

製品仕様書

製品名 リーダライタ制御用DLL

型 式 リーダライタ制御用DLL

仕様書名 リファレンスマニュアル

文書番号 ME329-6806A05

総構成 リファレンスマニュアル ME329-6806A

三和ニューテック株式会社
開発部

2013年 5月 20日

承認	審査	作成
----	----	----



長倉



近藤



川越

変更経歴書

版	記号	年月日	項番	変更内容	変更理由	担当
00		2003-09-10		新規発行		長倉
01		2005-04-21	1.1 3.5 3.6 3.10 3.15 3.21 3.25	対象機種に CR-S31 シリーズ追加 リセット指定追記 " CRWLReadData2 コマンドの追加 (3トラック対応のため) CRWLWriteData2 コマンド追加 (3トラック対応のため) 印字設定コマンドカード種類追加 プリントコマンドの印字データ 終了条件追記	対象機種追加のため 仕様追加のため " " " " 仕様明確化のため "	長倉
02		2005-06-24	1.2 3.5 3.6 3.21 3.22 4.1	動作確認 OS に、 WindowsXP Professional Service Pack 2 を 追記 CRWLInitialize、CRWLInitialize2 コマンドにリセット指定の追加 (S 3 1 のみ対象) CRWLSetPrint、CRWLSetPrint2 コ マンドにカード種類追加 (&H38) (S 3 1 のみ対象) 付録(参考資料)の内容を見直し、 4 . 1 項プログラミング上の注意 点に移動。	動作確認を実施した ため 仕様追加のため " 内容見直し	長倉
03		2011-04-12	1.2 1.2	動作確認 OS に、 Windows7 Enterprise 64bit 版 を追記 OS の表記方法を変更	動作確認を実施した ため 表記見直しのため	角
04		2011-05-31	1.2 3.9 3.10 3.12 3.38 3.39 3.41 4.1 4.2.1 4.2.2	動作確認 OS に、 Windows7 Enterprise 32bit 版 を追記 磁気データを読み込む変数の領域 を 2 0 7 バイト以上確保へ変更	動作確認を実施した ため 3 T r 対応に伴う修 正漏れ対応	角
05	△	2013-05-20	1.2 2.1 - 3 . の 表 3-1 3.31	動作確認 OS で、 Windows8 を追記 ハードウェア環境の CPU、メモリ、 ハードディスクを削除 32 ビット Windows 用の DLL として 実装 . . . を削除 改定履歴と各ページの著作権を削 除 CRWTakeInCardDispenserS31 の追 加	表記見直しのため 表記見直しのため 表記見直しのため API 追加対応のため	川越

目 次

はじめに	1
1 ハードウェアとソフトウェア	2
1.1 対象機種	2
1.2 使用環境	2
2 ソフトウェア構成	3
2.1 構成と方式	3
3 ライブラリ関数	4
3.1 CRWInitialize	6
3.2 CRWInitialize2	7
3.3 CRWSetCommunicateRetry	8
3.4 CRWTerminate	9
3.5 CRWInitialize (コマンドコード 10h)	10
3.6 CRWInitialize2 (コマンドコード 10h)	11
3.7 CRWLReadStatus (コマンドコード 20h)	12
3.8 CRWLReadData (コマンドコード 30h)	13
3.9 CRWLReadDataL (コマンドコード 33h)	14
3.10 CRWLReadData2 (コマンドコード 33h)	15
3.11 CRWLGetData (コマンドコード 35h)	17
3.12 CRWLCancel (コマンドコード 40h)	18
3.13 CRWLWriteData (コマンドコード 50h)	19
3.14 CRWLWriteDataL (コマンドコード 53h)	20
3.15 CRWLWriteData2 (コマンドコード 53h)	21
3.16 CRWLWriteDataDischargeCard (コマンドコード 55h)	23
3.17 CRWLWriteBinaryData (コマンドコード 5Ch)	24
3.18 CRWLPrint (コマンドコード 70h)	25
3.19 CRWLPrint20Lines (コマンドコード 73h)	26
3.20 CRWLPrintDischargeCard (コマンドコード 75h)	27
3.21 CRWLSetPrint (コマンドコード 78h)	28
3.22 CRWLSetPrint2 (コマンドコード 78h)	30
3.23 CRWLRegisterFont (コマンドコード 7Ah)	32
3.24 CRWLPrintImage (コマンドコード 7Bh)	32
3.25 CRWLPrintL (コマンドコード 7Ch)	35
3.26 CRWLPrintBarCode (コマンドコード 7Eh)	37
3.27 CRWLDischargeCard (コマンドコード 80h)	40
3.28 CRWLDischargeCardDispenser (コマンドコード 90h)	41
3.29 CRWLCleaning (コマンドコード A0h)	42
3.30 CRWLTakeInCardDispenser (コマンドコード B0h)	43
3.31 CRWLTakeInCardDispenserS31 (コマンドコード B0h)	44
3.32 CRWLControlLED (コマンドコード C0h)	45
3.33 CRWLSetRetry (コマンドコード C1h)	46
3.34 CRWLSetRtcDate (コマンドコード E1h)	47
3.35 CRWLGetVersion (コマンドコード F0h)	48
3.36 CRWLGetRtcDate (コマンドコード F1h)	49
3.37 CRWLBattCheck (コマンドコード F5h)	50
3.38 CRWLTestPrint	51
3.39 CRWLReadVariableLengthDataL	52
3.40 CRWLGetVariableLengthData	53
3.41 CRWLWriteVariableLengthDataL	54
3.42 CRWLEnq	55
3.43 CRWLEnq2	56
4 プログラミング	57
4.1 プログラミング上の注意点	57
4.2 コーディングサンプル	60
4.2.1 カードの新規発行例	61
4.2.2 カードの更新例	63

