STD-CUB-43USB-NFS_V1.0E

STEP TECHNICA CO.,LTD.

CUnet(MKY43)USB Unit

CUB-43USB-NFS

User's Manual

Note

1. The information in this document is subject to change without prior notice. Before using this product, please confirm that this is the latest version of this document.

2. Technical information in this document, such as explanations and circuit examples, are references for this product. When actually using this product, always fully evaluate the entire system according to the design purpose based on considerations of peripheral circuits and the PC board environment. We assume no responsibility for any incompatibility between this product and your system.

3. We assume no responsibility whatsoever for any losses or damages arising from the use of the information, products, and circuits in this document, or for infringement of patents and any other rights of a third party.

4. When using this product and the information and circuits in this document, we do not guarantee the right to use any property rights, intellectual property rights, and any other rights of a third party.

5. This product is not designed for use in critical applications, such as life support systems. Contact us when considering such applications.

6. No part of this document may be copied or reproduced in any form or by any means without prior written permission from StepTechnica Co., Ltd..

Preface

This manual describes the CUB-43USB-NFS,USB Unit on which the MKY43 is mounted, a kind of CUnet.
Be sure to read “CUnet Introduction Guide” before understanding this manual and the CUB-43USB-NFS.

● Target Readers

・Those who first build on CUnet
・Those who first use CUB-43USB-NFS to build on CUnet

● Prerequisites

This manual assumes that you are familiar with:
・Network technology
・Semiconductor products (especially microcontrollers and memory)

● Related Manuals

・CUnet Introduction Guide
・CUnet Technical Guide
・CUnet IC MKY43 User’s Manual

【Caution】

Some terms in this manual are different from those used on our website and in our product brochures.
The brochure uses ordinary terms to help many people in various industries understand our products.
Please understand technical information on CUnet Family based on technical documents (manuals).

CONTENTS

Chapter1　Product Summary

Chapter2　Hardware

Chapter3　Software

Chapter4　Appendix

Figures

Tables

Chapter 1　Product Summary

This chapter describes the CUB-43USB-NFS product summary.

1.1　Features

CUB-43USB-NFS is a product for connecting the PC to the network of CUnet.
By utilizing in conjunction with a library for Windows provided by Step Techinica, it is possible to easily use the functions of MKY43. Connection to the PC, connected by USB.
CUB-43USB-NFS have the 8P8C modular connector interface, and the operation can be evaluated using a straight cable for 100BASE-TX, which is CAT-5 or upper.
The operation experience with CUB-43USB-NFS encourage operating the microcomputer system in which the MKY43 is embedded.

1.2　Specifications

The Specifications of CUB-43USB-NFS is given in Table.1-1.

Table.1-1　Specifications of CUB-43USB-NFS

Type	CUB-43USB-NFS
Type of IC mounted	MKY43 × 1
Communication method	CUnet communication method
Baud rate	12M / 6M / 3Mbps half duplex
CUnet Connector	RJ-45 modular connector : TM11R-5L-88 (50) × 2
Host Interface	USB2.0 High Speed USB.
USB Connector	Micro USB Type B
Supported OS	Windows 8/7(64-bit/32-bit)
Power supply	USB bus power
Accessory	Micro USB cable
Consumption current	DC+5.0V ± 5%　less than 400mA
Conditions of use	Temperature1 0 – 50℃ Humidity 0 － 90% (with no condensation)
Board Size	80mm × 100mm

Chapter 2 Hardware

This chapter describes the CUB-43USB-NFS hardware.

2.1 LED , Swich , Connector

Fig.2-1 is CUB-43USB-NFS of LED, switch, the connector layout.

