

UHF 帯 RFID リーダ・ライタ PC との USB 接続ガイド

2019 年 9 月 2 日 第 1.0.0 版

株式会社アートファイネックス

変更履歴

日付	版数	変更内容
2019/09/02	1.0.0	初版発行

はじめに

本書は、アートファイネックス製 UHF 帯 RFID リーダ・ライタ(以下、リーダ・ライタとします)を PC と USB で接続する方法、注意点を説明したものです。

「UHF 帯 RFID リーダ・ライタ共通セットアップガイド」の内容をより詳細に記述しております。

無断転載を禁じます。

本書の内容は、断りなく変更することがあります。

- ※ Microsoft Windows は、米国 Microsoft Corp.の登録商標です。
 - ※ FTDI は、Future Technology Devices International Limited の商標または登録商標です。
 - ※ その他、商品名及び製品名などは一般に各社の商標または登録商標です。
-

目次

1. USB ドライバのインストール方法	1
1.1. 自動的にインストールするように事前準備しておく方法	2
1.2. インターネット上にあるドライバを探してインストールする方法	4
1.3. PC 内にあるドライバを指定してインストールする方法	5
2. USB ドライバのインストール確認方法	7
3. USB ドライバのアンインストール方法	8
3.1. アンインストーラを使用する方法	8
3.2. 手動でアンインストールする方法	12
4. 複数の USB 機器を接続するときの注意	13
5. Tips	14
5.1. BM 値	14
5.2. USB セレクティブサスペンド	16

1. USB ドライバのインストール方法

弊社のリーダー・ライターの USB インターフェースは FTDI 社製の USB シリアル変換 IC を使用しています。

そのため、PC に FTDI 社が提供するデバイスドライバをインストールしておく必要があります。

インストール方法は下の3種類ありますので、いずれかの方法でインストールをしてください。

1. 自動的にインストールするように事前準備しておく方法

事前にセットアップファイルを実行しておくことで、PC にリーダー・ライターを USB ケーブルで接続すると自動的にインストールされます。

2. インターネット上にあるドライバを探してインストールする方法

PC にリーダー・ライターを USB ケーブルで接続した後、インターネット上にあるドライバをインストールします。

3. PC 内にあるドライバを指定してインストールする方法

PC にリーダー・ライターを USB ケーブルで接続した後、PC 内にあるドライバを指定してインストールします。

1.1. 自動的にインストールするように事前準備しておく方法

弊社から提供している DK の [USBDriver] フォルダにある [CDMxxxx_Setup.exe] を実行しておく、PC とリーダ・ライタを USB ケーブルで接続すると自動的にインストールが行われるようになります。

この実行ファイルの最新版は、以下の手順で FTDI 社のホームページから入手することができます。

FTDI 社のホームページを開きます。<https://www.ftdichip.com/Drivers/VCP.htm>

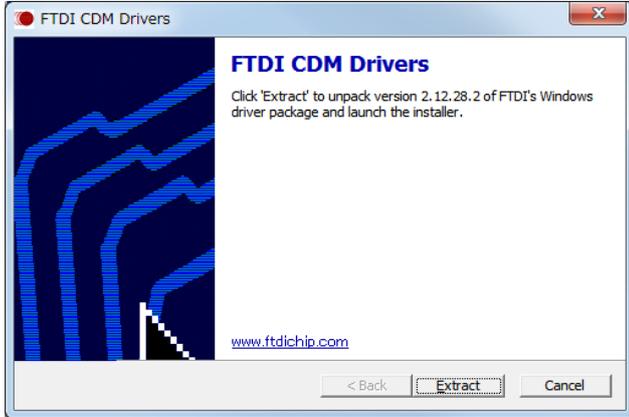
Currently Supported VCP Drivers:

Operating System	Release Date	Processor Architecture							Comments
		x86 (32-bit)	x64 (64-bit)	PPC	ARM	MIPSII	MIPSIV	SH4	
Windows*	2017-08-30	2.12.28	2.12.28	-	-	-	-	-	WHQL Certified. Includes VCP and D2XX. Available as a setup executable . Please read the Release Notes and Installation Guides .

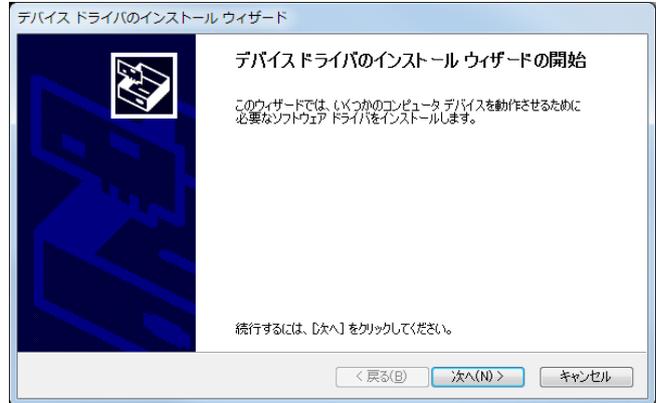
Windows の Comments 欄にある「[setup executable](#)」をクリックしてインストール実行ファイル入手します。(ZIP ファイルを解凍すると実行ファイル(exe ファイル)を入手できます。)