はじめに

三和ニューテック株式会社製の、磁気カードリーダー・ライタ制御用ドライバ（以降、本ドライバ又はリーダーライタドライバ）は、三和ニューテック株式会社製の磁気カードリーダー・ライタ（以降、リーダーライタ）を使用したシステムのアプリケーション開発を容易にするためのライブラリです。


本ドライバを利用することで、アプリケーションとリーダーライタ間で行う送受信処理を気にすることなく、アプリケーションから単純な関数呼び出しによってリーダーライタを制御することが可能となります。

注意）アプリケーションを作成する場合には、本マニュアルだけでなく、必ず機種ごとのソフトウェア仕様書を参照するようにして下さい。レスポンス、引数、動作等が本マニュアルに記載されているものと異なる事があります。ソフトウェア仕様書の記載に従い作成して下さい。

1 ハードウェアとソフトウェア

本ドライバを利用可能なハードウェア、ソフトウェア環境は、以下の通りとします。

1.1 対象機種

A B S - 1 2 3 1 L シリーズ
(A B S - 1 2 3 1 L / L R 、 A B S - 1 2 3 1 A / A R)
A B S - A 3 1
A B S - S 3 1
C R P - 1 2 3 1 L シリーズ (組込用)
(C R P - 1 2 3 1 L / L R 、 C R P - 1 2 3 1 A / A R)
C R - S 3 1 シリーズ (組込用) 
(C R - S 3 1 / S 3 1 R)



リーダライタのオプション機能は対象範囲外とします。

1.2 使用環境

(1) ソフトウェア環境

動作確認を行った環境です。下記以外の環境については、アプリケーション開発時に確認を行って下さい。

・ O S

Microsoft 日本語 Windows98 Second Edition
Microsoft 日本語 Windows2000 Service Pack 4
Microsoft Windows XP Professional 日本語版
Microsoft Windows XP Professional 日本語版 Service Pack 2
Microsoft Windows 7 Enterprise 日本語版 (32bit 版)
Microsoft Windows 7 Enterprise 日本語版 (64bit 版)
Microsoft Windows 8 日本語版 (32bit 版) 
Microsoft Windows 8 日本語版 (64bit 版) 

Microsoft 社のサポートが終了となった OS については、保証対象外とします。

・ 開発環境

Microsoft VisualBasic 6.0 SP5

(2) ハードウェア環境

・ P C 本体	P C / A T 互換機
・ C P U	P e n t i u m 1 3 3 M H z 以上
・ メモリ	3 2 M バイト以上
・ ハードディスク	2 M バイト以上 (ドライバ保存に必要な容量)
・ C O M ポート	(D - S u b 9 ピン、 1 2 0 0 b p s ~ 9 6 0 0 b p s) (A 3 1 : 1 2 0 0 b p s ~ 1 9 2 0 0 b p s とする) (S 3 1 : 1 2 0 0 b p s ~ 3 8 4 0 0 b p s とする)

C O M ポート 1 ~ 4 を使用してください。C O M ポート 5 以上には対応していません (U S B シリアル変換ケーブル等を使用した際も同様)。

2 ソフトウェア構成

リーダライタドライバとして以下のファイルを提供します。

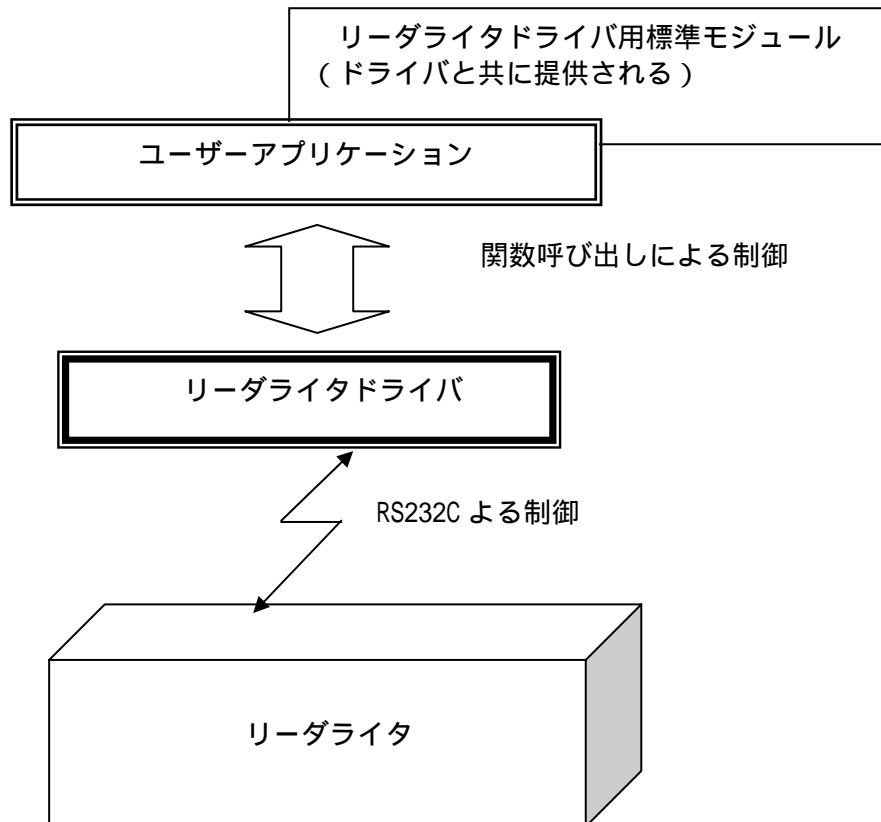
~~本ドライバは、32ビットWindows用のDLLとして実装されています。~~[△]

DLL本体

リーダライタドライバ用VB標準モジュール（インターフェース等を記述したもの）

2.1 構成と方式

本ドライバの、システム中の位置付けを以下に示します。



3 ライブラリ関数

本ドライバは、ドライバの初期化、終了処理関数とリーダライタのコマンドに1対1で対応する関数を提供します。ここでは、アプリケーション側(Visual Basic)から見たインターフェースを記述します。

