



CUnet (MKY43) 搭載 USB ユニット

CUB-43USB-NFS

ユーザーズマニュアル

ご注意

1. 本書に記載された内容は、将来予告なしに変更する場合があります。本製品をご使用になる際には、本書が最新の版数であるかをご確認ください。
2. 本書において記載されている説明や回路例などの技術情報は、お客様が用途に応じて本製品を適切にご利用をいただくための参考資料です。実際に本製品をご使用になる際には、基板上における本製品の周辺回路条件や環境を考慮の上、お客様の責任においてシステム全体を十分に評価し、お客様の目的に適合するようシステムを設計してください。当社は、お客様のシステムと本製品との適合可否に対する責任を負いません。
3. 本書に記載された情報、製品および回路等の使用に起因する損害または特許権その他権利の侵害に関して、当社は一切その責任を負いません。
4. 本製品および本書の情報や回路などをご使用になる際、当社は第三者の工業所有権、知的所有権およびその他権利に対する保証または実施権を許諾致しません。
5. 本製品は、人命に関わる装置用としては開発されておりません。人命に関わる用途への採用をご検討の際は、当社までご相談ください。
6. 本書の一部または全部を、当社に無断で転載および複製することを禁じます。

はじめに

本マニュアルは、CUnet 専用 IC の一品種である MKY43 を搭載した USB ユニットの CUB-43USB-NFS について記述します。

CUB-43USB-NFS の利用および本マニュアルの理解に先駆けて、“CUnet 導入ガイド”を必ずお読みください。

●対象読者

- ・ CUnet を初めて構築する方
- ・ CUnet を構築するために、弊社の CUB-43USB-NFS を初めてご利用になる方

●読者が必要とする知識

- ・ ネットワーク技術に関する標準的な知識
- ・ 半導体製品（特にマイクロコントローラおよびメモリ）に関する標準的な知識

●関連マニュアル

- ・ CUnet 導入ガイド
- ・ CUnet テクニカルガイド
- ・ CUnet MKY43 ユーザーズマニュアル

【注意事項】

本書において記載されている一部の用語は、弊社の Web および営業用ツール（総合カタログ等）において記載されている用語とは異なっています。営業用ツールにおいては、様々な業界において弊社製品をご理解いただけるよう、一般的用語を用いています。

CUnet ファミリに関する専門知識は、技術ドキュメント（マニュアル等）を基にご理解ください。

改定履歴

Ver	日付	改定内容	
		ページ	説明
Ver1.0J	2015/7/23	-	初版発行
Ver.2.0J	2016/1/14	-	複数ボード対応のためボード ID 機能追加 ・ SW2 にボード ID 機能割り付け ・ メモリマップ変更 Board ID Register 追加 ・ API に「CubBoardID」と「CubSearchBoard」追加 ・ API 「CubOpenHandle」修正 ・ API バージョン番号変更 メジャー番号 =2、マイナー番号 =0、アップデート番号 =0 ・ サンプルプログラム変更

目次

第1章 製品概要

1.1 特徴	1-1
1.2 仕様	1-1

第2章 ハードウェア

2.1 LED、スイッチ、コネクタ	2-1
2.2 メモリマップ	2-3
2.2.1 MKY43	2-3
2.2.2 Board ID Register	2-3

第3章 ソフトウェア

3.1 概要	3-1
3.2 著作権・免責	3-1
3.3 ファイルの種類	3-2
3.4 API仕様	3-2
3.4.1 CubGetVersion	3-3
3.4.2 CubGetLastError	3-4
3.4.3 CubCountDevice	3-5
3.4.4 CubBoardID	3-5
3.4.5 CubSearchBoard	3-6
3.4.6 CubStartAutoTrans	3-7
3.4.7 CubStopAutoTrans	3-8
3.4.8 CubOpenHandle	3-8
3.4.9 CubCloseHandle	3-9
3.4.10 CubReadWord	3-10
3.4.11 CubWriteWord	3-10
3.4.12 CubReadProtect	3-11
3.4.13 CubWriteProtect	3-11
3.4.14 CubReadGM	3-12
3.4.15 CubReadMFR	3-12
3.4.16 CubReadData	3-13
3.4.17 CubWriteData	3-13

第4章 付録

4.1 サンプルプログラム	4-1
---------------------	-----

図 目 次

図 2-1 LED、スイッチ、コネクタ配置図	2-1
図 2-2 CUnet 周辺回路図.....	2-2

表 目 次

表 1-1 仕様.....	1-1
表 2-1 LED 詳細.....	2-1
表 2-2 スイッチ詳細.....	2-2
表 2-3 コネクタ詳細.....	2-2
表 2-4 メモリマップ.....	2-3
表 2-5 Board ID Register 構成.....	2-3
表 3-1 API 一覧.....	3-2
表 3-2 バージョン番号の構成.....	3-3
表 3-3 エラーコードリスト.....	3-4
表 3-4 送信周期の設定リスト.....	3-7