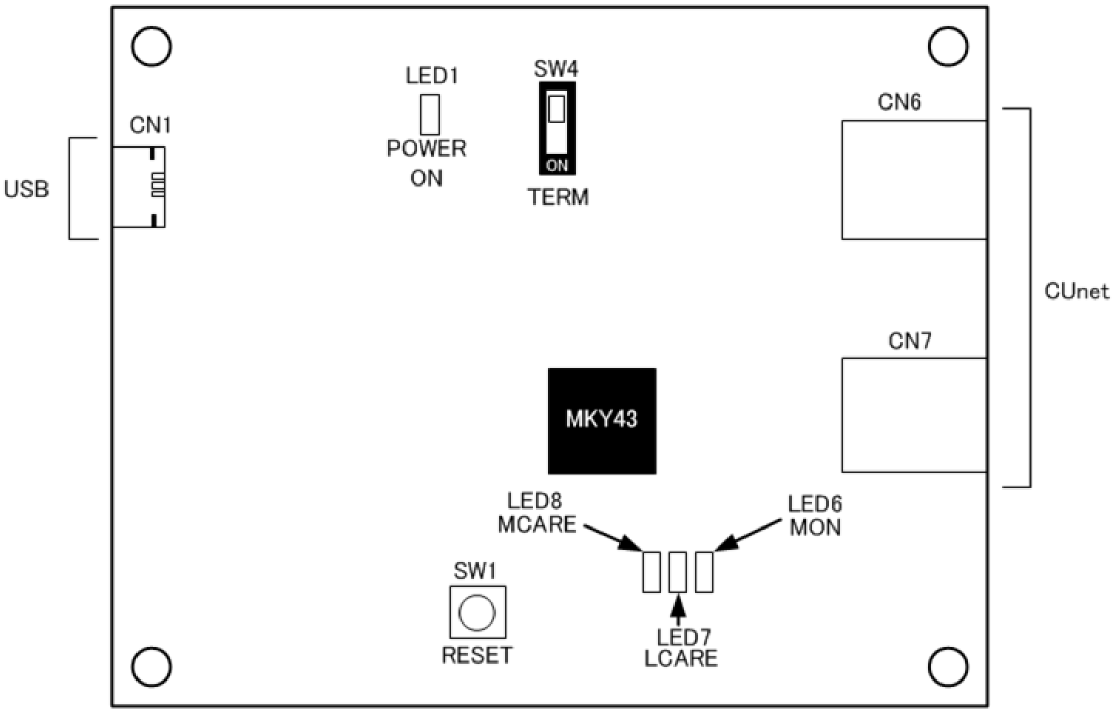

Fig.2-1 Outside view of CUB-43USB-NFS

LED details Table.2-1, Switch details Table.2-2, Connector details Table.2-3.

Table.2-1 LED details

LED	LED	Function
POWER ON	LED1	Indicates state of power supply "ON". When the USB bus power is supplied correctly, green lights.
MON	LED6	When CUnet is linked, green lights.
LCARE	LED7	When LCARE has occurred, yellow lights.
MCARE	LED8	When MCARE has occurred, red lights.

Table.2-2　Switch details

Switch	Switch no.	Function
RESET	SW1	Reset switch of the board. To reset STM32F207, MKY43 and USB3320, press this switch.
TERM	SW4	CUnet interface is terminated with 100 Ω . If CUB-43USB-NFS is at the terminal position of multi-drop connection, (that is, the terminus of communication cable), set SW4 “ON” for termination. (Default setting is OFF. Termination is disable.)

Table.2-3　Connector details

Connector	Connector no.	Function
USB	CN1	USB connector for PC (host) communication.
CUnet	CN6,CN7	CUnet communication connectors. CN6 and CN7 have the same pin assignment. 4　TRX- 5　TRX+ 8　Shield 1,2,3,6,7　not used

Fig.2-2 is CUnet peripheral circuit diagram.