関数実行時のリーダライタ動作の詳細については、リーダライタのソフトウェア仕様書を(表 3-1 関数一覧のコマンドコードをもとに)参照してください。

なお、引数でString型や配列を渡す場合、関数呼び出し側で領域の確保をしておく必要があります。

表 3-1 関数一覧

番号	関数名	コマンドコード	内容	ABS 1231L	ABS A31	ABS S31	CR S31
1	CRWInitialize	-	ドライバの初期処理を行う	○	○	○	○
2	CRWInitialize2	-	ドライバの初期処理を行う	○	○	○	○
3	CRWSetCommunicateRetry	-	通信のリトライ回数を設定	○	○	○	○
4	CRWTerminate	-	ドライバの終了処理を行う	○	○	○	○
5	CRWLInitialize	10h	イニシャル	○	○	○	○
6	CRWLInitialize2	10h	イニシャル	×	○	○	○
7	CRWLReadStatus	20h	ステータスリード	○	○	○	○
8	CRWLReadData	30h	リード(旧)	○	×	×	×
9	CRWLReadDataL	33h	リード	○	○	○	○
10	CRWLReadData2	33h	リード(複数トラック対応)	×	×	○	○
11	CRWLGetData	35h	リードデータ要求	○	×	×	×
12	CRWLCancel	40h	キャンセル	○	○	○	○
13	CRWLWriteData	50h	ライト(旧) 7ビット	○	×	×	×
14	CRWLWriteDataL	53h	ライト	○	○	○	○
15	CRWLWriteData2	53h	ライト(複数トラック対応)	×	×	○	○
16	CRWLWriteDataDischargeCard	55h	ライト&排出(旧)	○	×	×	×
17	CRWLWriteBinaryData	5Ch	ライト(旧) 8ビット	○	×	×	×
18	CRWLPrint	70h	プリント(旧)	○	×	×	×
19	CRWLPrint20Lines	73h	プリント(旧) 20行	○	×	×	×
20	CRWLPrintDischargeCard	75h	プリント&排出(旧)	○	×	×	×
21	CRWLSetPrint	78h	印字設定	○	○	○	○
22	CRWLSetPrint2	78h	印字設定	○	○	○	○
23	CRWLRegisterFont	7Ah	外字登録	○	○	○	○
24	CRWLPrintImage	7Bh	イメージ印字	×	×	○	○
25	CRWLPrintL	7Ch	プリント	○	○	○	○
26	CRWLPrintBarCode	7Eh	バーコード印字	×	○	○	○
27	CRWLDischargeCard	80h	カード排出	○	○	○	○
28	CRWLDischargeCardDispenser	90h	カード収納	○	×	×	×
29	CRWLCleaning	A0h	クリーニング	○	○	○	○
30	CRWLTakeInCardDispenser	B0h	カード後部取込み	○	×	×	×
31	CRWLTakeInCardDispenserS31	B0h	カード後部取込み(S31用)	×	×	×	○
32	CRWLControlLED	C0h	LED制御(旧)	○	×	×	×
33	CRWLSetRetry	C1h	リライト回数設定(旧)	○	×	×	×
34	CRWLSetRtcDate	E1h	時計設定	×	○	○	○
35	CRWLGetVersion	F0h	バージョン取得	×	○	○	○
36	CRWLGetRtcDate	F1h	現在時刻取得	×	○	○	○
37	CRWLBattCheck	F5h	バッテリーチェック	×	×	○	○
38	CRWLTestPrint	-	テスト印字	○	○	○	○
39	CRWLReadVariableLengthDataL	-	可変長リード	×		○	○
40	CRWLGetVariableLengthData	-	可変長リードデータ要求	×		○	○
41	CRWLWriteVariableLengthDataL	-	可変長ライト	×		○	○
42	CRWLEng	-	レスポンスの再送要求	○		○	○
43	CRWLEng2	-	レスポンスの再送要求	○		○	○

○: 各機種において使用可能な関数 (■: 各機種で使用を推奨するコマンド)

×: 各機種において使用不可能な関数

(旧): 旧機種互換用のコマンド(なるべく使用しないで下さい)

各関数の引数 baResponse の値は、コマンド実行に対するレスポンスです。以下にレスポンスの内容を示します。なお、実行したコマンドにより返されるレスポンス値が異なります。詳細はリーダライタのソフトウェア仕様書を参照してください。

表 3-2 レスポンス 1 : baResponse (1)

値	意味
&H30	装置内カード無し
&H31	装置内カード有り 1
&H32	カード状態異常
&H33	装置内カード有り 2
&H34	挿入口カード残留

表 3-3 レスポンス 2 : baResponse (2)

値	意味
&H30	正常終了 (エラー無し)
&H31	リードエラー
&H32	ライトエラー
&H33	カード詰まり
&H34	搬送系モーターエラー
&H35	プリンタエラー
&H38	不正処理エラー
&H40	電池電圧異常
&H41	リーダライタシステムエラー
&H51	トラック 1 リードエラー
&H52	トラック 2 リードエラー
&H53	トラック 3 リードエラー
&H54	トラック 1 , 2 リードエラー
&H55	トラック 1 , 3 リードエラー
&H56	トラック 2 , 3 リードエラー

表 3-4 レスポンス 3 : baResponse (3)

値	意味
&H30	コマンド実行終了
&H32	コマンド実行不可
&H33	コマンド実行中
&H34	カード挿入待ち
&H35	後部取込み実行時、ディスペンサーにカード無し
&H36	ディスペンサー無し
&H37	カード満杯

3.1 CRWInitialize

ドライバ、リーダライタの初期処理を行う。

<構文>

```
Function CRWInitialize( ByRef baResponse As Byte, _  
                        ByVal IngPortNo As Long, ByVal IngSpeed As Long, _  
                        ByVal IngParity As Long, ByVal IngStop As Long, _  
                        ByVal IngWait As Long ) As Long
```