第 1 章 製品概要

本章は、本製品（CUB-43USB-NFS）の製品概要について記述します。

1.1 特徴

CUB-43USB-NFS は、パソコンを CUnet のネットワークに接続するための製品です。ステップテクニカ提供の Windows 用のライブラリと併せて利用することにより、MKY43 の機能を簡単に利用することが可能です。パソコンとの接続は、USB にて接続します。

CUB-43USB-NFS には 8pin のモジュラコネクタを採用しており、100BASE-TX 用市販の CAT-3 以上のストレートケーブルで CUnet の動作を評価することができます。CUB-43USB-NFS の利用経験は、MKY43 を搭載したマイコンシステムにも活かす事ができます。

1.2 仕様

CUB-43USB-NFS の仕様を、表 1-1 に示します。

表 1-1 仕様

ボード名称	CUB-43USB-NFS
CUnet デバイス	MKY43 1 個
CUnet 通信方式	半二重通信
CUnet 通信速度	12M/6M/3Mbps (MKY43 レジスタにて設定)
コネクタ	MicroUSB コネクタ (PC 通信用、電源供給) CUnet 通信コネクタ (RJ-45 タイプ) 2 個
対応 OS	Windows8 (32bit、64bit) Windows7 (32bit、64bit)
電源	+5.0V (MicroUSB コネクタから供給)
付属品	USB ケーブル
消費電流	DC+5.0V ± 5% 400mA 以下
動作環境	温度 0 ~ 55℃ 湿度 0 ~ 90% (非結露)
保存環境	温度 0 ~ 80℃ 湿度 0 ~ 90% (非結露)
外形寸法	80mm × 100mm
最大同時使用台数	4 枚
提供ソフトウェア	Windows ドライバ ライブラリ CUeditor43J

第2章 ハードウェア

本章は、CUB-43USB-NFS のハードウェアについて記述します。

2.1 LED、スイッチ、コネクタ

CUB-43USB-NFS の LED、スイッチ、コネクタ配置図を図 2-1 に示します。

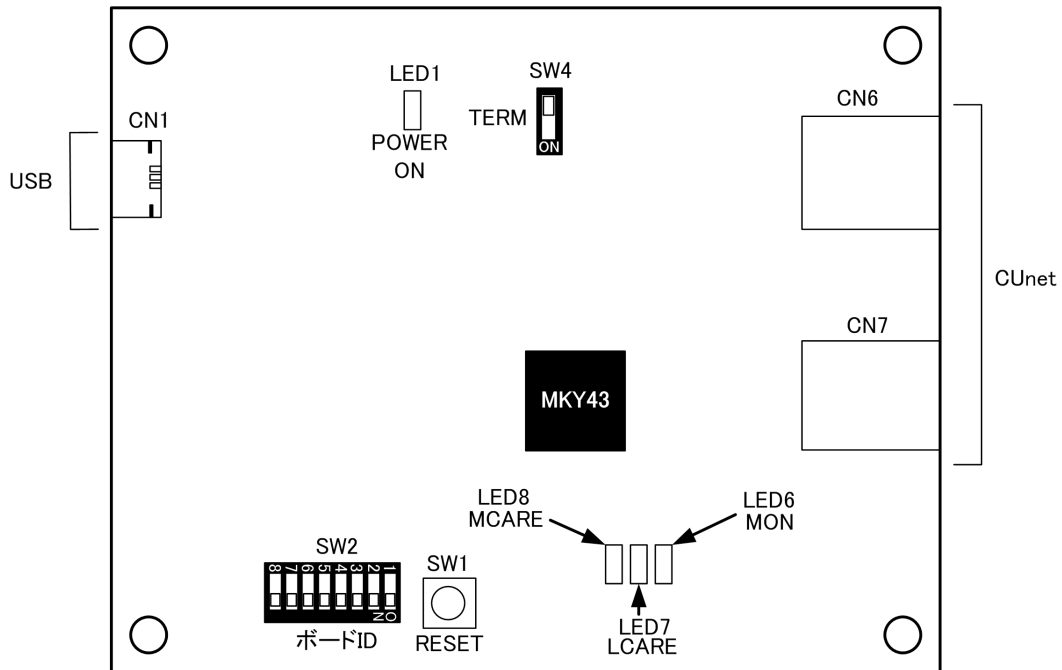


図 2-1 LED、スイッチ、コネクタ配置図

LED 詳細を表 2-1、スイッチ詳細を表 2-2、コネクタ詳細を表 2-3 に示します。

表 2-1 LED 詳細

名 称	LED No.	機能
POWER ON	LED1	電源 "ON" の状態であることを示します。ホストパソコンから USB のバスパワーが正しく供給されている時、緑色点灯します。
MON	LED6	他の CUnet ステーションとリンクしている時、緑色点灯します。
LCARE	LED7	LCARE が発生した時、黄色点灯します。
MCARE	LED8	MCARE が発生した時、赤色点灯します。

