

4080-601606
13.56MHz RFID リーダ
プログラミングマニュアル

ユニテック・ジャパン（株）
2006年1月初版
オリジナル V 1.0 7 November 2005

1 概要

本書は、4080-601606 13.56MHz RFID モジュールのプログラミングをするための Unitech SDK の使い方を説明しています。この RFID カードは ISO15693 仕様をサポートしていますので、ISO 15693 対応のタグ、例えば Tag-It、I-Code、Performa、MicrolD 350/450 を読むために本カードを使用することができます。サポートされているタグからタグ ID とプロトコルの読み込み、さらに完全な読み込み機能提供のために C++アプリケーションを実現するのに SDK を使用することができます。

2 SDK とサンプルプログラム

以下の Web サイトから SDK とサンプルプログラムをダウンロードすることができます。

http://adc.unitech.com.tw/pub/cs/sdk/4080-601606/4080-601606_dk.zip

Unzip すると、以下の 4 つのファイルと 1 つのフォルダがあります。

UnitechRFID.CAB	: RFID デモプログラム、RFIDSample.EXE と DLL
RFIDAPI.DLL	: RFID アプリケーション用 DLL
RFIDAPI.LIB	: C++ プログラミング用 LIB ファイル
RFIDAPI.H	: C++ プログラミング用 Include ファイル
RFID_Sample_Souce	: RFIDSample のソースコード

UnitechRFID.CAB を以下のステップで PA962/966 に インストールすることによって RFID をテストすることができます。

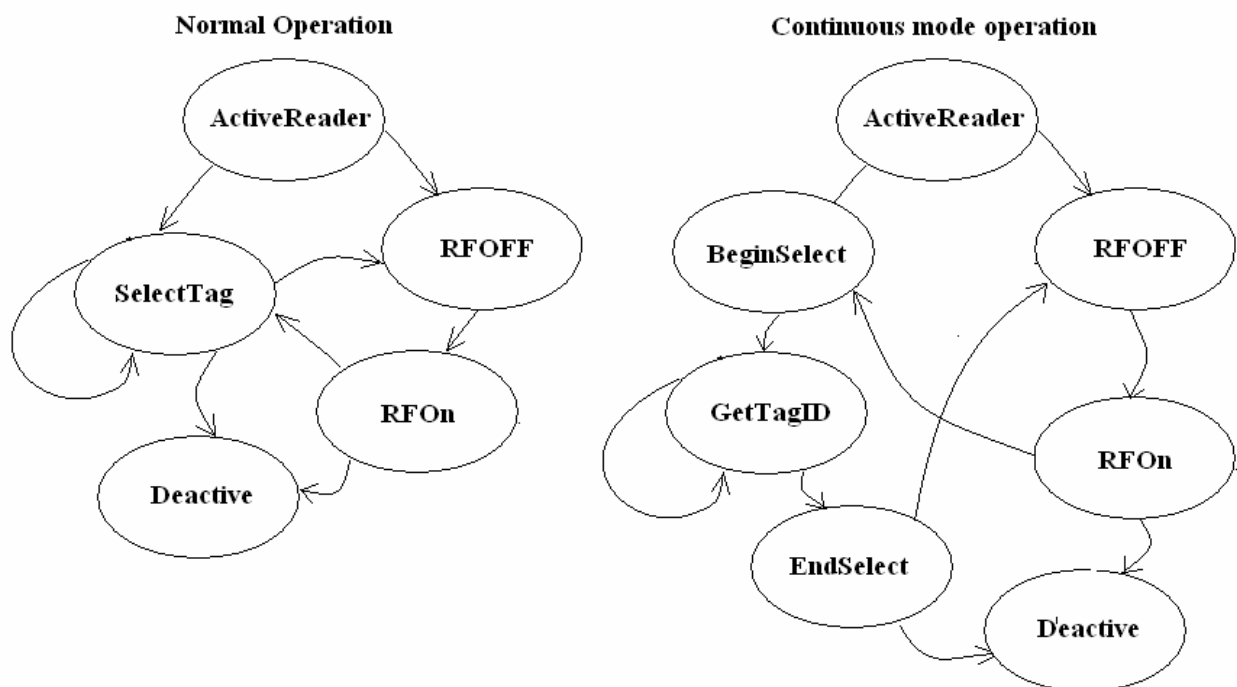
1. UnitechRFID.CAB を PA962/PA966 にインストールします。
2. UnitechRFID.CAB が保存されたフォルダを見るためにエクスプローラを起動します。
3. UnitechRFID.CAB をダブルタップしてインストールします。