<引数>

baResponse レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）

IngPortNo COMポート番号（1：COM1 2：COM2 3：COM3 4：COM4）

IngSpeed通信速度 1200： 1 2 0 0 b p s 2400： 2 4 0 0 b p s
 4800： 4 8 0 0 b p s 9600： 9 6 0 0 b p s
 19200： 1 9 2 0 0 b p s 38400： 3 8 4 0 0 b p s
 19200bps は A 3 1 / S 3 1 のみ対象
 38400bps は S 3 1 のみ対象

IngParity パリティビット （0：パリティ無し 1：偶数パリティ 2：奇数パリティ）

IngStop ストップビット （1：1ビット 2：2ビット）

IngWait タイムアウト時間（ミリ秒）
 コマンド送信後のレスポンス待ち時間（リーダライタの動作が終了するまで
 の時間を基に、余裕を持って設定して下さい） 例）10秒（10000）

<戻り値>

CRW_SUCCESS（0） 正常終了
CRW_ERR_HOOKUP リーダライタが接続されていないか、電源が OFF である
CRW_ERR_COMPORT 通信ポートがオープンできない。エラーが発生した。
CRW_ERROR その他のエラー

<説明>

ドライバの初期処理、通信ポートのオープン、リーダライタの初期化コマンド（コマンドコード：10h）の発行を行う。

アプリケーションから本ドライバの関数を呼び出す前に、呼び出さなければならない。

リーダライタ内にカードがある場合は排出する。

<対象機種>

全機種

3.2 CRWInitialize2

ドライバの初期処理を行う。

< 構文 >

```
Function CRWInitialize2( ByVal IngPortNo As Long, ByVal IngSpeed As Long, _
                        ByVal IngParity As Long, ByVal IngStop As Long, _
                        ByVal IngWait As Long ) As Long
```

<引数>

IngPortNo COMポート番号 (1 : COM 1 2 : COM 2 3 : COM 3 4 : COM 4)

IngSpeed通信速度

1200 :	1 2 0 0 b p s	2400 :	2 4 0 0 b p s
4800 :	4 8 0 0 b p s	9600 :	9 6 0 0 b p s
19200 :	1 9 2 0 0 b p s	38400 :	3 8 4 0 0 b p s)

19200bps は A 3 1 / S 3 1 のみ対象
38400bps は S 3 1 のみ対象

IngParity	パリティビット	(0 : パリティ無し 1 : 偶数パリティ 2 : 奇数パリティ)
IngStop	ストップビット	(1 : 1 ビット 2 : 2 ビット)
IngWait	タイムアウト時間	(ミリ秒)
		コマンド送信後のレスポンス待ち時間 (リーダライタの動作が終了するまでの時間を基に、余裕を持って設定して下さい) 例) 10 秒 (10000)

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_COMPORT	通信ポートがオープンできない。エラーが発生した。
CRW_ERROR	その他のエラー

< 説明 >

ドライバの初期処理、通信ポートのオープンを行う。

<対象機種>

全機種

3.3 CRWSetCommunicateRetry

通信リトライの回数設定を行う。

< 構文 >

Function CRWSetCommunicateRetry(ByVal lngRetry As Long) As Long

< 引数 >

lngRetry リトライ回数 0 ~ 5 回

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERROR	その他のエラー (リトライ回数が範囲外)

< 説明 >

通信上のエラー (パリティ異常 / タイムアウト等) が発生した時のリトライ回数を設定する。

(通信エラー時に、設定した回数分、自動的に通信リトライする)

設定しない場合、リトライ回数は 0 となる (CRWInitialize または CRWInitialize2 で初期化を行ったときに、初期値 (0) が設定される)。

CRWInitialize または CRWInitialize2 でドライバを初期化後、リトライ回数の設定を行うこと。

< 対象機種 >

全機種

3.4 CRWTerminate

ドライバの終了処理を行う。

<構文>

```
Sub CRWTerminate( )
```

<引数>

<戻り値>

<説明>

通信ポートのクローズ、ドライバの終了処理を行う。

<対象機種>

全機種

3.5 CRWInitialize (コマンドコード 10h)

リーダライタの初期化を行う。

< 構文 >

Function CRWInitialize(ByRef baResponse As Byte, ByVal bMode As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bMode 初期化方法の指定

&H30 イニシャルのみ実行

&H31 イニシャル実行後、搬送路内にカードがある場合排出

&H32 リセット指定 (S 3 1 のみ対象。詳細はソフトウェア仕様書を参照。)

< 戻り値 >

CRW_SUCCESS (0) 正常終了

CRW_ERR_HOOKUP リーダライタが接続されていないか、電源が OFF である

CRW_ERR_INITIALIZE ドライバが使用可能状態で無い

CRW_ERR_COMPORT 通信ポートにエラーが発生した

CRW_ERROR その他のエラー

< 説明 >

リーダライタの初期化を行う。

< 対象機種 >

全機種

3.6 CRWInitialize2 (コマンドコード 10h)

リーダーライタの初期化を行う。

< 構文 >

```
Function CRWInitialize2( ByRef baResponse As Byte, _  
                        ByVal bMode As Byte, ByVal bAbort As Byte ) As Long
```

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bMode 初期化方法の指定
 &H30 イニシャルのみ実行
 &H31 イニシャル実行後、搬送路内にカードがある場合排出
 &H32 リセット指定 (S 3 1 のみ対象。詳細はソフトウェア仕様書を参照。)

bAbort 不正処理指定
 &H30 不正処理レスポンス無し
 &H31 不正処理レスポンス有り

< 戻り値 >

CRW_SUCCESS (0) 正常終了
CRW_ERR_HOOKUP リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE ドライバが使用可能状態で無い
CRW_ERR_COMPORT 通信ポートにエラーが発生した
CRW_ERROR その他のエラー