1. 実行ファイル [CDMxxxx_Setup.exe] を実行します。

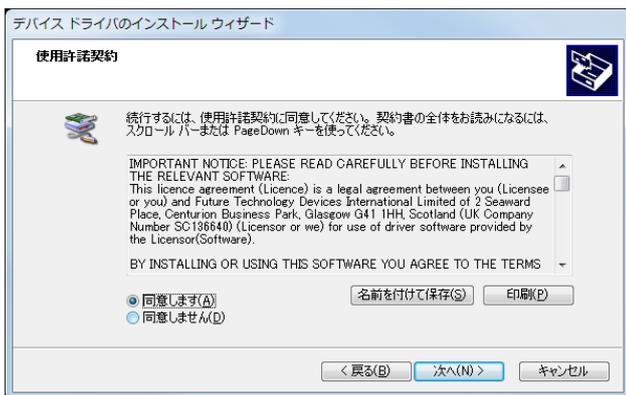
① [Extract] を押下します。



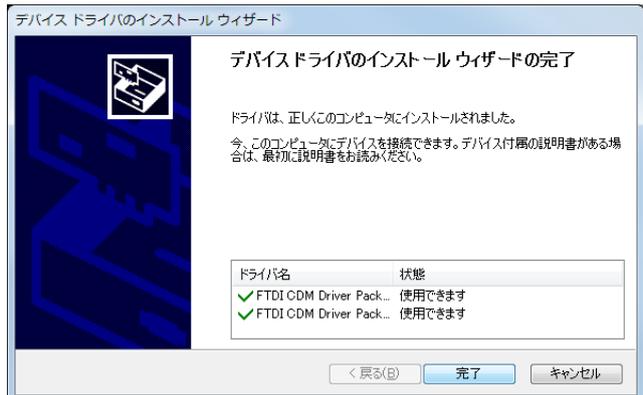
② [次へ] を押下します。



③ 「使用許諾契約内容」に問題がなければ [同意します] を選択し [次へ] を押下します。



④ ドライバのインストールが完了すると、以下の画面が表示されます。 [完了] ボタンを押下し、インストーラを終了します。



2. これ以降は、PC とリーダ・ライタを USB ケーブルで接続すると自動的にインストールが行われます。

1.2. インターネット上にあるドライバを探してインストールする方法

インターネット上にあるドライバをインストールする方法です。

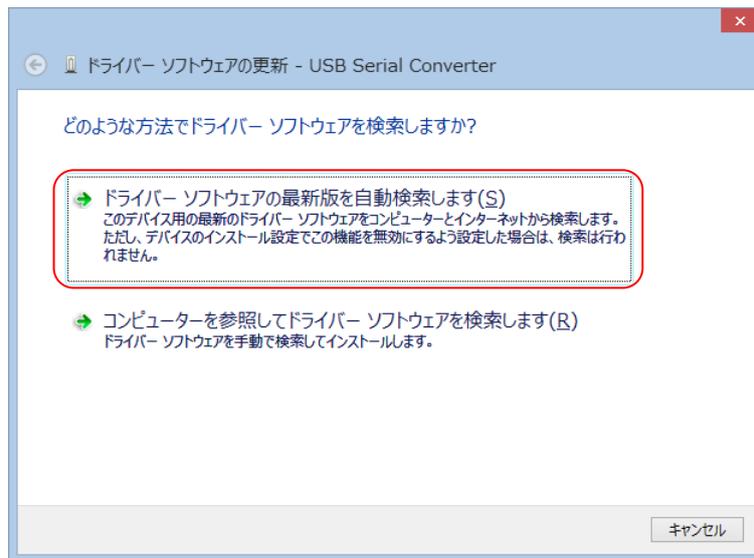
0. PC がインターネットに接続されていること。

1. PC とリーダ・ライタを USB ケーブルで接続します。

2. 「コントロールパネル」 - 「デバイスとプリンター」 - 「デバイスマネージャー」を開きます。

「ほかのデバイス」に「！」で表示されているものがあるはずです。

それを右クリックし「ドライバーソフトウェアの更新...」をクリックすると下図のダイアログが表示されます。



「ドライバーソフトウェアの最新版を自動検索します」をクリックします。

するとインターネット上からドライバを探してインストールを行います。

これで1つ目のインストールは終了です。デバイスマネージャーの「ほかのデバイス」を見ると「！」で表示されているものがまだあるはずです。先ほどと同様に「ドライバーソフトウェアの更新...」をクリックしてインストールします。

これ以降、PC とリーダ・ライタを USB ケーブルで接続すると自動的にインストールを行うようになります。

1.3. PC 内にあるドライバを指定してインストールする方法

PC 内にあるドライバを指定してインストールする方法です。

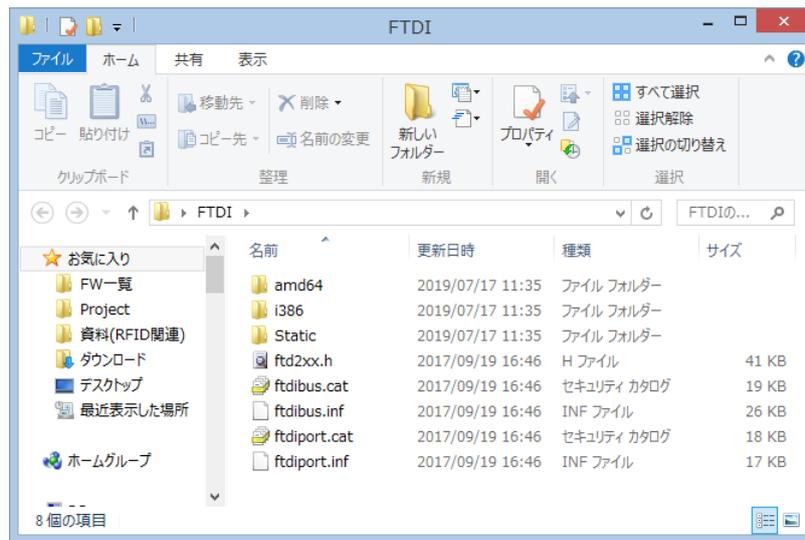
FTDI 社のホームページからデバイスドライバを入手しておき、PC とリーダー・ライタを USB ケーブルに接続した後、それを指定してインストールします。

FTDI 社のホームページを開きます。<https://www.ftdichip.com/Drivers/VCP.htm>

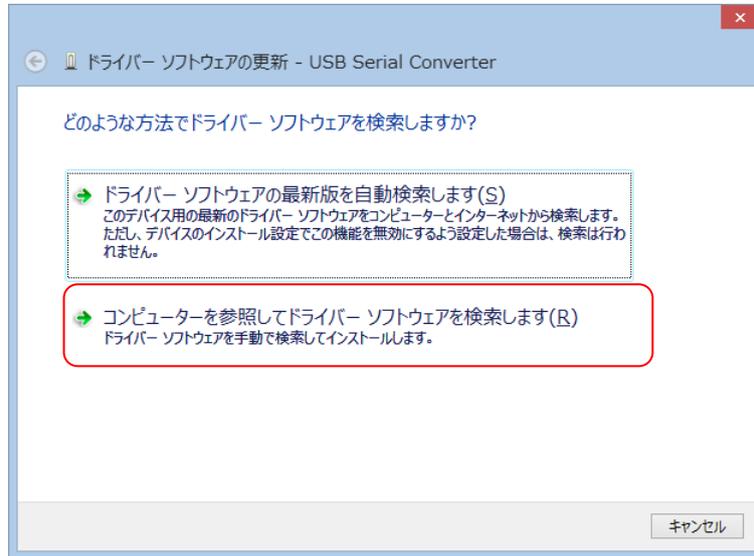
Currently Supported VCP Drivers:

Operating System	Release Date	Processor Architecture							Comments
		x86 (32-bit)	x64 (64-bit)	PPC	ARM	MIPSII	MIPSIV	SH4	
Windows*	2017-08-30	2.12.28	2.12.28	-	-	-	-	-	WHQL Certified. Includes VCP and D2XX. Available as a setup executable. Please read the Release Notes and Installation Guides.

リーダー・ライタと USB 接続する PC に合わせて Windows の x86(32bit) または x64(64bit) 欄にあるバージョン番号をクリックしてドライバファイル一式を入手します。(ZIP ファイルを解凍すると下図のファイルを入手できます。)



1. PC とリーダ・ライタを USB ケーブルで接続します。
2. 「コントロールパネル」 - 「デバイスとプリンター」 - 「デバイスマネージャー」を開きます。
「ほかのデバイス」に「！」で表示されているものがあるはずです。
それを右クリックし「ドライバーソフトウェアの更新...」をクリックすると下図のダイアログが表示されます。



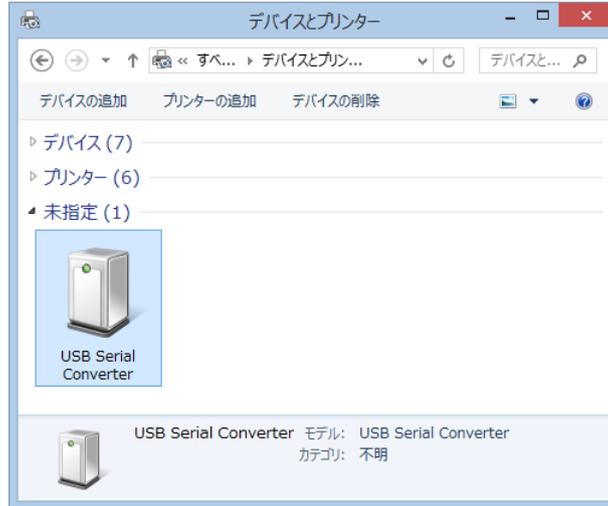
- 「コンピューターを参照してドライバーソフトウェアを検索します」をクリックします。
入手済みのドライバがあるフォルダを指定してインストールを行います。
これで1つ目のインストールは終了です。デバイスマネージャーの「ほかのデバイス」を見ると「！」で表示されているものがまだあるはずです。先ほどと同様に「ドライバーソフトウェアの更新...」をクリックしてインストールします。
これ以降、PC とリーダ・ライタを USB ケーブルで接続すると自動的にインストールを行うようになります。