表 2-2 スイッチ詳細

名称	SW No.	機能															
RESET	SW1	本スイッチ押下により、本ボード全体のリセットを行います。															
ボード ID	SW2	<p>ボード ID を設定します。SW2-3 ~ SW2-8 は未使用です。 (出荷時のボード ID は、0 です。)</p> <table border="1"> <thead> <tr> <th>ボード ID</th> <th>SW2-2</th> <th>SW2-1</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>OFF</td> <td>OFF</td> </tr> <tr> <td>1</td> <td>OFF</td> <td>ON</td> </tr> <tr> <td>2</td> <td>ON</td> <td>OFF</td> </tr> <tr> <td>3</td> <td>ON</td> <td>ON</td> </tr> </tbody> </table>	ボード ID	SW2-2	SW2-1	0	OFF	OFF	1	OFF	ON	2	ON	OFF	3	ON	ON
ボード ID	SW2-2	SW2-1															
0	OFF	OFF															
1	OFF	ON															
2	ON	OFF															
3	ON	ON															
TERM	SW4	CUNet インターフェースが 100 Ω で終端されます。CUB-43USB-NFS が、マルチドロップ接続の末端位置に存在する場合は、SW4 を ON にしてターミネーションを有効にしてください。(出荷時は OFF: 無効状態です。)															

表 2-3 コネクタ詳細

名称	CN No.	機能										
USB	CN1	ホストパソコンとの通信用 MicroUSB										
CUNet	CN6,CN7	<p>CUNet 接続用 CN6 と CN7 は同ピン配列です。</p> <table border="1"> <thead> <tr> <th>ピン番号</th> <th>信号名</th> </tr> </thead> <tbody> <tr> <td>4</td> <td>TRX-</td> </tr> <tr> <td>5</td> <td>TRX+</td> </tr> <tr> <td>5</td> <td>シールド</td> </tr> <tr> <td>1、2、3、6、7</td> <td>未使用</td> </tr> </tbody> </table>	ピン番号	信号名	4	TRX-	5	TRX+	5	シールド	1、2、3、6、7	未使用
ピン番号	信号名											
4	TRX-											
5	TRX+											
5	シールド											
1、2、3、6、7	未使用											