3 動作フロー

4080-601606 RFID リーダは、PA96x に挿入した後はシリアルポートに接続されたデバイスとして扱われます。PA96x はこのシリアルポートを COM6 に固定します。しかし、COM6 を通してリーダと通信する方法について気にする必要はありません。リーダのコントロールは Unitech の API を使用するだけです。

リーダの設計に従い、最初のステップとして ActiveReader API によってリーダをアクティブにして、そしてリーダの使用が不要になった場合にアクティブにするのをやめる(でアクティブ)にする必要があります。動作中、省電力のために無線をオフにするのに RFOff を使用することができます。そしてタグを読むために無線をオンにするのに RFOn をコールします。

RFID の動作フローには、ノーマルモードと連続モードの2種類があります。アプリケーションプログラムの実現方法を理解するために以下のフローチャート API リストを参照して下さい。



a. ノーマルモード

リーダは SelectTag を実行するまで RFID タグを読みません。これはタグを読む・読まないに関わらずすぐに戻ります。アプリケーションは RFID タグが読みとり距離に近づいた場合 SelectTag API からタグのユニークな ID を得ます。そして、アプリケーションは RFID タグを読みとり (ReadBlock) または RFID タグに書き込みます。

b. 連続モード

読みとり可能な範囲にあるタグを連続的に読み、データを内部バッファに入れます。同じタグはタグが常にリーダの近くにある場合何度も読まれます。アプリケーションはすべてのタグ ID を得るために GetTagID (ワンタイム・ワンタグ ID) をコールする必要があります。

4 API リスト

API を説明する前に、セクション 4.1 はパラメータまたは戻りコードとして各 API で使用される定数について説明しています。

4.1 パラメータまたは戻りコードの定数

- 本リーダーは ISO15693 をサポートしていますので、TagProtocol プロパティを使用することによって読みたいタグの種類を指定する必要があります。以下は使用可能な TagProtocol とその値です。

Unitech PA96x と TagProtocol プロパティは以下の値をサポートしています。

TAGIT	= 1
ICODE	= 2
Performa	= 3
MicroID 350	= 4
MicroID 450	= 5
AUTO	= 6
ISO15693	= 7
EPC	= 8

- エラーコードプロパティのサポートしている値:
 - ERR_CMD_SUCCESS = 0
 - ERR_CMD_READER_NOT_ACTIVE = 1
 - ERR_CMD_FAIL = 2
 - ERR_CMD_CANT_FIND_READER = 3
 - ERR_CMD_CANT_OPEN_PORT = 4
 - ERR_CMD_INVALID_PORT = 5
 - ERR_CMD_INVALID_PROTOCOL = 6
 - ERR_CMD_INVALID_PARAMETERS = 7
 - ERR_CMD_UNSUPPORTED_COMMAND = 8
 - ERR_CMD_NO_TAG = 9
 - ERR_CMD_IN_SCAN_MODE = 10
 - ERR_CMD_NOT_IN_SCAN_MODE = 11
 - ERR_CMD_READ_FAIL = 12
 - ERR_CMD_WRITE_FAIL = 13

4.2 API テーブル

以下はAPIの全リストです。

No	関数名	動作	ステータス
一般関数			
1	ActivateReader	COMポートを開き、リーダをアクティブにする	
2	Deactivate	リーダをディアクティブにして、COMポートを閉じる	
3	SelectTag	現在のタグについての情報を検索する	
4	GetFirmware	リーダからファームウェアラベルを読む	
5	RFOOn	リーダのRFフィールドをオンにする	
6	RFOff	リーダのRFフィールドをオフにする	
7	ReadBlocks	サポートされたRFIDタグからブロックのリストを読む	
8	WriteBlocks	サポートされたRFIDタグのブロックのセットにデータを書く	
9	GetErrStringByCode	エラーコードをテキストメッセージに変換する	
連続モード関数			
10	BeginSelect	プロトコルのために連続スキャンモードを有効にする	
11	GetTagID	連続スキャンモードで、リーダがタグを見つけたかどうかをチェックする。あったら、次のタグIDスキャンに戻る。	
12	EndSelect	連続スキャンモードを無効にする。タグのスキャンが終わった時、標準動作モードにリーダを戻すためにこの関数を使用する。	

4.3 一般関数

4.3.1 リーダをアクティブにする

関数の説明:

COMポートを開き、リーダをアクティブにする

関数コール:

```
LONG ActivateReader(int iCom)
```

パラメータ(入力)