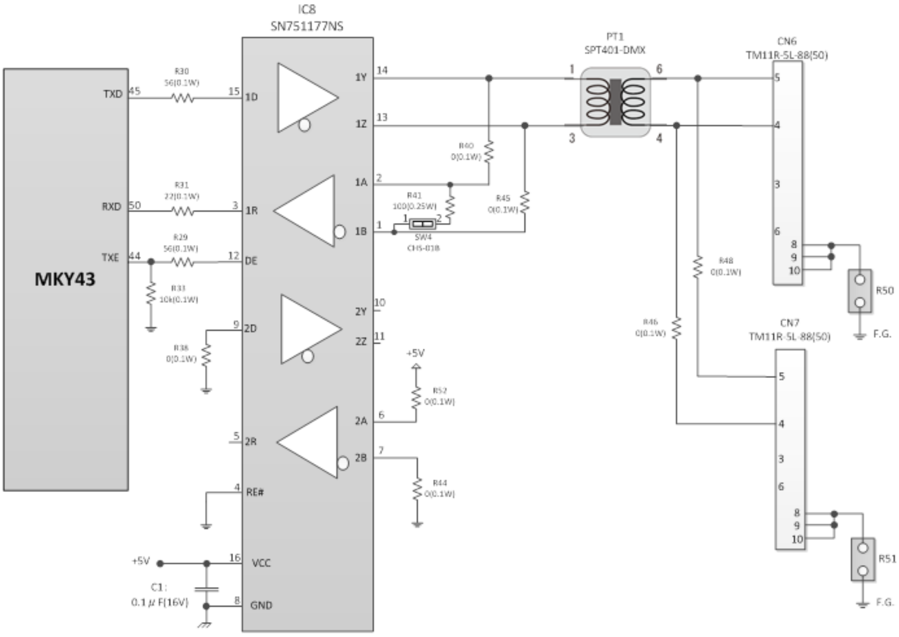

Fig.2-2　CUnet peripheral circuit diagram

2.2 Memory map

Memory map for CUB-43USB-NFS is listed in Table.2-4.
Address value in memory map is relative with a starting address of CUB-43USB-NFS, and actual address has the value that added a starting address of the board.

Table.2-4 Memory Map

Address	Function
000H 〜 1FFH	GM:Global Memory
200H 〜 2FFH	MSB:Mail Send Buffer
300H 〜 3FFH	Register and Reserved(390H 〜 3FFH)
400H 〜 4FFH	MRB0:Mail Receive Buffer0
500H 〜 5FFH	MRB1:Mail Receive Buffer1
600H 〜 7FFH	Manufacture's Reserved

To access to the MKY43 mounted on CUB-43USB-NFS, write access or read access to addresses "000H − 5FFH" listed in Table.2-4 Memory map. For details of each area, refer to "MKY43 User's Manual."

Caution

Do not access to addresses 600H to 7FFH (Manufacturer's reserved area.)
To do so, it may cause unstable system operation.

Chapter 3 Software

This chapter describes API from StepTechnica for using CUB-43USB-NFS.

3.1 Outline

Offers a DLL in order to simplify the access to CUB-43USB-NFS from the user application on Windows.
DLL can be downloaded from the download page of StepTechnica.
URL：http://www.steptechnica.com/jp/download/index.html
Supported OS:

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

DLL can be called from Microsoft Visual Studio and VB6.

Reference

For the latest information including the addition of compliant OS or software update,
visit our website (http://www.steptechnica.com)

3.2 Copyright and disclaimer

The copyright of all documents / program / program sources in the attached driver disc are belong to StepTechnica Co., Ltd.
The individuals, companies or other parties only who accept the cautions written below and use our CUB-43USB-NFS is licensed to copy or use these works of StepTechnica Co., Ltd.
StepTechnica Co., Ltd. does not accept not only to revise or distribute any part of these copyrighted works but also to copy and use for purposes not mentioned above without prior permission.

Caution

① StepTechnica Co., Ltd. assume no responsibility for any results caused by using the attached driver disc or all softwares downloaded from our website.

② Use library in proper ways with its instructions.

③ All specifications and contents in the disc is subject to change without prior notice. StepTechnica Co., Ltd. does not guarantees for any future replacements.

④ StepTechnica Co., Ltd. does not support for the questions regarding OS or development environment not related to StepTechnica products.

⑤ If any bugs or errors are found, contact StepTechnica's engineer team.

3.3 Type of files

The files in “DLL” folder in the disc are given below.