2. USB ドライバのインストール確認方法

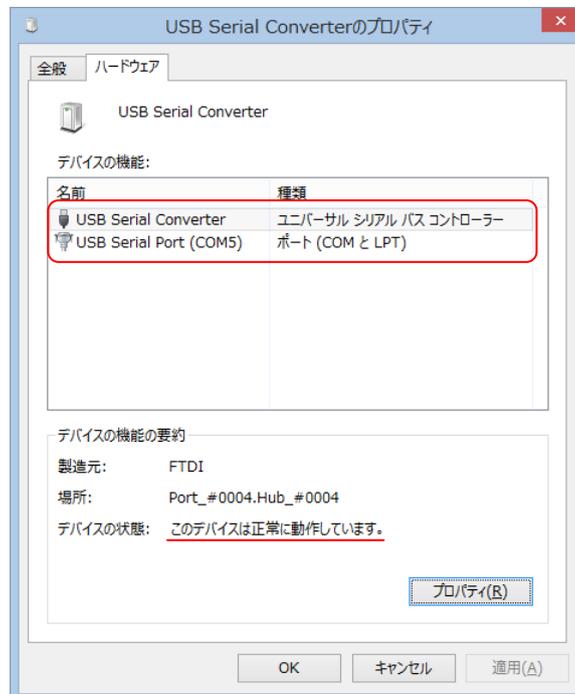
ドライバのインストールが正常に完了したかどうかを下の手順で確認できます。

「コントロールパネル」 - 「デバイスとプリンター」を開きます。

「USB Serial Converter」を右クリック - 「プロパティ」 - 「ハードウェア」を開きます。



下図のように「USB Serial Converter」と「USB Serial Port (COM**)」が表示されていたら成功です。

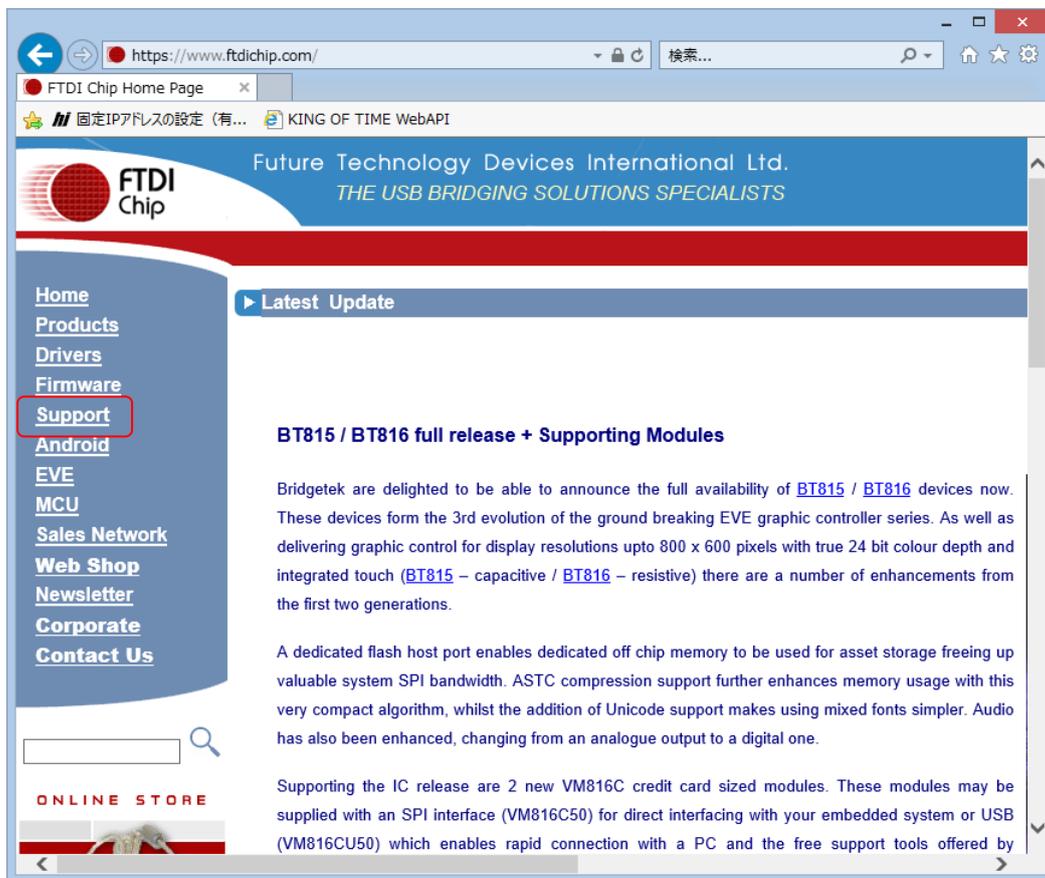


3. USB ドライバのアンインストール方法

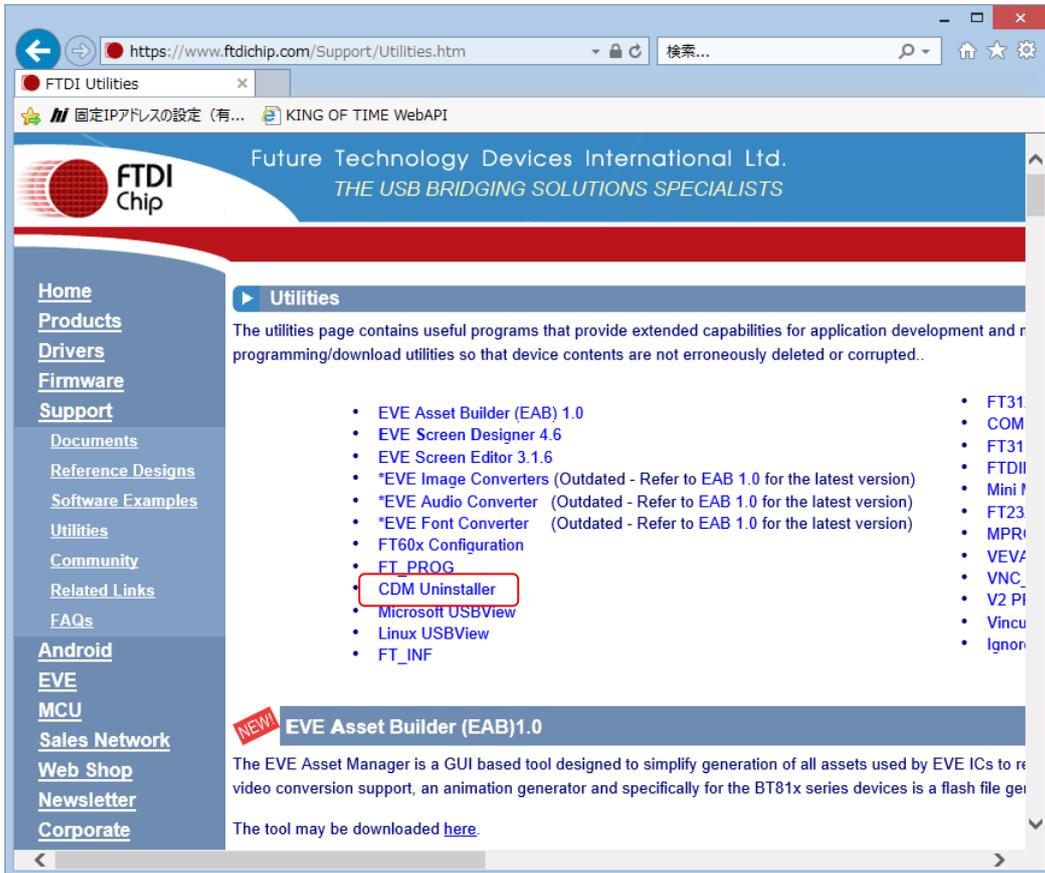
3.1. アンインストーラを使用する方法

FTDI 社が提供しているアンインストーラを使用します。

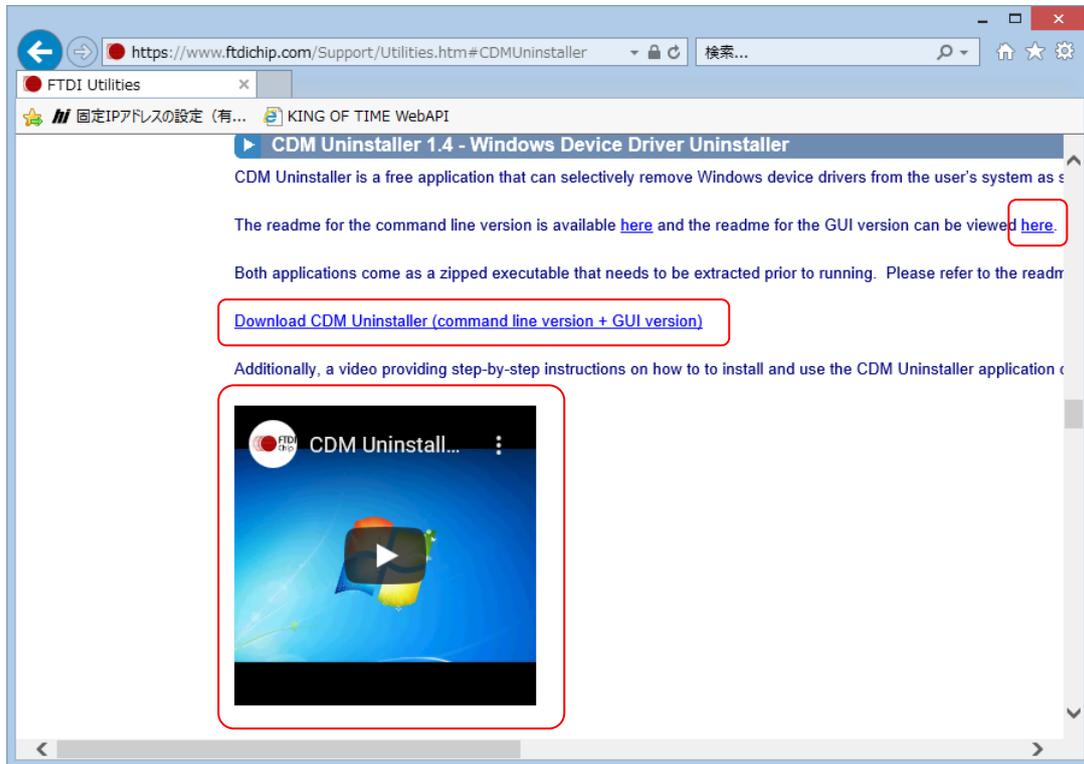
- (1) FTDI 社のホームページをひらきます。<https://www.ftdichip.com/>
- (2) 左側にあるメニューから [Support] – [Utilities] を開きます。



