1800 秒
- ・受信待ちタイムアウト レジストリに設定がない場合 30 秒
- ・セッション終了待ちタイムアウト レジストリに設定がない場合 3 秒
- ・Windows98 通信モード レジストリに設定がない場合は 0
- ・ドライブレターの意味付け対応 レジストリに設定がなかった場合のデフォルト
 - C ¥(内部 RAM のオブジェクトストア)
 - D ¥Storage Card(ストレージカード)
 - E ¥Storage Card2(ストレージカード2)
 - F ¥FlashDisk¥(NAND FLASH)

・レジストリ位置 ¥HKEY_CURRENT_USER¥Software¥CASIO¥FLCE¥

内容

キー名	型	値
CONWAIT	DWORD	セッション確立タイムアウト時間
RECVWAIT	DWORD	受信タイムアウト時間
DISCONWAIT	DWORD	セッション終了タイムアウト時間
WIN98MODE	DWORD	0(1を設定すると Windows98 通信モード)
DRIVE¥A	STRING	A ドライブとして意味付けするデバイスのパス
DRIVE¥B	STRING	B ドライブとして意味付けするデバイスのパス
DRIVE¥C	STRING	C ドライブとして意味付けするデバイスのパス
：		
DRIVE¥Z	STRING	Z ドライブとして意味付けするデバイスのパス