< 説明 >

リーダーライタの初期化を行う。

< 対象機種 >

A B S - A 3 1 / A B S - S 3 1

3.7 CRWLReadStatus (コマンドコード 20h)

リーダライタのステータスを取得する。

< 構文 >

```
Function CRWLReadStatus( ByRef baResponse As Byte ) As Long
```

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

リーダライタのステータスを取得する。

カード排出後、本関数でカードが取り除かれたこと (baResponse(1) が &H30) を確認する必要がある。
この確認を行わないと次のカードの取込み動作が行われない。

< 対象機種 >

全機種

3.8 CRWLReadData (コマンドコード 30h)

カードの取込み、磁気データのリードを行う。

< 構文 >

Function CRWLReadData(ByRef baResponse As Byte, ByVal bMode As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bMode

読み取り方法の指定

&H30 7 b i t 磁気リード (データチェック有り)

&H31 磁気リード (データチェック無し)

&H32 8 b i t 磁気リード (データチェック有り)

< 戻り値 >

CRW_SUCCESS (0) 正常終了

CRW_ERR_HOOKUP リーダライタが接続されていないか、電源が OFF である

CRW_ERR_INITIALIZE ドライバが使用可能状態で無い

CRW_ERR_COMPORT 通信ポートにエラーが発生した

CRW_ERROR その他のエラー

< 説明 >

カードの取込み、磁気データのリードを行う。

カードが装置内に有る場合は磁気データのリードを行う。

カードが装置内に無い場合は、baResponse(1)に &H 3 0、baResponse(3)に &H 3 4 を返し、カード挿入待ちとなる。

カード挿入待ちのキャンセルは、CRWLCancel で行う。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.9 CRWLReadDataL (コマンドコード 33h)

カードの取込み、磁気データのリードを行う。

< 構文 >

```
Function CRWLReadDataL( ByRef baResponse As Byte, ByVal bMode As Byte, _  
                        ByVal bBits As Byte, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bMode	リードモードの指定 &H30 標準 (69 バイト) リード &H31 C R A - 2 2 0 0 互換リード (磁気データは 1 ~ 47 バイト可変長) &H32 カード取込み
bBits	データビット数の指定 &H30 7 b i t 磁気リード (パリティ有り) &H31 8 b i t 磁気リード (パリティ無し)
strData	磁気データを読み込む変数 (事前に 207 バイト以上の領域を確保しておくこと) 磁気データが可変長の場合、磁気データ以外の領域にはスペース (H20) をセット

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードの取込み、磁気データのリードを行う。

カードが装置内に有る場合は磁気データのリードを行う。この場合は、strData に磁気データが格納される。

カードが装置内に無い場合は、baResponse(1)に &H30、baResponse(3)に &H34 を返し、カード挿入待ちとなる。

カード挿入待ちのキャンセルは、CRWLCancel で行う。

< 対象機種 >

全機種

3.10 CRWLReadData2 (コマンドコード 33h)

カードの取込み、磁気データのリードを行う。

< 構文 >

```
Function CRWLReadData2( ByRef baResponse As Byte, ByVal bMode As Byte, _  
                        ByVal bBits As Byte, ByVal bTrack As Byte, _  
                        ByVal strData As String ) As Long  
    プログラミング上の注意点を参照してください。
```

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bMode	リードモードの指定 &H30 標準 (69 バイト) リード &H31 C R A - 2 2 0 0 互換リード (磁気データは 1 ~ 47 バイト可変長) &H32 カード取込み
bBits	データビット数の指定 &H30 7 b i t 磁気リード (パリティ有り) &H31 8 b i t 磁気リード (パリティ無し)
bTrack	トラック指定 &H30 トラック 1 のリード指定 &H31 トラック 2 のリード指定 &H32 トラック 3 のリード指定 &H33 トラック 1 , 2 のリード指定 &H34 トラック 1 , 3 のリード指定 &H35 トラック 2 , 3 のリード指定 &H36 トラック 1 , 2 , 3 のリード指定
strData	磁気データを読み込む変数 (207 バイト以上の領域を確保しておくこと) 磁気データが可変長の場合、 <i>CRWLReadVariableLengthDataL</i> 関数を使用する事

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_RESOURCE	システムの資源不足のため失敗
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードの取込み、指定されたトラックの磁気データのリードを行う。

3 トラック仕様の機種でない場合にトラック 1 以外の指定を行うと、リードエラーが発生する

カードが装置内に有る場合は磁気データのリードを行う。この場合は、strData に磁気データが格納

される。

カードが装置内に無い場合は、baResponse(1)に & H 3 0、baResponse(3)に & H 3 4 を返し、カード挿入待ちとなる。

カード挿入待ちのキャンセルは、CRWLCancel で行う。

<対象機種>

A B S - S 3 1

3.1.1 CRWLGetData (コマンドコード 35h)

リードした磁気データを取得する。

< 構文 >

Function CRWLGetData(ByRef baResponse As Byte, ByVal strData As String) As Long

プログラミング上の注意点を参照してください。

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData 磁気データを読み込む変数 (事前に 207 バイト以上の領域を確保しておくこと)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

CRWLReadData()、CRWLReadDataL()関数でリードしたデータを取得する。

ただし、CRWLWriteData()などで磁気データの書き込みを行った場合は、書き込んだデータが返される。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.12 CRWLCancel (コマンドコード 40h)

カード待ち状態をキャンセルする。

< 構文 >