(3) Utilities 一覧にある [CDM Uninstaller] をクリックします。

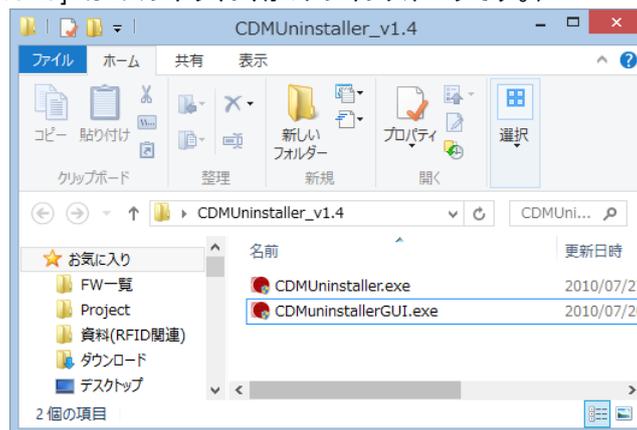


- (4) [Download CDM Uninstaller (command line version + GUI version)] をクリックして、ZIP ファイルをダウンロードします。

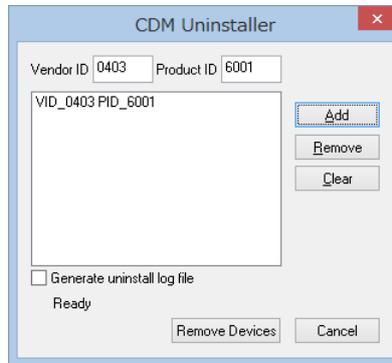


上画面の [here] で詳しい使い方を見ることができます。また、動画でも見るすることができます。

- (5) ダウンロードしたファイルを解凍します。
(6) [CDMuninstallerGUI.exe] を実行します。
([CDMUninstaller.exe] はコマンドライン用のアンインストーラです。)



- (7) [Add] をクリックして画面中央にアンインストールするデバイスを表示した後、[Remove Devices] をクリックしてアンインストールします。この時、PC に接続されているデバイスは抜いておきます。



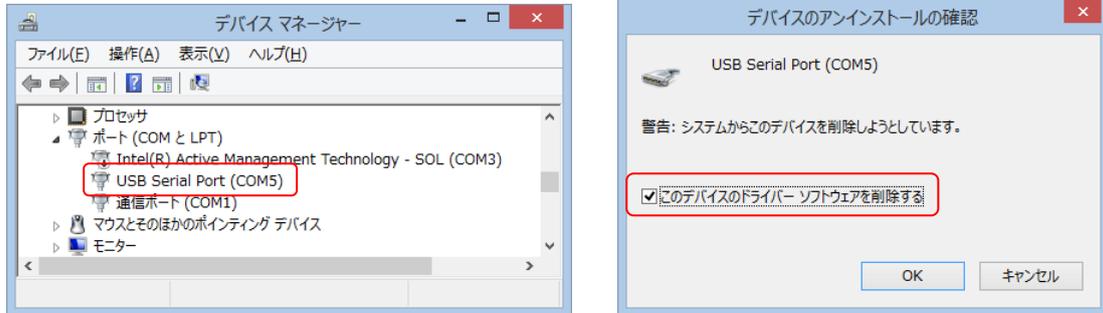
- (8) アンインストールが完了したというダイアログが表示されますので [OK] をクリックして閉じます。