CUNet 周辺回路図を図 2-2 に記します。

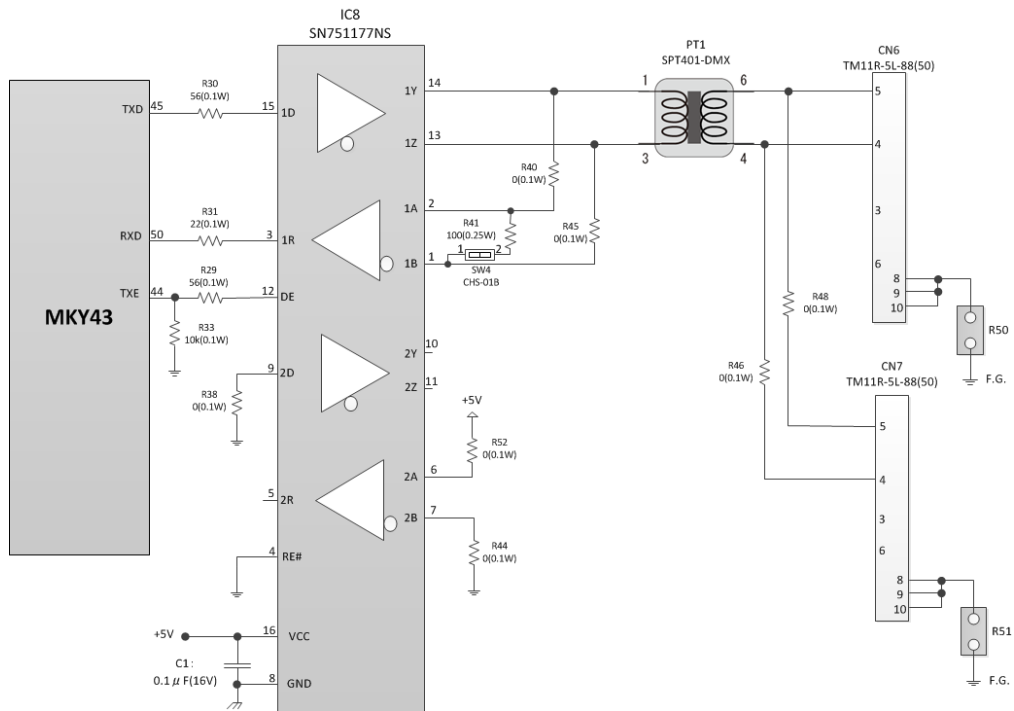


図 2-2 CUNet 周辺回路図

2.2 メモリマップ

CUB-43USB-NFS のメモリマップを表 2-4 に示します。メモリマップ中のアドレスは CUB-43USB-NFS の先頭アドレスからの相対値であり、本製品の API を使用して MKY43 アクセスする場合は、このアドレス情報に基づいてアクセスすることが可能です。

表 2-4 メモリマップ

アドレス	機能
000H ~ 5FFH	MKY43
600H ~ F01H	未使用
F02H	Board ID Register

2.2.1 MKY43

CUB-43USB-NFS 上の MKY43 は、表 2-4 に示す通り、000H ~ 5FFH にマッピングしています。MKY43 のメモリマップについては「MKY43 ユーザーズマニュアル」をご参照ください。

2.2.2 Board ID Register

BID0,1 (Board ID) ビットをリードすることにより、SW2-2、SW2-1 によって設定されたボード ID を取得することができます。

表 2-5 Board ID Register 構成

bit	15 ~ 2	1	0
R/W	R	R	R
機能	—	BID1	BID0

表 2-6 ボード ID と BID0,1 の関係

ボード ID	BID1	BID0
0	0	0
1	0	1
2	1	0
3	1	1



注意事項

表 2-4 メモリマップに示されている未使用領域(“600H ~ F01H”)はアクセスしないでください。システムを不安定にする可能性があります。

第3章 ソフトウェア

本章では、ステップテクニカ社提供の API について記述します。

3.1 概要

Windows 上のユーザアプリケーションからの CUB-43USB-NFS へのアクセスを簡略化するために DLL を用意しています。

下記ステップテクニカ社のダウンロードページより DLL をダウンロードできます。

URL : <http://www.steptechnica.com/jp/download/index.html>

対応 OS は

- Windows 8 (64bit/32bit)
- Windows 7 (64bit/32bit)

になります。

提供している DLL は、Microsoft Visual Studio や VB6 などから呼び出すことが可能となっています。

以下から DLL に収録されている API 関数の説明を記します。



参考

対応 OS や、最新のソフトウェア情報は、弊社 web サイト (<http://www.steptechnica.com>) をご確認ください。

3.2 著作権・免責

ステップテクニカ社が提供している、全てのドキュメント・プログラム・プログラムソースの著作権は、株式会社ステップテクニカが所有しています。株式会社ステップテクニカは、以下の注意事項を了承された個人・法人、または、その他の団体が弊社製品 CUB-43USB-NFS を利用する場合に限り、これら著作物の複製・利用をする権利をライセンスするものであり、株式会社ステップテクニカに断り無く、これら著作物の一部または全部を改訂・再配布や上記以外の目的のために複製・利用することはできません。



注意事項

- ① 本製品付属ディスク内のソフトウェア及び、弊社 web ページより入手した全てのソフトウェアの使用による、いかなる結果に対しても弊社は一切責任を負いません。
- ② ライブラリは、説明に従って正しくお使いください。
- ③ 仕様・内容は、将来予告無く変更になる場合があります。弊社は、将来への互換性について、一切保証いたしません。
- ④ 弊社製品以外の OS や開発環境等に関するお問い合わせはサポートいたしかねます。
- ⑤ バグ・不具合などを発見された方は、弊社システム開発部までご連絡ください。

3.3 ファイルの種類

"DLL" フォルダに収められているファイルは以下の通りです。

【DLL】

- |
- + ---- [cub43usbfnfs.dll] : DLL 本体です。Windows のシステムフォルダ、もしくは、本 DLL を使用する
| ユーザプログラムと同じディレクトリにコピーしてお使いください。
- + ---- [cub43usbfnfs.lib] : インポートライブラリです。
- + ---- [cub43usbfnfs.h] : DLL のヘッダファイルです。Windows.h より後ろにインクルードして下さい。