PA962/PA966 では 6 に固定

戻りコード:

エラーグループプロパティを参照

4.3.2 リーダをディアクティブにする

関数の説明:

リーダーをディアクティブ(作動しない)にして COM ポートを閉じる

関数コール:

void Deactivate()

戻りコード:

なし

4.3.3 タグ ID 選択

関数の説明:

リーダーは使用可能なタグを一つ自動的に選択して、タグ ID を読む

関数コール:

long SelectTag(int iTagProtocol,LPTSTR &lpBuf)

パラメータ(入力)

iTagProtocol : TagProtocol プロパティ参照

パラメータ(出力)

lpBuf : タグ ID 文字列

戻りコード:

エラーコードプロパティ参照

コードの例:

```
LPTSTR strTAG;  
int protocol = 6;  
long ErrCode = SelectTag(protocol,strTAG);  
if (ErrCode == 0) {  
}
```

4.3.4 ファームウェアバージョンを得る

関数の説明:

リーダーのファームウェアバージョンを読む

関数コール:

long GetFirmware(LPTSTR &lpBuf)

パラメータ(出力)

lpBuf : ファームウェア文字列

戻りコード:

エラーコードプロパティ参照

コードの例:

```
LPTSTR strFirmware;  
long ErrCode = GetFirmware(strFirmware);  
if (ErrCode == 0) {  
}
```

4.3.5 RFOn

関数の説明:

無線をオンにする

関数コール:

long RFOn()

戻りコード:

エラーコードプロパティ参照

4.3.6 RFOff

関数の説明:

無線をオフにする

関数コール:

long RFOff()

戻りコード:

エラーコードプロパティ参照

4.3.7 ReadBlocks

関数の説明:

RFID タグから書き込み済みのデータブロックを読む

関数コール:

long ReadBlocks(int iTagProtocol,int StartIndex,byte BlocksData[])

パラメータ(入力)

iTagProtocol : TagProtocol プロパティ参照

StartIndex : 読み込み開始ブロックインデックス

パラメータ(出力)

BlocksData : 読み込みブロックデータ

戻りコード:

エラーコードプロパティ参照

コードの例:

```
byte data[4];  
int protocol = 7;  
int iBlockIndex = 0;  
long ErrCode = ReadBlocks(protocol,iBlockIndex,data);
```

```
if (ErrCode == 0) {  
}
```

4.3.8 WriteBlocks

関数の説明:

RFID タグにデータブロックを書く

関数コール:

long WriteBlocks(int iTagProtocol,int StartIndex,int numBlocks,byte BlocksData[])

パラメータ(入力)

iTagProtocol : TagProtocol プロパティ参照

StartIndex : 書き込み開始ブロックインデックス

numBlocks : 書き込みブロック数

パラメータ(出力)

BlocksData : 書き込みブロックデータ

戻りコード:

エラーコードプロパティ参照

コードの例:

```
byte data[4];  
int protocol = 7;  
int iBlockIndex = 0;  
long ErrCode = WriteBlocks(protocol,iBlockIndex,1,data);  
if (ErrCode == 0) {  
}
```

4.3.9 GetErrStringByCode

関数の説明:

エラーコードをテキストメッセージに変換

関数コール:

GetErrStringByCode(int ErrCode,LPTSTR &lpBuf)

パラメータ(入力)

ErrCode : エラーコードプロパティ参照

パラメータ(出力)

lpBuf : エラーメッセージ

戻りコード:

エラーコードプロパティ参照

4.4 連続モード

4.4.1 連続モード開始

関数の説明:

あらかじめ定義したプロトコルによってシステムバッファに連続的にタグを読み込むことを有効にする

関数コール:

long BeginSelect(int iTagProtocol)

パラメータ(入力)

iTagProtocol : TagProtocol プロパティ参照

戻りコード:

エラーコードプロパティ参照

4.4.2 タグ ID を得る

関数の説明:

BeginSelect を実行した後で、タグ ID をシステムバッファに連続的に読み込みます。同じタグが読みとり範囲内に置かれていた場合、タグが除けられるか EndSelect が呼ばれるまでシステムバッファに何度も読まれます。

関数コール:

long GetTagID(LPTSTR &lpBuf)

パラメータ(出力)

lpBuf : タグ ID 文字列

戻りコード:

エラーコードプロパティ参照

4.3.1 連続モード終了

関数の説明:

連続モードを無効にする

関数コール:

long EndSelect()

戻りコード:

エラーコードプロパティ参照