【 DLL】

|

+ ----	[cub43usbnfs.dll]	： DLL body.Copy it to the system folder of Windows or the directory in which there is the user program using this DLL before use.
+ ----	[cub43usbnfs.lib]	： Import libraries.
+ ----	[cub43usbnfs.h]	： DLL header file. Get this included after Windows.h at use.

3.4 API specification

Describes CUBAPI specification.

Table.3-1 API function

Function	Description
CubGetVersion	Obtain the version information of library
CubGetLastError	Obtain the termination status of CUB API function
CubOpenHandle	Open handles to CUB-43USB-NFS board
CubCloseHandle	Closed handle generated by CubOpenHandle
CubCountDevice	Obtain the number of CUB-43USB-NFS devices
CubStartAutoTrans	Start of periodic transmission
CubStopAutoTrans	Stop of periodic transmission
CubReadWord	2-byte data read from CUB-43USB-NFS
CubWriteWord	2-byte data write to CUB-43USB-NFS
CubReadProtect	Read using the hazard prevention function from the global memory
CubWriteProtect	Write using the hazard prevention function to the global memory
CubReadGM	Get all global memory sent by the periodic transmission function
CubReadMFR	Get MFR sent by the periodic transmission function
CubReadData	Data read of the specified word length from the specified address of the global memory
CubWriteData	Data write of the specified word length to the specified address of the global memory

3.4.1 CubGetVersion

Format

UINT CubGetVersion(void);

Description

Get the API version number.

Parameter

None

Return parameter

Number representing the API version number (BCD code hexadecimal)

Error code

CUB_SUCCESS	Terminated normally

Caution

It is not required, but can increase the safety by making the compatibility check for DLL in the user application that uses the cub43usbnfs.dll.
In the safety to say here, it means avoiding such forced termination of the program by avoiding incompatible function call in advance.
CubGetVersion API obtains the version of DLL. The version number of the numbers this API returns, configured as shown in Table.3-2. Cause that each number is up is as follows.

Major number : The revision with no compatibility such as API specification change.
Minor number : The revision with backword compatibility such as API addition.
Update number : This revision with no specification change such as bug fixes.

Number of minor number and update number may be ignored on compatibility. If the value of the major number has changed, it is recommended that do not call the API. Compatibility needs to be checked prior to the initialization.

Table.3-2 configuration of the version number

Return parameter(case)	Major number (bit15～8)	Minor number (bit7～4)	Update number (bit3～0)
0x0102	1	0	2
0x1398	13	9	8

3.4.2 CubGetLastError

Format

UINT CubGetLastError(void);

Description

Get the exit status of CUBAPI function called last time.

Parameter

None

Return parameter

Return an error code that is defined in the cub43usbnfs.h.

Caution

Table.3-3 describing the error code that is defined in the cub43usbnfs.h.

Table.3-3 Error code list

Character constant	Value	Detail
CUB_SUCCESS	0	Terminated normally.
CUB_ERR_DEVICENOTEXIST	1	Device does not exist.
CUB_ERR_ALREADYOPENED	2	Already opened.
CUB_ERR_CLOSED	3	CubOpenHandle has never been called.
CUB_ERR_INVALIDPARAM	4	Called with invalid parameter.
CUB_ERR_NORESOUCE	5	No resource to execute the process.
CUB_ERR_FAILED	6	The process failed due to unknown reason.
CUB_ERR_AUTO_TRANS_ALREADY_START	7	Periodic transmission is already started.
CUB_ERR_AUTO_TRANS_STOP	8	Periodic transmission is not started.
CUB_NOTCALLYET	99	CUBAPI has never been called yet.

3.4.3 CubCountDevice

Format

INT CubCountDevice(void);

Description

Obtain the number of CUB-43USB-NFS devices.

Parameter

None

Return parameter

Return the CUB-43USB-NFS of the number.