```
Function CRWLCancel( ByRef baResponse As Byte ) As Long
```

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カード待ち状態をキャンセルする。

< 対象機種 >

全機種

3.13 CRWLWriteData (コマンドコード 50h)

磁気データを書き込む (7ビット)

< 構文 >

Function CRWLWriteData(ByRef baResponse As Byte, ByVal strData As String) As Long

プログラミング上の注意点を参照してください。

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData 書き込む磁気データ (69 バイト以上の領域が必要)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

69 バイト分の磁気データを書き込む。

書き込みはパリティビット付きで行うため、利用可能なデータは7ビットのデータのみ。

8ビットデータの書き込みは CRWLWriteBinaryData() で行うこと。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.14 CRWLWriteDataL (コマンドコード 53h)

磁気データを書き込む。

< 構文 >

```
Function CRWLWriteDataL( ByRef baResponse As Byte,  
                        ByVal bMode As Byte, _  
                        ByVal bBits As Byte, _  
                        ByVal strData As String ) As Long  
    プログラミング上の注意点を参照してください。
```

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bMode	ライトモードの指定 &H30 標準 (69 バイト) ライト &H31 C R A - 2 2 0 0 互換ライト (磁気データは 1 ~ 47 バイト可変長)
bBits	データビット数の指定 &H30 7 b i t 磁気ライト (パリティ有り) &H31 8 b i t 磁気ライト (パリティ無し)
strData	書き込む磁気データ (69 バイト以上の領域が必要)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

磁気データの書き込みを行う。

< 対象機種 >

全機種

3.15 CRWLWriteData2 (コマンドコード 53h)

磁気データを書き込む。

< 構文 >

```
Function CRWLWriteData2( ByRef baResponse As Byte, ByVal bMode As Byte, _  
                        ByVal bBits As Byte, ByVal bTrack As Byte, _  
                        ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bMode	ライトモードの指定 &H30 標準 (69 バイト) ライト &H31 C R A - 2 2 0 0 互換ライト (磁気データは 1 ~ 47 バイト可変長)
bBits	データビット数の指定 &H30 7 b i t 磁気ライト (パリティ有り) &H31 8 b i t 磁気ライト (パリティ無し)
bTrack	トラック指定 &H30 トラック 1 のライト指定 &H31 トラック 2 のライト指定 &H32 トラック 3 のライト指定 &H33 トラック 1 , 2 のライト指定 &H34 トラック 1 , 3 のライト指定 &H35 トラック 2 , 3 のライト指定 &H36 トラック 1 , 2 , 3 のライト指定
strData	書き込む磁気データ (書き込むトラック数 × 69 バイト以上の領域が必要)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_RESOUCE	システムの資源不足のため失敗
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

指定されたトラックに磁気データの書き込みを行う。

3トラック仕様の機種でない場合にトラック1以外の指定を行うと、ライトエラーが発生する
磁気データが可変長の場合は、*CRWLWriteVariableLengthDataL* 関数を使用すること

<対象機種>

A B S - S 3 1

3.16 CRWLWriteDataDischargeCard (コマンドコード 55h)

磁気データの書き込みとカードの排出を行う。

< 構文 >

```
Function CRWLWriteDataDischargeCard( ByRef baResponse As Byte, _  
                                     ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）

strData 書き込む磁気データ（ 69 バイト以上の領域が必要）

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

磁気データ（ 7 ビット）の書き込み後、カードの排出を行う。
カード排出後、CRWLReadStatus() でカードが取り除かれたこと（ baResponse(1) が &H30 ）を確認する必要がある。この確認を行わないと次のカードの取込み動作が行われない。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.17 CRWLWriteBinaryData (コマンドコード 5Ch)

磁気データを書き込む (8 ビット)

< 構文 >

```
Function CRWLWriteBinaryData( ByRef baResponse As Byte, _  
                               ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData 書き込む磁気データ (69 バイト以上の領域が必要)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

69 バイト分の磁気データを書き込む。
漢字コードなど 8 ビットのデータを扱う場合に使用する。

< 対象機種 >

ABS - 1231L シリーズ

3.18 CRWLPrint (コマンドコード 70h)

印字を行う。

< 構文 >

```
Function CRWLPrint( ByRef baResponse As Byte, _  
                  ByVal bLine As Byte, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

<引数>

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ~ 3 を格納する）
------------	--

bLine	印字開始行の指定	
	8行座標	&H 1 ~ &H 8
	左回転 (旧機種互換用)	&H 1 ~ &H a
	右回転 (旧機種互換用)	&H 1 ~ &H a
	左回転	&H 1 ~ &H d
	右回転	&H 1 ~ &H d
strData	印字データ (最大 2 4 8 文字)	
	印字データはヌル (0 0 h) で終了と判断します	

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

印字を行う。

印字データの制御コードとして以下のものが指定可能。

&H d	: 改行
&H 1 1	: 縦倍角指定
&H 1 4	: 文字サイズ解除
&H 1 b + &H 7 3 + y y + x x	: 文字サイズ設定
	y y &H 3 1 (縦1倍) ~ &H 3 4 (縦4倍)
	x x &H 3 1 (横1倍) ~ &H 3 4 (横4倍)
&H 1 b + &H 6 7 + 外字No .	: 外字No . で登録された外字フォント

<対象機種>

A B S - 1 2 3 1 L シリーズ

3.19 CRWLPrint20Lines (コマンドコード 73h)

印字を行う。

< 構文 >

```
Function CRWLPrint20Lines( ByRef baResponse As Byte, _  
                           ByVal bLine As Byte, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bLine	印字開始行の指定 &H 1 ~ &H 14
strData	印字データ (最大 248 文字) 印字データはヌル (00h) で終了と判断します

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

印字を行う。

CRWLPrint() とは、印字範囲が異なる。

印字データの制御コードとして以下のものが指定可能。

&H d	: 改行
&H 1 1	: 縦倍角指定
&H 1 4	: 文字サイズ解除
&H 1 b + &H 7 3 + y y + x x	: 文字サイズ設定 y y &H 3 1 (縦 1 倍) ~ &H 3 4 (縦 4 倍) x x &H 3 1 (横 1 倍) ~ &H 3 4 (横 4 倍)
&H 1 b + &H 6 7 + 外字 No .	: 外字 No . で登録された外字フォント

