

デバイスアートツールキット
シリアル通信モジュール

SC02

ソフトウェア マニュアル Rev. 1.0



2015 年 8 月

アークデバイス

目次

1. 概要	3
1.1 プログラム作成に必要なファイル	3
1.2 プログラム開始から終了まで	3
1.3 送受信データサイズ	4
2. クラス SC02	5
2.1 メンバ変数	5
2.2 SC02::Open	5
2.3 SC02::Close	5
2.4 SC02::Reset	6
2.5 SC02::Ping	6
2.6 SC02::SetID	6
2.7 SC02::GetID	7
2.8 SC02::GetVersion	7
2.9 SC02::Blink	7
2.10 SC02::SetParam	8
2.11 SC02::GetParam	8
付録	8
・ CM_PARAM 型構造体	8
注意事項	9
改版履歴	9

1. 概要

DATK (Device Art Tool Kit) のシリアル通信モジュール SC02 にはブートローダプログラム SC02_Boot, ユーザプログラムとして制御プログラム SC02_Ctrl が出荷時に書き込まれています。ブートローダは自己書き換え機能を持ちユーザプログラムの書き換えが可能です。SC02 ではまずブートローダが起動します。自己書き換への指令がなければ、次にユーザプログラムである制御プログラム SC02_Ctrl が呼び出されます。

SC02 をパソコンから通信制御するソフトウェアを制作するには、制御プログラム SC02_Ctrl に対応したクラス SC02 を用います。クラス SC02 は Windows の Visual C++(Visual Studio 2008) のコンソールアプリ用です (Unicode には対応していません)。基本的にファームウェアの制御プログラム SC02_Ctrl とクラス SC02 は同じバージョンのものを使用します。本マニュアルの対象は Ver.2.1 です。

SC02 のプログラム (ファームウェア・パソコン側ソフトウェア) はオープンソースになっておりますので、用途・環境に応じて適宜書き換えてご利用ください。

1.1 プログラム作成に必要なファイル

プログラムを作成する場合、以下のファイルが必要となります。

SC02.h	クラス SC02 のヘッダファイル
SC02.cpp	クラス SC02 のソースファイル
ControlModule.h	クラス ControlModule のヘッダファイル
ftd2xx.h	FTDI 社 D2xx 用のヘッダファイル
ftd2xx.lib	FTDI 社 D2xx 用のライブラリ
ftd2xx.dll	FTDI 社 D2xx 用のダイナミックリンクライブラリ

SC02 は FTDI 社の USB チップ FT245RL を使用しているため、USB のデバイスドライバと専用ライブラリが必要となります。FTDI 社のホームページより D2XX ドライバをダウンロード・インストールしてください。ダウンロードした D2XX ファイルにはドライバ本体のほかに、ftd2xx.h, ftd2xx.dll, ftd2xx.lib が一括して含まれています。

DATK の各モジュール用のクラスは基底クラス ControlModule を継承しているため、クラス ControlModule 用のファイルが必要となります。

1.2 プログラム開始から終了まで

通常 SC02 には CM01 などの制御モジュールを接続して使用しますが、もっとも簡単な動作例としてここでは SC02 を単体で使う例を示します。ID=1 のモジュール

ルをオープンして、基板上の LED の点滅指令を出すプログラムです。エラー処理は省略しています。詳しくはサンプルプログラムのソースコードをご覧ください。

```

SC02 sc;           // クラスの生成

sc.Open( 1 );     // ID = 1 でオープン
sc.Blink( 3 );   // LED を 3 回点滅
sc.Close();      // クローズ
    
```

1.3 送受信データサイズ

SC02 では複数の制御モジュールの送受信をまとめて行うことで通信効率を向上できます。しかしながら一度に送受信できるデータ量には限界があり、SC02 では通信する全データサイズを送・受信ともに 256 バイト以内とする必要があります。

2 台の制御モジュールを SC02 に連結した場合の送信パケットの構成を表 1 に受信パケットの構成を表 2 に示します。送受信のデータサイズは各制御モジュールごとに異なるため、モジュールの構成により計算が必要となります。計算の詳細は各制御モジュールのソフトウェアマニュアルをご参照ください。

表 1 SC02 送信パケット

項目	サイズ (バイト)
コマンド	1
送受信デバイス数	1
モジュール 1 のパケット	
ポート番号	1
ID	1
送信サイズ	2
受信サイズ	2
コマンド	1
データ	X
チェックサム	1
モジュール 2 のパケット	

表 2 SC02 受信パケット

項目	サイズ (バイト)
モジュール 1 のデータ	
ポート番号	1
ID	1
受信サイズ	2
データ	X
ステータス	1
チェックサム	1
モジュール 2 のパケット	
ステータス	1