3.2. 手動でアンインストールする方法

(1) デバイスを PC に接続した状態で、コントロールパネルからデバイスマネージャーを開きます。

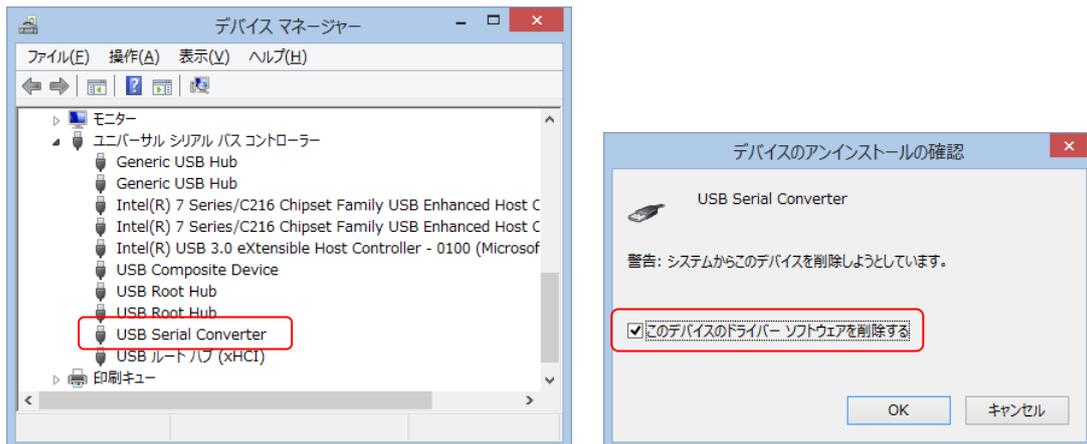
(2) デバイスが接続されているポートを確認し、右クリックして [削除] をクリックします。

確認ダイアログが表示されますので、チェックをして [OK] をクリックします。



(3) 引き続き、デバイスが接続されているユニバーサルシリアルバスコントローラを確認し、右クリックして [削除] をクリックします。

確認ダイアログが表示されますので、チェックをして [OK] をクリックします。



4. 複数の USB 機器を接続するときの注意

PC に複数のリーダー・ライターが USB ケーブルで接続されている場合、もしくはリーダー・ライターは1台だが他の FTDI 社製の USB シリアル変換 IC を搭載した機器が接続されている場合に、気を付けなければならない API 使用時の注意事項を記します。

始めにリーダー・ライターとコネクトするために `so_CommSetup()` を発行します。

この関数では USB 接続のリーダー・ライターとコネクトするときには下の2通りの方法があります。

1. 仮想 COM 番号と回線速度を指定して接続する。(RS-232C インターフェース)
2. 任意の USB シリアル変換機器と接続する。(USB インターフェース)

もし、上のように PC に複数の USB シリアル機器が接続されている状態で2. の方法で接続すると、どの USB シリアル機器とコネクトするかは不定です。そのため、このような場合は1. の方法で接続します。

リーダー・ライタの仮想 COM ポート番号は、デバイスマネージャーでご確認ください。リーダー・ライターを抜き差しすると該当の「USB Serial Port(COMxx)」が消えるまたは表示されます。



例えば、上図のように2つのリーダー・ライターが COM5 と COM6 につながっていたら、それぞれ

```
so_CommSetup(255, 5, 19200);
```

```
so_CommSetup(255, 6, 115200);
```

のようにします。

5. Tips

5.1. BM 値

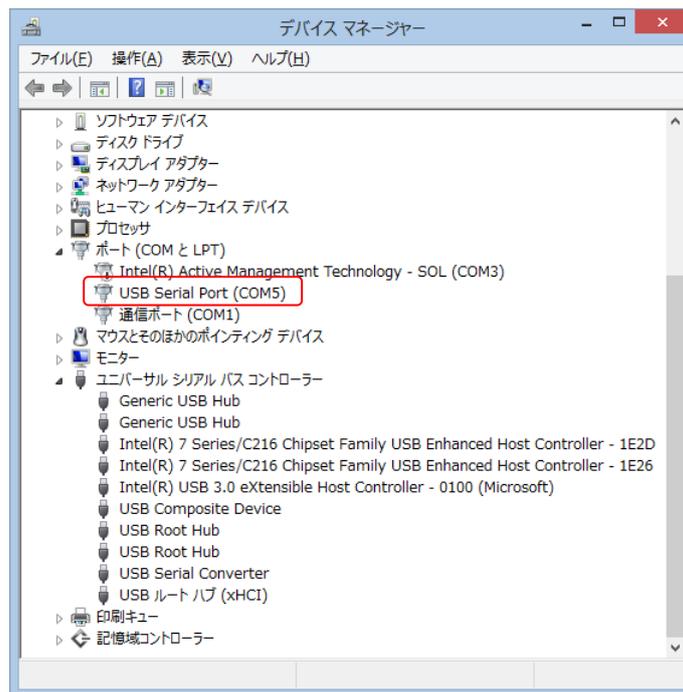
「PC とリーダ・ライタとの通信が遅い」、「通信タイムアウトが発生する」や「通信が途絶える」などの症状が稀に発生する場合、BM 値を変更することで解消することがあります。

《BM 値とは》

USB シリアル変換 IC は、受信したシリアルデータをバッファに溜めて、バッファがいっぱいになったら USB ホスト側へ転送します。また、シリアルデータが「指定された時間」来なくなったら、バッファがいっぱいでなくても USB ホスト側へ転送します。このときの「指定された時間」が BM 値です。

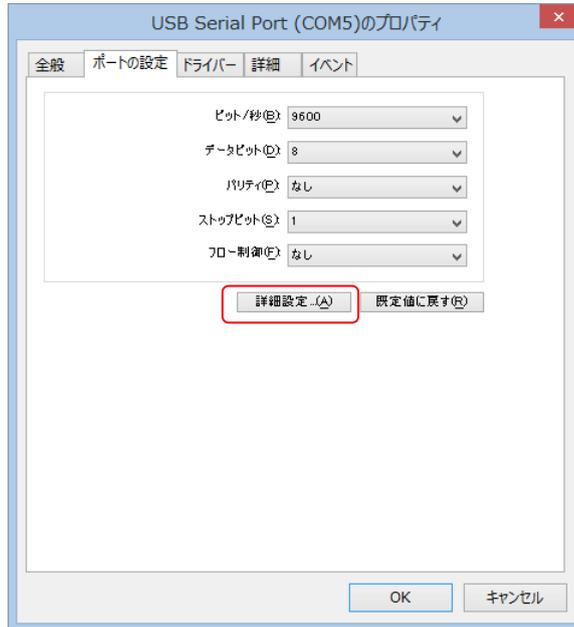
BM 値は以下の手順で変更できます。

1. 「コントロールパネル」 - 「デバイスとプリンター」 - 「デバイスマネージャー」を開きます。

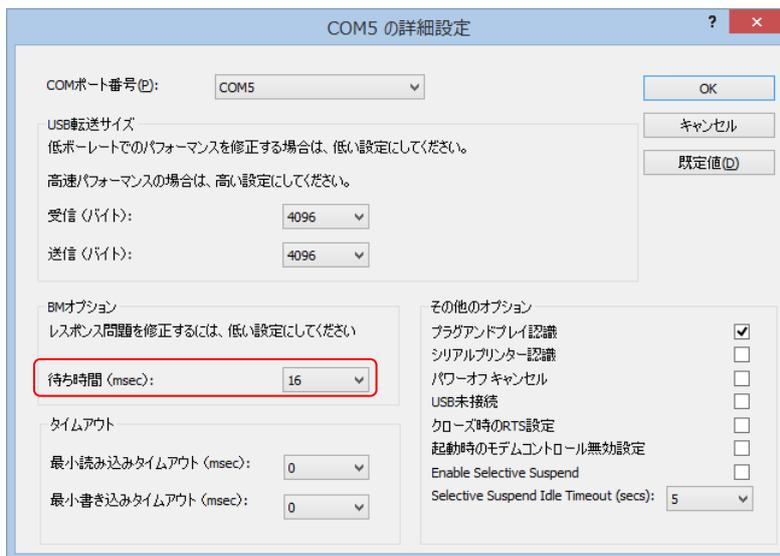


PC に USB ケーブルで接続されているリーダ・ライタを右クリックし、「プロパティ」 - 「ポートの設定」をクリックします。

下図のダイアログの「詳細設定」をクリックします。



下図のダイアログが表示されます。



上図の赤色枠が BM 値です。この時間を小さくします。(ex.16msec⇒1msec)