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

印字を行い、カードを排出する。

```
Function CRWLPrintDischargeCard( ByRef baResponse As Byte, _  
                                ByVal bLine As Byte, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ~ 3 を格納する）
------------	--

<戻り値>

< 説明 >

印字データの制御コードとして以下のものが指定可能。

<対象機種>

27

3.2.1 CRWLSetPrint (コマンドコード 78h)

印字条件の設定を行う。

<構文>

```
Function CRWLSetPrint( ByRef baResponse As Byte, _
                      ByVal bCard As Byte, ByVal bForm As Byte, _
                      ByVal bCoordinates) As Long
```

<引数>

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bCard カードの種類

&H30	白濁リライト (白	FB661 - M	消去バー用)
&H31	ロイコ式リライト (黒	431FB	消去バー用)
&H32	ロイコ式リライト (青	TRCGAACH(S)	消去バー用)
&H33	サーマル (黒	90)	
&H34	サーマル (黒	110)	
&H35	白濁リライト (白	クーダス03	OPニス有り)
&H36	白濁リライト (白	クーダス04	OPニス有り)
&H37	ロイコ式リライト (青)		
&H38	ロイコ式リライト (黒	631FB	消去バー用)

表 3-5 機種別対応カード一覧

BCard	1 2 3 1 L	A 3 1	S 3 1
&H30	×	×	
&H31	×	×	
&H32	×	×	
&H33			
&H34			
&H35		×	×
&H36			×
&H37			
&H38	×	×	

上表は参考用です。対応カードについては、各機種のソフトウェア仕様書を参照して下さい。

bForm 印字方式

&H30	オーバーライト印字 (リライトカード時のみ有効)
&H31	重ね書き印字

bCoordinates 印字座標

&H30	標準座標
&H31	左回転 (旧機種互換用)
&H32	右回転 (旧機種互換用)
&H33	左回転

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

印字条件の設定を行う。

機種により設定可能な引数が異なるので注意すること。詳細は、機種ごとのソフトウェア仕様書を参照すること。

< 対象機種 >

全機種

3.2.2 CRWLSetPrint2 (コマンドコード 78h)

印字条件の設定を行う (半角フォント指定有り)

<構文>

```
Function CRWLSetPrint2( ByRef baResponse As Byte, _  
                        ByVal bCard As Byte, ByVal bForm As Byte, _  
                        ByVal bFont As Byte, ByVal bCoordinates As Byte) As Long
```

<引数>

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bCard カードの種類

&H30	白濁リライト	(白	FB661-M	消去バー用)
&H31	ロイコ式リライト	(黒	431FB	消去バー用)
&H32	ロイコ式リライト	(青	TRCGAACH(S)	消去バー用)
&H33	サーマル	(黒	90)
&H34	サーマル	(黒	110)
&H35	白濁リライト	(白	クーダス03	OPニス有り)
&H36	白濁リライト	(白	クーダス04	OPニス有り)
&H37	ロイコ式リライト	(青)		
&H38	ロイコ式リライト	(黒	631FB	消去バー用)

機種別対応カード一覧は「3.2.1 CRWLSetPrint」の表3-5を参考にする

bForm 印字方式

&H30	オーバーライト印字 (リライトカード時のみ有効)
&H31	重ね書き印字

bFont 半角フォント

&H30	標準フォント (デフォルト)
&H31	軸太フォント

bCoordinates	印字座標
&H30	標準座標
&H31	左回転（旧機種互換用）
&H32	右回転（旧機種互換用）
&H33	左回転
&H34	右回転

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

印字条件の設定を行う。

機種により設定可能な引数が異なるので注意すること。詳細は、機種ごとのソフトウェア仕様書を参照すること。

<対象機種>

全機種

3.23 CRWLRegisterFont (コマンドコード 7Ah)

外字フォント登録を行う。

< 構文 >

Function CRWLRegisterFont(ByRef baResponse As Byte, _
ByVal bNumber As Byte, ByRef baData As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bNumber 外字登録番号 (&H1 ~ &H14)
(&H1 ~ &H2F、&H3A ~ &HFF) S31のみ

baData 登録データの先頭 (72 バイト)

< 戻り値 >

CRW_SUCCESS (0) 正常終了
CRW_ERR_HOOKUP リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE ドライバが使用可能状態で無い
CRW_ERR_COMPORT 通信ポートにエラーが発生した
CRW_ERROR その他のエラー

< 説明 >

24 × 24 ドットの外字フォントを登録する。

baData は 72 バイトの配列とし、その中にフォントデータをセットして呼び出す。

データのイメージは以下になる。

123.....24 ドット

1	データ1	データ2	データ3
2	データ4	データ5	データ6
24	データ70	データ71	データ72

各データは、2進数として表現した場合にビット
が 1 か 0 かによりドットが ON、OFF となるよう
に設定する。

例えば、左上隅に 1 ドットのみ点を打つようなフ
ォントを設定する場合は、baData(1)=&H80 とし
て残りの baData(2) ~ baData(72) はすべて &H0 とす
る。

データ1、データ2...が baData(1), baData(2) にバイト単位として設定される。

< 対象機種 >

全機種

3.24 CRWLPrintImage (コマンドコード 7Bh)

イメージ印字を行う。

< 構文 >

Function CRWLPrintImage(ByRef baResponse As Byte, _
ByVal bMode As Byte, ByVal bBuffClr As Byte, _
ByVal nXw As Long, ByVal nYw As Long, _
ByVal nXp As Long, ByVal nYp As Long, _
ByRef bytData As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bMode	印字実行指定
&H30	印字データ展開後、印字を行う
&H31	印字データ展開のみ
bBuffClr	印字バッファクリア指定
&H30	印字バッファをクリアする
&H31	印字バッファをクリアしない
nXw	X 軸イメージ幅 1 ~ 3 1 2
nYw	Y 軸イメージ幅 1 ~ 5 5 6
nXp	イメージデータの X 軸印字開始位置 0 ~ 3 1 1
nYp	イメージデータの Y 軸印字開始位置 0 ~ 5 5 5
bytData	イメージ印字データ（詳細は＜説明＞を参照）

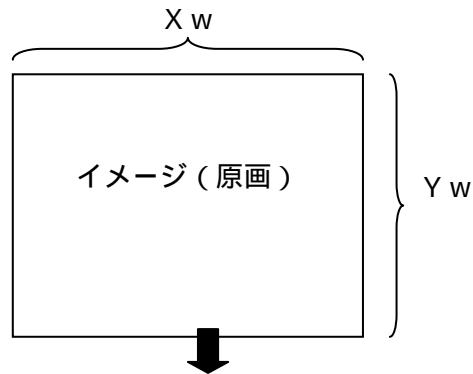
< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

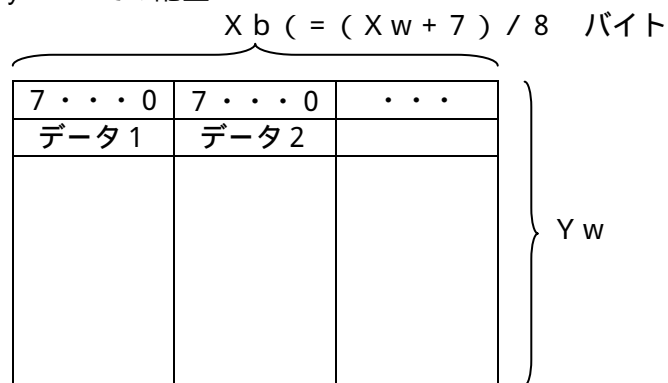
<説明>

イメージ印字を行う。

印字データは次のフォーマットで格納しておくこと。



bytdata での配置



外字登録と同様に、各イメージは、2進数として表現した場合にビットが1か0によりドットがON、OFFとなるように設定する。

例えば、左上隅に1ドットのみ点を打つようなイメージを設定する場合は、データ1=&H80として残りのデータはすべて&H0とする。

<対象機種>

ABS - S31

3.25 CRWLPrintL (コマンドコード 7Ch)

印字を行う。

< 構文 >