2. クラス SC02

クラス SC02 はクラス ControlModule を継承しています。このためクラス ControlModule の変数や関数がメンバとなりますが、実際には使用されず無効となる変数や機能もあります。

2.1 メンバ変数

BYTE bPort;	SC02の接続されているポート（無効）。
BYTE bID;	SC02のID。
BYTE bStatus;	SC02本体及びパソコンのステータス。
BYTE bVersion;	ファームウェアバージョン。
char Error[256];	エラーメッセージ。

2.2 SC02::Open

書式

```
BOOL Open( BYTE bModuleID );
```

引数

BYTE bModuleID SC02 の ID (0~255)

戻値

成功 : TRUE

失敗 : FALSE

動作

SC02 をオープンします。bModuleID に CM_MASTER_ID (255) を指定すると、すべての ID の制御モジュールが応答します。ID が分からない場合や、制御モジュールが 1 枚しか使用しない場合に CM_MASTER_ID を設定します。

2.3 SC02::Close

書式

```
BOOL Close( void );
```

引数

なし

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 をクローズします。

2.4 SC02::Reset

書式

```
BOOL Reset( void );
```

引数

なし

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 をリセットします。接続されている他のモジュールもすべてリセットされます。また USB もリセットされます（ケーブルを再接続したときと同じ動作）。

2.5 SC02::Ping

書式

```
BOOL Ping( void );
```

引数

なし

戻値

成功 : TRUE, 失敗 : FALSE

動作

動作チェック用のコマンドです。パソコン側から送信した 1 バイトデータと同じデータが SC02 から返信されるかを確認します。

2.6 SC02::SetID

書式

```
BOOL SetID( BYTE bNewID );
```

引数

BYTE bNewID 新しい ID (0~254)

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 に新しい ID を設定します。新しい ID は SC02 の EEPROM に書き込まれるため、次回起動時にも有効になります。ID=255(CM_MASTER_ID)は指定できません。

2.7 SC02::GetID

書式

```
BOOL GetID( void );
```

引数

なし

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 に設定されている ID を取得し bID に書き込みます。

2.8 SC02::GetVersion

書式

```
BOOL GetVersion( void );
```

引数

なし

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 のプログラムバージョンを取得し bVersion に書き込みます。
プログラムバージョンが 2.1 の場合 bVersion = 0x21 となります。

2.9 SC02::Blink

書式

```
BOOL Blink( BYTE bNum );
```

引数

BYTE bNum 点滅回数

戻値

成功 : TRUE, 失敗 : FALSE

動作

bNum に設定した回数 LED が点滅します。チェック時などに使用します。

2.10 SC02::SetParam

書式

```
BOOL SetParam( PCM_PARAM pParam );
```

引数

PCM_PARAM pParam 制御パラメータ構造体のポインタ

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 に制御パラメータを設定します。fComTimeout はモジュール間通信のタイムアウト値となり USB のタイムアウトは fComTimeout +1 秒となります。SC02 では制御タイムアウトは使用されていないので fCtrlTimeout の値は無効となります。

2.11 SC02::GetParam

書式

```
BOOL GetParam( PCM_PARAM pParam );
```

引数

PCM_PARAM pParam 制御パラメータ構造体のポインタ

戻値

成功 : TRUE, 失敗 : FALSE

動作

SC02 に設定されている制御パラメータを取得します。

付録

・ CM_PARAM 型構造体

メンバ変数

- | | |
|----------------------------|--|
| float dt; | 制御周期をミリ秒単位で指定します。デフォルトは1.0ミリ秒です。 |
| float fCtrlTimeout; | 制御タイムアウトをミリ秒単位で指定します。デフォルトは1000ミリ秒です。0でタイムアウトは無効になります。 |
| float fCommTimeout; | パソコン側，モジュール側双方の送受信通信タイムアウトをミリ秒単位で指定します。デフォルトは1000ミリ秒です。0でタイムアウトは無効になります。 |

注意事項

- ・本ソフトウェアは、ソースコードをオープンにしておりますが、商業利用・無断転載はおやめください。また、本ソフトウェアはお客様が自由に改変して御使用いただけますが、お客様のソフトウェア改変に伴う事故や故障等に関して当方では、一切の責任を負いません。
- ・本ソフトウェアにより生じるお客様の製品に起因して発生したいかなる損害に対しても当方では、一切の責任を負いません。
- ・本ソフトウェアの仕様は改良のため予告なく変更することがあります。

改版履歴

2015 年 8 月 初版

お問い合わせはメールにてお願いいたします。

アークデバイス

E-mail: info@arcdevice.com

URL: <http://www.arcdevice.com/>