3.4 API 仕様

CUBAPI の仕様について記述します。

表 3-1 API 一覧

関数	機能概要
CubGetVersion	CUBAPI のバージョン番号取得
CubGetLastError	CUBAPI 関数の終了状態を取得
CubOpenHandle	CUB-43USB-NFS への接続を識別するハンドルを生成
CubCloseHandle	CubOpenHandle で生成されたハンドルをクローズ
CubCountDevice	接続された CUB-43USB-NFS の台数を取得
CubBoardID	ボード ID を取得
CubSearchBoard	CUB-43USB-NFS の台数とそのボード ID を取得
CubStartAutoTrans	定期送信の開始
CubStopAutoTrans	定期送信の停止
CubReadWord	CUB-43USB-NFS から 2 バイトデータ読み込み
CubWriteWord	CUB-43USB-NFS へ 2 バイトデータ書き込み
CubReadProtect	グローバルメモリからハザード防止機能を使用した読み込み
CubWriteProtect	グローバルメモリへハザード防止機能を使用した書き込み
CubReadGM	定期送信機能により送られた全グローバルメモリを取得
CubReadMFR	定期送信機能により送られた MFR を取得
CubReadData	グローバルメモリの指定されたアドレスから指定ワード長のデータ読み込み
CubWriteData	グローバルメモリの指定されたアドレスから指定ワード長のデータ書き込み

3.4.1 CubGetVersion

書式

UINT CubGetVersion(void);

機能

API のバージョン番号を取得します。

パラメータ

なし

リターンパラメータ

API のバージョン番号を表す数値 (BCD コード 16 進数)
(メジャー番号 + マイナー番号 + アップデート番号)

エラーコード

CUB_SUCCESS 正常終了

注意

必須ではありませんが、cub43usbfnfs.dll を利用するユーザアプリケーションにおいて DLL に対する互換性チェックを行うことで安全性を高めることができます。ここでいう安全性とは、互換性の無い関数コールを予め避けることでプログラムの強制終了などを避けることを意味します。CubGetVersion は DLL バージョンを取得する API です。この API の返す数値のバージョン番号には表 3-2 のような構成になっています。それぞれの番号がアップされる原因は以下の通りです。

メジャー番号：API の仕様変更など、互換性を保てなかった変更がなされた時に変わります。

マイナー番号：API の追加など、下位互換を保ったままの変更がなされた時に変わります。

アップデート番号：バグ修正など、仕様上の変更が全くない変更がなされた時に変わります。

マイナー番号やアップデート番号の数字は互換上無視しても構いませんがメジャー番号の値が変わっている場合は、API をコールしないことをお奨めします。この互換性チェックは、初期化処理よりも先に行う必要があります。

表 3-2 バージョン番号の構成

戻り値 (例)	メジャー番号 (ビット 15 ~ 8)	マイナー番号 (ビット 7 ~ 4)	アップデート番号 (ビット 3 ~ 0)
0x0102	1	0	2
0x1398	13	9	8

3.4.2 CubGetLastError

書式

UINT CubGetLastError(void);

機能

最後に呼び出された CUBAPI 関数の終了状態を取得します。

パラメータ

なし

リターンパラメータ

cub43usbdfs.h で定義されているエラーコードを返します。

注意

表 3-3 に cub43usbdfs.h で定義されているエラーコードを記します。

表 3-3 エラーコードリスト

文字定数	値	内容
CUB_SUCCESS	0	正常終了
CUB_ERR_DEVICENOTEXIST	1	デバイスが存在しない
CUB_ERR_ALREADYOPENED	2	すでにオープンされている
CUB_ERR_CLOSED	3	CubOpenHandle が一度もコールされていない
CUB_ERR_INVALIDPARAM	4	無効なパラメータでコールされた
CUB_ERR_NORESOUCE	5	実行に必要なリソースが足りない
CUB_ERR_FAILED	6	原因不明により処理が遂行されなかった
CUB_ERR_AUTO_TRANS_ALREADY_START	7	定期送信がすでに開始されている
CUB_ERR_AUTO_TRANS_STOP	8	定期送信が開始されていない
CUB_NOTCALLYET	99	CUBAPI が一度もコールされていない

3.4.3 CubCountDevice

書式

```
INT CubCountDevice(void);
```

機能

接続されている CUB-43USB-NFS の台数を返します。同一 PC 上に 5 台以上の台数を認識できません。

パラメータ

なし

リターンパラメータ

CUB-43USB-NFS の台数を返します。

-1	: 5 台以上
0	: 1 台も接続されていない
1 ~ 4	: 1 ~ 4 台

エラーコード

CUB_SUCCESS	正常終了
-------------	------

3.4.4 CubBoardID

書式

```
INT CubBoardID(HANDLE CUBHandle);
```

機能

ハンドル値で指定された CUB-43USB-NFS のボード ID を取得します。

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS へのハンドル値をセットする。
------------------	------------------------------------

リターンパラメータ

正常終了時はボード ID (0 ~ 3) を返します。失敗時は -1 を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.5 CubSearchBoard