-1	：Over 3 boards exist
0	：No boards exist
1 ～ 2	：1 or 2 boards exist

Error code

CUB_SUCCESS	Terminated normally

3.4.4 CubStartAutoTrans

Format

BOOL CubStartAutoTrans(HANDLE CUBHandle, WORD MfCnt);

Description

Start of periodic transmission from CUB-43USB-NFS. The periodic transmission is a function to transmit data to the PC in the transmission period specified all global memory and MFR from CUB-43USB-NFS.
The transmitted data will be retained by the internal API. Held data can be obtained by CubReadGM or CubReadMFR.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
WORD MfCnt	Set the transmission period of the periodic transmission. Transmission cycle can be specified with 125us interval from 1ms to 100ms. Periodic transmission determine the transmission period with reference to Table.3-4 When not perform periodic transmission,set to 0.

Table.3-4 Set list of transmission cycle

Setting	Transmission cycle(usec)	Setting	Transmission cycle(usec)
0	not perform periodic transmission	:	:
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(100ms)

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_AUTO_TRANS_ALREADY_START	Periodic transmission is already started.
CUB_ERR_FAILED	The process failed due to unknown reason.

Caution

There is a possibility that can not receive the data that is sent periodically from CUB-43USB-NFS by spec of PC or other running application. API other than CubReadGM and CubReadMFR even when not performing periodic communication is available.

3.4.5 CubStopAutoTrans

Format

BOOL CubStopAutoTrans(HANDLE CUBHandle);

Description

Stop of periodic transmission.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_AUTO_TRANS_STOP	Periodic transmission is not started.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.6 CubOpenHandle

Format

HANDLE CubOpenHandle(int Instance);

Description

Generate handle that identifies a connection to the CUB-43USB-NFS.

Parameter

int Instance	When a plurality of CUB-43USB-NFS exist , sets the number for obtaining the handles separately. If CUB-43USB-NFS is a single, it is set to zero

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_ALREADYOPENED	Already opened.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.7 CubCloseHandle

Format

BOOL CubCloseHandle(HANDLE CUBHandle);

Description

Closed handle generated by CubOpenHandle.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.8 CubReadWord

Format

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

Description

2-byte data read from CUB-43USB-NFS.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
const ULONG Adr	Access destination address value. Specify a multiple of 2. Input range is 0x000 ～ 0x7FE.
WORD *Dat	Read data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.9 CubWriteWord

Format

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

Description

2-byte data write to CUB-43USB-NFS.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
const ULONG Adr	Access destination address value. Specify a multiple of 2. Input range is 0x000 ～ 0x7FE.
constWORD Dat	Write data.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.10 CubReadProtect

Format

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

Description

Read using the hazard prevention function from the global memory.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
WORD BlockNo	Memory Block number. Input range is 0 ～ 63.
void *Data	8-byte read data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.11 CubWriteProtect

Format

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

Description

Write using the hazard prevention function to the global memory.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
WORD BlockNo	Memory Block number. Input range is 0 ～ 63.
void *Data	8-byte write data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

3.4.12 CubReadGM

Format

BOOL CubReadGM(HANDLE CUBHandle, void*Data);

Description

Get all global memory sent by the periodic transmission function.
Return an error if call CubReadGM during periodic transmission outage.

Parameter

HANDLE CUBHFandle	Handle value to the CUB-43USB-NFS
void *Data	512-byte write data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_AUTO_TRANS_STOP	Periodic transmission is not started.
CUB_ERR_FAILED	The process failed due to unknown reason.

Caution

CubReadGM is not access to direct MKY43.
If get the global memory directly from the MKY43, use the "CubReadWord" , "CubReadData" or "CubReadProtect".

3.4.13 CubReadMFR

Format

BOOL CubReadMFR(HANDLE CUBHandle, void *Data);

Description

Get MFR sent by the periodic transmission function.
Return an error if call CubReadMFR during periodic transmission outage.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
void *Data	8-byte write data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_AUTO_TRANS_STOP	Periodic transmission is not started.
CUB_ERR_FAILED	The process failed due to unknown reason.

Caution

CubReadMFR is not access to direct MKY43.
If get the global memory directly from the MKY43, use the "CubReadWord" or "CubReadData".

3.4.14 CubReadData

Format

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

Description

Data read of the specified word length from the specified address of the global memory.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
WORD Adr	Access destination address value. Specify a multiple of 2. Input range is 0x000 ～ 0x7FE.
WORD WordLen	Specify the word length. Input range is 0x001 ～ 0x0400.
void *Data	Read data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

Caution

If the specified read range exceeds the range of MKY43 (0x800), return an error.