5.2. USB セレクティブサスペンド

外部電源が不要なリーダ・ライタの場合、電源は USB ケーブルから供給されます。

PC から電源が供給されなくなるとリーダ・ライタは停止し、PC とのコネクトが切断されます。

「一定の時間が経過するとリーダ・ライタとのコネクトが切断される」などの症状が発生する場合は、この設定を変更することで解消することがあります。

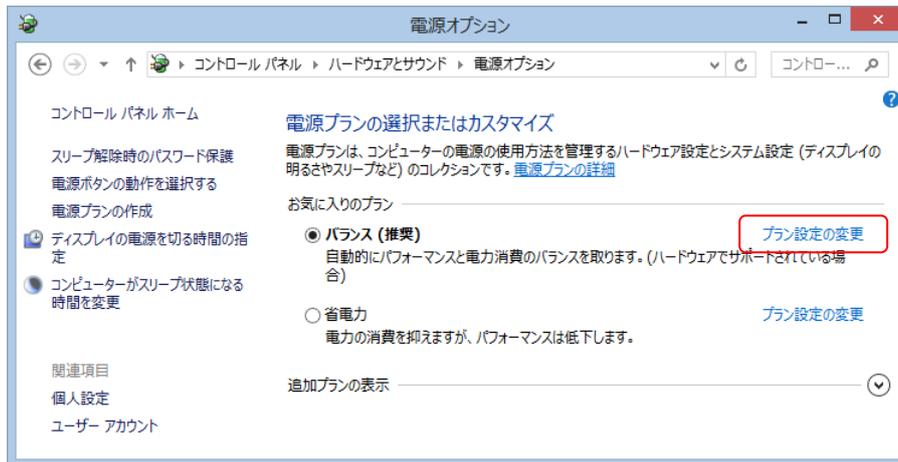
《USB セレクティブサスペンドとは》

パソコンの消費電力を抑えるため、一定の時間が経過すると USB 機器の電源が切れるようになっており、初期設定では USB セレクティブサスペンドは有効になっております。

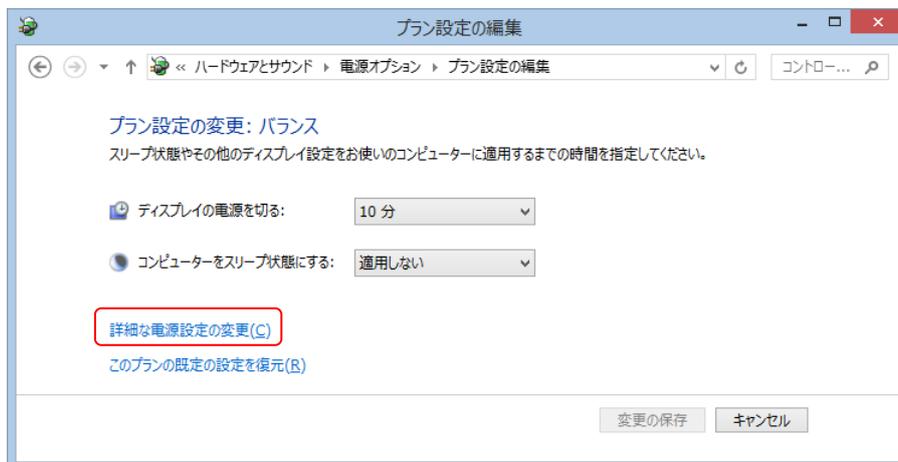
以下の手順で USB セレクティブサスペンドを無効にできます。

「コントロールパネル」 - 「電源オプション」を開きます。

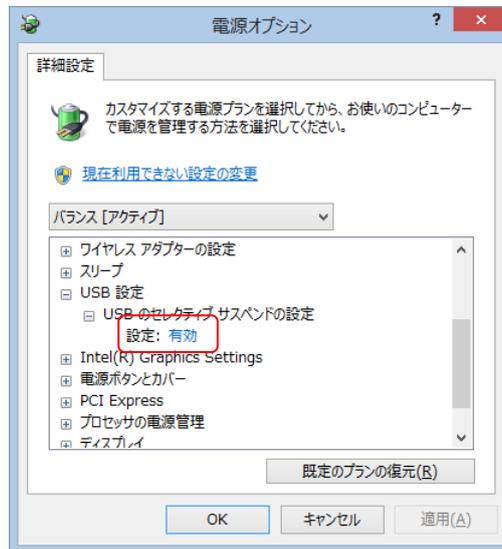
「お気に入りのプラン」で選択している方の「プラン設定の変更」をクリックします。



「詳細な電源設定の変更」をクリックします。



下図のダイアログが表示されますので、「USB 設定」 - 「USB のセレクトティブサスペンドの設定」の設定を「無効」にします。



※ノート PC の場合は、上図の「設定」が「バッテリー駆動」と「電源に接続」の2項目あります。

※PC によっては「USB 設定」の項目が無い場合があります。

以上

UHF 帯 RFID リーダ・ライタを FTDI でリセットする方法

2021 年 2 月 22 日 第 2.0.0 版

株式会社アートファイネックス

変更履歴

日付	版数	変更内容
2020/10/02	1.0.0	初版発行
2021/02/22	2.0.0	FTDI デバイスの GPIO を利用してリセットすることを明記

はじめに

本書は、アートファイネックス製 UHF 帯 RFID リーダ・ライタ(以下、リーダー・ライターとします)を FTDI デバイス経由でリセットする方法を説明したものです。

無断転載を禁じます。

本書の内容は、断りなく変更することがあります。

- ※ Microsoft Windows は、米国 Microsoft Corp.の登録商標です。
 - ※ FTDI は、Future Technology Devices International Limited の商標または登録商標です。
 - ※ その他、商品名及び製品名などは一般に各社の商標または登録商標です。
-

目次

1. はじめに.....	1
2. リセットの原理.....	1
3. Windows の場合.....	1
4. Windows 以外の場合.....	2
4.1. 事前に用意しておくもの.....	3
4.2. FTDI 社提供の関数.....	4
4.3. リセットサンプルアプリ.....	4
4.3.1. FTDI デバイスが1つだけ接続されている場合.....	4
4.3.2. FTDI デバイスが複数接続されている場合.....	7
4.4. FTDI デバイスのシリアル番号取得アプリ.....	8

1. はじめに

リーダー・ライターを制御しているときに、何らかのミスやトラブルにより、コマンドを送信しても返答がこなくなる場合があります。そのときはリーダー・ライターをリセットする必要があります。その方法は主に下の 3 つあります。

- (1) USB ケーブルを抜き差しします。
- (2) USB バスパワーを OFF/ON 制御します。
- (3) FTDI デバイスの GPIO を利用してリーダー・ライターにリセット信号を入れます。

このドキュメントでは上の(3)の方法について記します。

2. リセットの原理

リーダー・ライタの UART⇒USB 変換に FTDI 製 USB シリアル変換デバイス(以下 FTDI デバイス)を使用しています。リーダー・ライターが応答しない場合はホストから FTDI デバイスの GPIO を利用しリーダー・ライタのリセットピンを制御することでリセットします。