書式

```
BOOL CubSearchBoard(BYTE *board_num, BYTE *board_id_list);
```

機能

PC に接続されている CUB-43USB-NFS の台数とそのボード ID リストを返します。

パラメータ

*board_num ボード台数がセットされる変数へのポインタ
セットされた値の意味は以下の通りです。
・ -1 : 5 枚以上確認された
・ 0 : 1 枚もない
・ 1 ~ 4 : 認識したボード枚数

*board_id_list unsigned char 型の要素数 4 つの配列へのアドレスを指定します。
または NULL を指定することも可能です。
NULL が指定された場合は、ボード枚数のみを数えます。
セットされた値の意味は以下の通りです。
・ 0 ~ 3 : ボード ID
・ 0x80 : すでに CubOpenHandle でハンドル値を取得されている
・ 0xFF : 認識できなかった

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

補足

CUB-43USB-NFS には、SW2-2 と SW2-1 にてボード ID を指定できます。

ボード ID を設定することにより CUB-43USB-NFS が複数台接続している場合に個別に識別することが可能です。

本 API では、最大 4 台までの CUB-43USB-NFS を識別します。API 使用時には、

```
BYTE board_num;
```

```
BYTE board_id_list[4];
```

```
CubSearchBoard(&board_num, &board_id_list[0]);
```

とバイト型の配列を宣言してパラメータとしてセットして下さい。

例として、PC に 3 台の CUB-43USB-NFS が接続されており、それぞれのボード ID が
1 台目 ボード ID=0、2 台目 ボード ID=1、3 台目 ボード ID=2 と設定されているとします。

PC が認識した順番が 1、3、2 となっている状況で CubSearchBoard が実行された場合は

```
board_num = 3;
```

```
board_id_list [0] = 0、board_id_list [1] = 2、board_id_list [2] = 1、board_id_list [3] = 0xFF
```

と返します。

3.4.6 CubStartAutoTrans

書式

BOOL CubStartAutoTrans(HANDLE CUBHandle, WORD MfCnt);

機能

CUB-43USB-NFS からの定期送信を開始します。定期送信とは、CUB-43USB-NFS から全グローバルメモリと MFR を指定された送信周期で PC ヘデータ送信を行う機能です。

送信されたデータは、API 内部にて保持されます。保持されたデータは、それぞれ CubReadGM、CubReadMFR にて取得できます。

パラメータ

HANDLE CUBHandle 対象となる CUB-43USB-NFS へのハンドル値をセットする。
WORD MfCnt 定期送信の送信周期をセットします。送信周期は 1ms から 100ms まで 125 μ s 間隔で指定できます。定期送信を行う場合は、表 3-4 を参照して頂き送信周期を決定して下さい。定期送信を行わない場合は、0 をセットして下さい。

表 3-4 送信周期の設定リスト

設定値	送信周期 (μ sec)	設定値	送信周期 (μ sec)
0	定期送信を行わない	:	:
8	1,000 (1msec)	792	99,000 (99msec)
9	1,125	793	99,125
10	1,250	794	99,250
11	1,375	795	99,375
12	1,500	796	99,500
13	1,625	797	99,625
14	1,750	798	99,750
15	1,875	799	99,875
16	2,000 (2msec)	800	100,000 (100msec)

リターンパラメータ

処理結果を返します。正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS 正常終了
CUB_ERR_INVALIDPARAM 無効なパラメータでコールされた
CUB_ERR_AUTO_TRANS_ALREADY_START 定期送信がすでに開始されている
CUB_ERR_FAILED 原因不明により処理が遂行されなかった

注意

PC のスペックや他に動作しているアプリケーションにより CUB-43USB-NFS から定期的に送信されるデータを受信できない可能性がありますので注意ください。

定期通信を行わなくても CubReadGM と CubReadMFR 以外の API は使用できます。

3.4.7 CubStopAutoTrans

書式

```
BOOL CubStopAutoTrans(HANDLE CUBHandle);
```

機能

CUB-43USB-NFS からの定期送信を停止します。

パラメータ

HANDLE CUBHandle 対象となる CUB-43USB-NFS へのハンドル値。

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_AUTO_TRANS_STOP	定期送信が開始されていない
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.8 CubOpenHandle

書式

```
HANDLE CubOpenHandle(int index_no);
```

機能

CUB-43USB-NFS へのハンドル値を取得します。

パラメータ

int index_no インデックス番号
インデックス番号には、0～3が指定できます。
CUB-43USB-NFS が 1 台しかないときは、0 をセットしてください。詳しくは、“補足”を参照してください。

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_ALREADYOPENED	すでにオープンされている
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_DEVICENOTEXIST	デバイスが存在しない
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