3.4.15 CubWriteData

Format

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

Description

Data write of the specified word length to the specified address of the global memory.

Parameter

HANDLE CUBHandle	Handle value to the CUB-43USB-NFS
WORD Adr	Access destination address value. Specify a multiple of 2. Input range is 0x000 ～ 0x7FE.
WORD WordLen	Specify the word length. Input range is 0x001 ～ 0x0400.
void *Data	Write data storage destination address.

Return parameter

Return processing result. Succeeded: TRUE(1) is returned. Failed: FALSE(0) is returned.

Error code

CUB_SUCCESS	Terminated normally.
CUB_ERR_INVALIDPARAM	Called with invalid parameter.
CUB_ERR_FAILED	The process failed due to unknown reason.

Caution

If the specified write range exceeds the range of MKY43 (0x800), return an error.

Chapter 4　Appendix

4.1　Outline

Describes the sample program of the initialization processing and end processing to control the CUB-43USB-NFS.
The structure and function of MKY43 registers that are described in the sample program is refer to "MKY43 User's Manual" of "Chapter 5 Register Reference".

```
int main(int argc, char argv[])
{
    unsigned char buf[0x200];

    /* Check the API version number */
    int version=CubGetVersion();
    if (version < 0x100 || version > 0x199) {
            printf("It is incompatible version of cub43usbnfs.dll.\n");
            exit(1);
    }

    /** Obtain the number of CUB-43USB-NFS devices
    * (If CUB-43USB-NFS is one, can be omitted.)
    */
    const int count=CubCountDevice();
    if (count < 1) {
      printf("CUB-43USB-NFS not exist.");
      exit(1);
    }

    /** Generate handle that identifies a connection to the CUB-43USB-NFS.
    * If CUB-43USB-NFS is known to be not connected only one, it open by argument 0.
    */
    HANDLE dev_handle;
    dev_handle=CubOpenHandle(0);
    if (dev_handle == INVALID_HANDLE_VALUE) {
      exit(1);
    }
```

```
        memset(buf, 0, sizeof(buf));
        /** Initialization of CUnet*/

        // Clear of global memory.
        CubWriteData(dev_handle, 0, 0x100, buf);

        // Clear of mail transmission buffer.
        CubWriteData (dev_handle, 0x200, 0x80, buf);

        // Clear of mail receive buffer0.
        CubWriteData (dev_handle, 0x400, 0x80, buf);

        // Clear of mail receive buffer1.
        CubWriteData (dev_handle, 0x500, 0x80, buf);

        // Transition to GMM mode by setting 0x8000 to SCR.
        CubWriteWord (dev_handle, 0x366, 0x8000);

        // Configuration change to SA = 1, OWN = 1, BPS = 3Mbps.
        CubWriteWord (dev_handle, 0x356, 0x0141);

        // GMM mode cancellation. F
        CubWriteWord (dev_handle, 0x366, 0);

        /** After the initialization is complete communication start.
        * Start-bit of SCR set to "1" , to start the network.
        */
        CubWriteWord (dev_handle, 0x366, 0x0100);
        /** Start of periodic transmission.
        * 3000us (3msec) cycle in data transmission.
        */
        CubStartAutoTrans(dev_handle, 32 );
        /** -- Programming the user processing -- **/

        /** Stop of periodic transmission. */
        CubStopAutoTrans(dev_handle);
        /** Closed handle generated by CubOpenHandle. */
        CubCloseHandle(dev_handle);
        return 0;
}
```

■ Developed and manufactured by

StepTechnica Co., Ltd

757-3, Shimofujisawa, Iruma, Saitama
TEL: 04-2964-8804
http://www.steptechnica.com/
info@steptechnica.com

CUnet(MKY43)USB Unit
CUB-43USB-NFS
User's Manual

Document No：STD-CUB-43USB-NFS_V1.0E
Issued: August 2015