※USB 接続されていないリーダー・ライターに対してはリセットできません。

※FTDI デバイスとリーダー・ライタの間にリセット信号線が配線されていないリーダー・ライターはリセットできません。

お使いのリーダー・ライターがリセットに対応しているかどうかは、機種名とシリアル番号でお問い合わせください。

3. Windows の場合

弊社提供の Windows アプリ用 API 関数を利用してリーダー・ライターを制御しているときは、次のように対処します。

- ・API 関数 `so_CommSetup` でホスト PC と RFID リーダ・ライターを接続しているときは
API 関数 `so_ResetReader` や `so_ResetReaderAnyTime` をお使いください。
- ・まだ API 関数 `so_CommSetup` でホスト PC と RFID リーダ・ライターを接続していないときは
API 関数 `so_ResetReaderBeforeConnect` をお使いください。

4. Windows 以外の場合

FTDI 社から下の OS に対応した関数が提供されています。

Linux

Mac OS X(10.4 以降)

Windows(2000 以降)

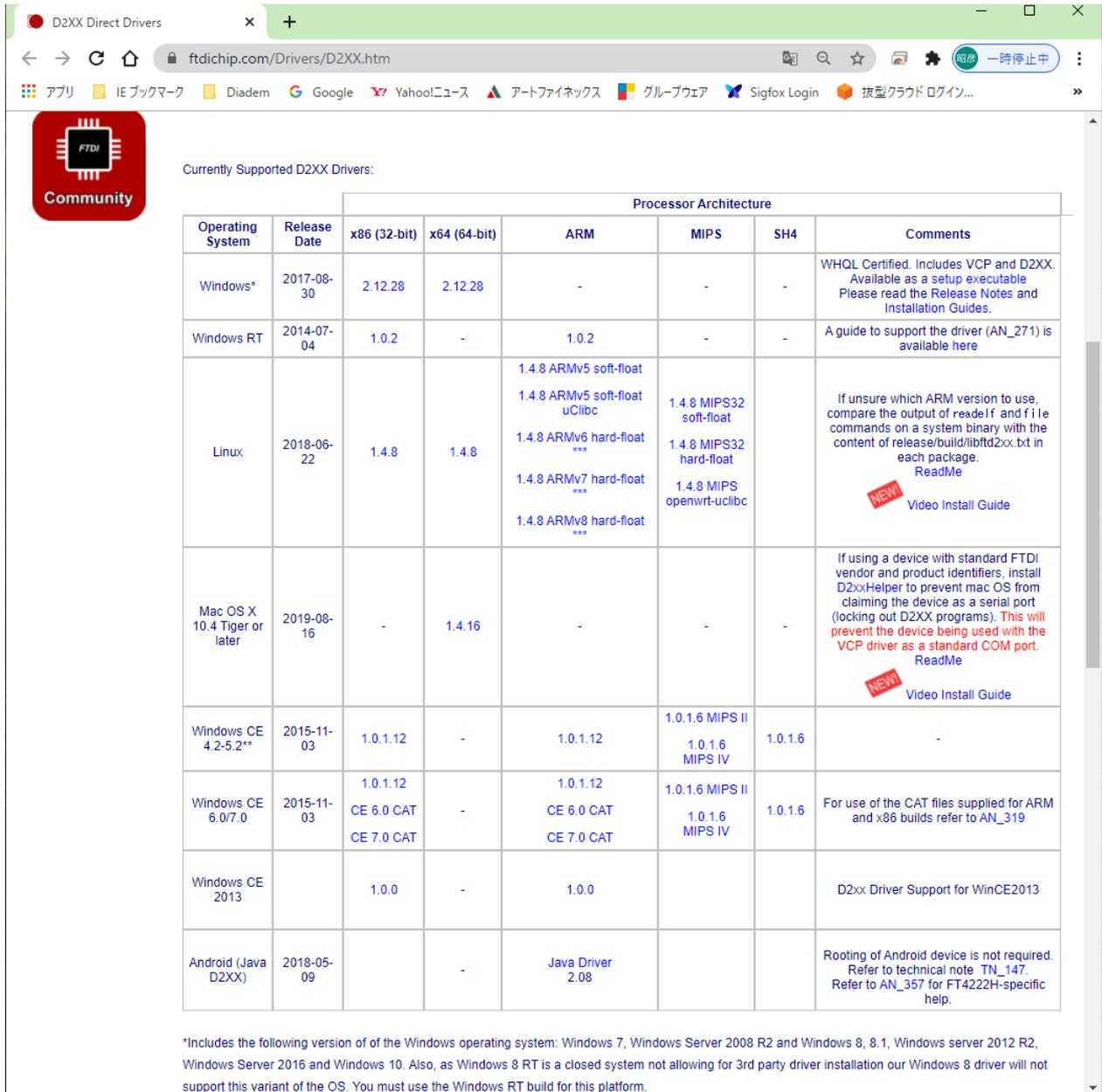
Windows CE(4.2 以降)

この章では、FTDI 社提供の関数を使用して RFID リーダ・ライタをリセットする方法を記します。

4.1. 事前に用意しておくもの

FTDI 社の下 URL からドライバを入手します。その中にある ftd2xx.h と ftd2xx.lib を使用します。

<https://www.ftdichip.com/Drivers/D2XX.htm>



Currently Supported D2XX Drivers:

Operating System	Release Date	Processor Architecture					Comments
		x86 (32-bit)	x64 (64-bit)	ARM	MIPS	SH4	
Windows*	2017-08-30	2.12.28	2.12.28	-	-	-	WHQL Certified. Includes VCP and D2XX. Available as a setup executable. Please read the Release Notes and Installation Guides.
Windows RT	2014-07-04	1.0.2	-	1.0.2	-	-	A guide to support the driver (AN_271) is available here
Linux	2018-06-22	1.4.8	1.4.8	1.4.8 ARMv5 soft-float 1.4.8 ARMv5 soft-float uClibc 1.4.8 ARMv6 hard-float *** 1.4.8 ARMv7 hard-float *** 1.4.8 ARMv8 hard-float ***	1.4.8 MIPS32 soft-float 1.4.8 MIPS32 hard-float 1.4.8 MIPS openwrt-uclibc	-	If unsure which ARM version to use, compare the output of readelf and file commands on a system binary with the content of release/build/libftd2xx.txt in each package. ReadMe NEW! Video Install Guide
Mac OS X 10.4 Tiger or later	2019-08-16	-	1.4.16	-	-	-	If using a device with standard FTDI vendor and product identifiers, install D2xxHelper to prevent mac OS from claiming the device as a serial port (locking out D2XX programs). This will prevent the device being used with the VCP driver as a standard COM port. ReadMe NEW! Video Install Guide
Windows CE 4.2-5.2**	2015-11-03	1.0.1.12	-	1.0.1.12	1.0.1.6 MIPS II 1.0.1.6 MIPS IV	1.0.1.6	-
Windows CE 6.0/7.0	2015-11-03	1.0.1.12 CE 6.0 CAT CE 7.0 CAT	-	1.0.1.12 CE 6.0 CAT CE 7.0 CAT	1.0.1.6 MIPS II 1.0.1.6 MIPS IV	1.0.1.6	For use of the CAT files supplied for ARM and x86 builds refer to AN_319
Windows CE 2013		1.0.0	-	1.0.0			D2xx Driver Support for WinCE2013
Android (Java D2XX)	2018-05-09		-	Java Driver 2.08			Rooting of Android device is not required. Refer to technical note TN_147. Refer to AN_357 for FT4222H-specific help.