補足

CUB-43USB-NFS が 1 台のみ接続されている場合には、CubSearchBoard を実行せずに CubOpenHandle (0); としても問題ありません。
CUB-43USB-NFS が複数台接続されている場合には、“CubSearchBoard” を先に実行し、操作を行う対象の CUB-43USB-NFS を確認しておく必要があります。
例として、PC に 3 台の CUB-43USB-NFS が接続されており、それぞれのボード ID が 1 台目 ボード ID=0、2 台目 ボード ID=1、3 台目 ボード ID=2 と設定されているとします。ここでボード ID=2 のハンドル値を取得するには

```
BYTE board_num;
BYTE board_id_list[4];
CubSearchBoard(&board_num, &board_id_list[0]);
```

と実行し、取得したリストからボード ID が 2 であるインデックス番号を探します。
board_id_list [0] = 0、board_id_list [1] = 2、board_id_list [2] = 1、board_id_list [3] = 0xFF
インデックス番号 1 がボード ID=2 であることが確認できます。
そこで CubOpenHandle のパラメータにインデックス番号 1 をセットし実行します。

```
CubOpenHandle(1);
```

プログラム終了時、CubCloseHandle によりハンドルをクローズしてください。

3.4.9 CubCloseHandle

書式

BOOL CubCloseHandle(HANDLE CUBHandle);

機能

CubOpenHandle によって取得したハンドルを閉じます。
定期送信を行っている場合は、停止処理も一緒に行います。

パラメータ

HANDLE CUBHandle 対象となる CUB-43USB-NFS のハンドル値

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.10 CubReadWord

書式

```
BOOL CubReadWord(HANDLE CUBHandle, const ULONG Adr, WORD *Dat);
```

機能

MKY43 から 2 バイトデータ読み込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
const ULONG Adr	アクセス先アドレス値。2 の倍数を指定する。入力範囲は 0x000 ~ 0xF02。
WORD *Dat	読み込みデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.11 CubWriteWord

書式

```
BOOL CubWriteWord(HANDLE CUBHandle, const ULONG Adr, const WORD Dat);
```

機能

MKY43 へ 2 バイトデータ書き込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
const ULONG Adr	アクセス先アドレス値。2 の倍数を指定する。入力範囲は 0x000 ~ 0xF02。
const WORD Dat	書き込みデータ

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.12 CubReadProtect

書式

```
BOOL CubReadProtect(HANDLE CUBHandle, WORD BlockNo, void *Data);
```

機能

グローバルメモリからハザード防止機能を使用した読み込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
WORD BlockNo	メモリブロック番号。入力範囲は 0 ~ 63。
void *Data	8 バイト読み込みデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.13 CubWriteProtect

書式

```
BOOL CubWriteProtect(HANDLE CUBHandle, WORD BlockNo, void *Data);
```

機能

グローバルメモリへハザード防止機能を使用した書き込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
WORD BlockNo	メモリブロック番号。入力範囲は 0 ~ 63。
void *Data	8 バイト読み込みデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

3.4.14 CubReadGM

書式

```
BOOL CubReadGM(HANDLE CUBHandle, void*Data);
```

機能

定期送信機能により送られた全グローバルメモリを取得します。

定期送信機能が停止中に CubReadGM をコールした場合エラーを返します。

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
void *Data	512 バイトデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_AUTO_TRANS_STOP	定期送信が開始されていない
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

注意

CubReadGM は、直接 MKY43 にアクセスしていません。直接 MKY43 からグローバルメモリを取得する場合には、“CubReadWord” か “CubReadData”、“CubReadProtect” を使用して下さい。

3.4.15 CubReadMFR

書式

```
BOOL CubReadMFR(HANDLE CUBHandle, void *Data)
```

機能

定期送信機能により送られた MFR を取得します。

定期送信機能が停止中に CubReadMFR をコールした場合エラーを返します。

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
void *Data	8 バイトデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_AUTO_TRANS_STOP	定期送信が開始されていない
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

注意

CubReadMFR は、直接 MKY43 にアクセスしていません。直接 MKY43 から MFR を取得する場合には、“CubReadWord” か “CubReadData” を使用して下さい。

3.4.16 CubReadData

書式

```
BOOL CubReadData(HANDLE CUBHandle, WORD Adr, WORD WordLen, void *Data);
```

機能

グローバルメモリの指定されたアドレスから指定ワード長のデータ読み込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
WORD Adr	アクセス先アドレス値。2 の倍数を指定する。入力範囲は、0x0000 ~ 0x07FE。
WORD WordLen	ワード長を指定する。入力範囲は 0x0001 ~ 0x0400。
void *Data	読み込みデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

注意

指定された読み込み範囲が MKY43 の範囲 (0x800) を超えた場合、エラーを返します。

3.4.17 CubWriteData

書式

```
BOOL CubWriteData(HANDLE CUBHandle, WORD Adr, WORD WordLen, void *Data);
```

機能

グローバルメモリの指定されたアドレスから指定ワード長のデータ書き込み

パラメータ

HANDLE CUBHandle	対象となる CUB-43USB-NFS のハンドル値
WORD Adr	アクセス先アドレス値。2 の倍数を指定する。入力範囲は、0x0000 ~ 0x07FE。
WORD WordLen	ワード長を指定する。入力範囲は 0x0001 ~ 0x0400。
void *Data	読み込みデータ格納先へのアドレス

リターンパラメータ

正常終了時は TRUE(1)、失敗時は FALSE(0) を返します。

エラーコード

CUB_SUCCESS	正常終了
CUB_ERR_INVALIDPARAM	無効なパラメータでコールされた
CUB_ERR_FAILED	原因不明により処理が遂行されなかった

注意

指定された書き込み範囲が MKY43 の範囲 (0x800) を超えた場合、エラーを返します。

第4章 付録

4.1 サンプルプログラム

CUB-43USB-NFS をコントロールする為の初期化処理と終了処理のサンプルプログラムを記載します。
サンプルプログラムに記載されている MKY43 レジスタの構成や機能については、
「MKY43 ユーザーズマニュアル」の「第5章 MKY43 のレジスタリファレンス」を参照して下さい。

```
int main(int argc, char argv[])
{
    unsigned char buf[0x580];
    unsigned char board_count;
    unsigned char board_id_list[4];

    /** API のバージョン番号をチェック */
    if (version < 0x200 || version > 0x299) {
        printf(" 互換性の無いバージョンの cub43usbdfs.dll です。 \n");
        exit(1);
    }

    /** CUB-43USB-NFS の検索
    * 最大 4 台の CUB-43USB-NFS を識別できます。4 台以上接続されている場合にはエラーを返します。
    * PC が認識した台数分順番にボード ID が board_id_list へセットされます。
    * 認識した台数分以降は、board_id_list に 0xFF がセットされます。
    * CUB-43USB-NFS が 1 台だけの場合は、本関数を必ずしも呼ぶ必要はありません。
    */
    if (CubSearchBoard(&board_count, &board_id_list[0])) {
        exit(1);
    }

    If (board_count == 0) {
        printf("CUB-43USB-NFS が 1 台もありません。 \n");
        exit(1);
    } else if (board_count == 0xFF) {
        printf("CUB-43USB-NFS が 5 台以上あります。 \n");
        exit(1);
    }

    /** コントロールする CUB-43USB-NFS のハンドルを生成します。
    * CUB-43USB-NFS が 1 台しか接続されていないことがわかっている場合は、引数 0 でオープンします。
    */
    HANDLE dev_handle;
    dev_handle=CubOpenHandle(0);
    if (dev_handle == INVALID_HANDLE_VALUE) {
        exit(1);
    }
    memset(buf, 0, sizeof(buf));
    /** CUNET の初期化 */
    // グローバルメモリのクリア
```



```
CubWriteData(dev_handle, 0, 0x100, buf);

// メール送信バッファのクリア
CubWriteData (dev_handle, 0x200, 0x80, buf);
// メール受信バッファ 0 のクリア
CubWriteData (dev_handle, 0x400, 0x80, buf);
// メール受信バッファ 1 のクリア
CubWriteData (dev_handle, 0x500, 0x80, buf);
// SCR へ 0x8000 をセットして GMM モードへ移行
CubWriteWord (dev_handle, 0x366, 0x8000);
// SA=1,OWN=1,BPS=3Mbps に設定変更
CubWriteWord (dev_handle, 0x356, 0x0141);
// GMM モード解除
CubWriteWord (dev_handle, 0x366, 0);
/** 初期化完了後 通信開始
 * SCR の START bit に "1" をセットしてネットワークを開始します。
 */
CubWriteWord (dev_handle, 0x366, 0x0100);
/** 定期通信開始 (CubReadGM、CubReadMFR を使用しない場合は実行する必要はありません)
 * 3000us(3msec) 周期でデータ送信
 */
CubStartAutoTrans(dev_handle, 32);

/** -- ユーザ処理を記述 -- */

/** 定期通信停止 (定期通信 (CubStartAutoTrans) を使用していない場合は実行する必要はありません) */
CubStopAutoTrans(dev_handle);
/** 生成したハンドルを閉じる */
CubCloseHandle(dev_handle);
return 0;
}
```

■開発・製造

株式会社ステップテクニカ

〒 358-0011 埼玉県入間市下藤沢 757-3

TEL: 04-2964-8804

<http://www.steptecnica.com/>

info@steptecnica.com

**CUnet (MKY43) 搭載 USB ユニット
CUB-43USB-NFS
ユーザーズマニュアル**

ドキュメント No. : STD-CUB43USBNFS_V2.0J

発行年月日 : 2016 年 1 月