*Includes the following version of of the Windows operating system: Windows 7, Windows Server 2008 R2 and Windows 8, 8.1, Windows server 2012 R2, Windows Server 2016 and Windows 10. Also, as Windows 8 RT is a closed system not allowing for 3rd party driver installation our Windows 8 driver will not support this variant of the OS. You must use the Windows RT build for this platform.

4.2. FTDI 社提供の関数

FTDI 社の下 URL に関数のドキュメントがあります。この中の関数を使用します。

[https://www.ftdichip.com/Support/Documents/ProgramGuides/D2XX_Programmer's_Guide_\(FT_000071\).pdf](https://www.ftdichip.com/Support/Documents/ProgramGuides/D2XX_Programmer's_Guide_(FT_000071).pdf)

4.3. リセットサンプルアプリ

ここでは Visual Studio の Win32 コンソールアプリケーションのときのサンプルソースを記します。

4.3.1. FTDI デバイスが1つだけ接続されている場合

接続されている FTDI デバイスを検出し、検出された 1 つ目の FTDI デバイスの GPIO を利用してリーダー・ライターにリセット信号を入れます。

```
#include "stdafx.h"
#include <stdlib.h>
#include "ftd2xx.h"

int main(int argc, char *argv[])
{
    FT_STATUS ftStatus;
    DWORD numDevs;
    char Buffer[64];
    HANDLE hc;
    FT_PROGRAM_DATA ftData;
    char szManufacturer[32], szManufacturerId[16], szDescription[64], szSerialNumber[16];
    unsigned char data;

    // 接続されている台数を入手
    ftStatus = FT_ListDevices(&numDevs, NULL, FT_LIST_NUMBER_ONLY);
    if (ftStatus != FT_OK) {
        // エラー処理
        return 0;
    }
    if (numDevs < 1) { // 1 台も接続されていなかったらエラー
```

```
// エラー処理
return 0;
}

ftStatus = FT_ListDevices((PVOID)0, Buffer, FT_LIST_BY_INDEX | FT_OPEN_BY_DESCRIPTION);
if (ftStatus != FT_OK) {
    // エラー処理
    return 0;
}

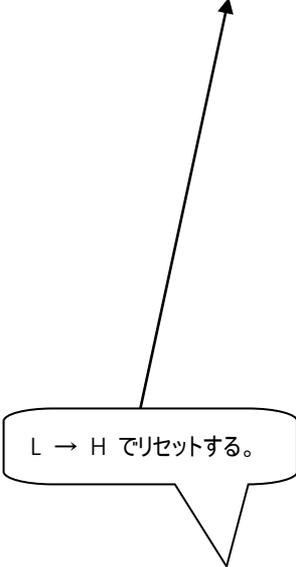
hc = FT_W32_CreateFile((LPCTSTR)Buffer, GENERIC_READ | GENERIC_WRITE, 0,
    NULL, OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED |
    FT_OPEN_BY_DESCRIPTION, NULL);
if (hc == INVALID_HANDLE_VALUE) {
    // エラー処理
    return 0;
}

// 以下 リセット処理
memset(&ftData, 0, sizeof(FT_PROGRAM_DATA));
ftData.Manufacturer = szManufacturer;
ftData.ManufacturerId = szManufacturerId;
ftData.Description = szDescription;
ftData.SerialNumber = szSerialNumber;
ftStatus = FT_EE_Read(hc, &ftData); // 現在の設定値を入手
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
ftData.Signature1 = 0x00000000;
ftData.Signature2 = 0xFFFFFFFF;
ftData.Version = 2;
ftData.Cbus0 = FT_232R_CBUS_IOMODE; // CBUS0 を I/O モードにする
ftStatus = FT_EE_Program(hc, &ftData); // 設定値を書込む
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
```

検出された FTDI デバイスの 1 つ目 (0 番目) を入手

```
}

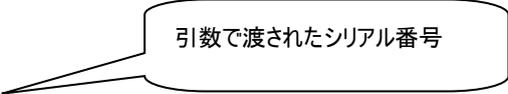
ftStatus = FT_GetBitMode(hc, &data);
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
data |= 0x10;
data &= 0xFE;
ftStatus = FT_SetBitMode(hc, data, FT_BITMODE_CBUS_BITBANG); // CBUS0 を出力&L
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
Sleep(100);
ftStatus = FT_GetBitMode(hc, &data);
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
data |= 0x10;
data |= 0x01;
ftStatus = FT_SetBitMode(hc, data, FT_BITMODE_CBUS_BITBANG); // CBUS0 を出力&H
if (ftStatus != FT_OK) {
    // エラー処理
    goto end;
}
}
end:
FT_W32_CloseHandle(hc);
return 0;
}
```



4.3.2.FTDI デバイスが複数接続されている場合

複数接続されている場合は、リセットしたい FTDI デバイスのシリアル番号を指定します。
そのシリアル番号の FTDI デバイスの GPIO を利用してリーダー・ライターにリセット信号を入れます。
FTDI デバイスのシリアル番号の入手方法は後述します。

```
        :  
        :  
    // 接続されている台数を入手  
    ftStatus = FT_ListDevices(&numDevs, NULL, FT_LIST_NUMBER_ONLY);  
    if (ftStatus != FT_OK) {  
        // エラー処理  
        return 0;  
    }  
    if (numDevs < 1) { // 1 台も接続されていなかったらエラー  
        // エラー処理  
        return 0;  
    }  
    memset(Buffer, 0, sizeof(Buffer));  
    memcpy(Buffer, argv[1], strlen(argv[1]));  
    hc = FT_W32_CreateFile((LPCTSTR)Buffer, GENERIC_READ | GENERIC_WRITE, 0, NULL,  
        OPEN_EXISTING, FILE_ATTRIBUTE_NORMAL | FILE_FLAG_OVERLAPPED |  
        FT_OPEN_BY_SERIAL_NUMBER, NULL);  
    if (hc == INVALID_HANDLE_VALUE) {  
        // エラー処理  
        return 0;  
    }  
  
    // 以下 リセット処理  
    memset(&ftData, 0, sizeof(FT_PROGRAM_DATA));  
        :  
        :
```



引数で渡されたシリアル番号



シリアル番号指定でハンドルを入手する

4.4. FTDI デバイスのシリアル番号取得アプリ

```
#include "stdafx.h"
#include <stdlib.h>
#include "ftd2xx.h"

int main(int argc, char *argv[])
{
    FT_STATUS ftStatus;
    DWORD numDevs, Flags, ID, Type, LocId, i;
    char SerialNumber[16], Description[64];
    FT_HANDLE ftHandle;

    // 接続されている台数を入手
    numDevs = 0;
    ftStatus = FT_CreateDeviceInfoList(&numDevs);
    if (ftStatus == FT_OK)
    {
        printf("%nNumber of devices is %d%n%n", numDevs); // 台数を画面表示
    }
    else {
        // エラー処理
        return 0;
    }
    // シリアルナンバーを入手する
    for (i = 0; i < numDevs; i++) {
        memset(SerialNumber, 0, sizeof(SerialNumber));
        ftStatus = FT_GetDeviceInfoDetail(i, &Flags, &Type, &ID, &LocId, SerialNumber,
            Description, &ftHandle);
        printf("%d : %s%n", i + 1, SerialNumber); // シリアル番号を画面表示
    }
    return 0;
}
```

以上