```
Function CRWLPrintL( ByRef baResponse As Byte, _  
                    ByVal bMode As Byte, ByVal bBuffClr As Byte, _  
                    ByVal bLine As Byte, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）		
bMode	印字実行指定		
	&H30	印字データ展開後、印字を行う	
	&H31	印字データ展開のみ	
bBuffClr	印字バッファクリア指定		
	&H30	印字バッファをクリアする	
	&H31	印字バッファをクリアしない	
bLine	印字開始行指定		
	標準座標	&H01 ～ &H14	
	左回転座標	&H01 ～ &H0d	
	右回転座標	&H01 ～ &H0d	
strData	印字データ（最大 2 4 6 文字）		
	印字データはヌル（ 0 0 h ）で終了と判断します		

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

印字を行う。

印字データの制御コードとして以下のものが指定可能。

& H d	: 改行
& H 1 1	: 縦倍角指定
& H 1 4	: 文字サイズ解除
& H 1 b + & H 7 3 + y y + x x	: 文字サイズ設定 y y & H 3 1 (縦 1 倍) ~ & H 3 4 (縦 4 倍) x x & H 3 1 (横 1 倍) ~ & H 3 4 (横 4 倍)
& H 1 b + & H 6 7 + 外字 N o .	: 外字 N o . で登録された外字フォント

< 対象機種 >

全機種

3.2.6 CRWLPrintBarcode (コマンドコード 7Eh)

バーコード印字を行う。

< 構文 >

```
Function CRWLPrintBarcode( ByRef baResponse As Byte, _  
    ByVal bMode As Byte, ByVal bBuffClr As Byte, _  
    ByVal bBarcode As Byte, ByVal bGuide As Byte, _  
    ByVal bWidth As Byte, ByVal bHeight As Byte, _  
    ByVal bXPos As Byte, ByVal bYPos As Byte, _  
    ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)	
bMode	印字実行指定	
	&H30	印字データ展開後、印字を行う
	&H31	印字データ展開のみ
bBuffClr	印字バッファクリア指定	
	&H30	印字バッファをクリアする
	&H31	印字バッファをクリアしない
bBarcodeバーコード種類指定		
	&H30	J A N 標準 (データサイズ : 13 バイト固定)
	&H31	J A N 短縮 (データサイズ : 8 バイト固定)
	&H32	N W - 7 (データサイズ : 可変)
	&H33	C O D E 3 9 (データサイズ : 可変)
	印字範囲を超えるバーコードデータは無視されます。	
bGuide	バーコード下の解説文字の有無	
	&H30	有り
	&H31	無し

bWidth

バーコード幅指定

&H30 1モジュール寸法が2ドット(0.250mm)
 &H31 1モジュール寸法が3ドット(0.375mm)

データ		30h	31h
バーコード種類			
JAN標準: 1モジュール寸法		0.250mm	0.375mm
JAN短縮: 1モジュール寸法		0.250mm	0.375mm
NW-7	ナローバー寸法	0.250mm	0.375mm
CODE39	ワイドバー寸法	0.625mm	0.750mm

bHeight

バーコード高さ指定

&H01 ~ &H30 1mm/devで1mm~48mmの指定が可能

注) 但しバーコードの上下には、上記指定の高さ以外に、上1.5mm、
 下0.5mmのマージンを自動的に取ります(解説文字有りの時は、
 下0.5mmのマージンからさらに1.5mmの解説文字を付加します)

bXPos

印字X座標

印字するバーコードの座標を指定(1mm/dev)

バーコード種類		指定できるデータの範囲
JAN標準	1モジュール=2dot	00h~0Dh
	1モジュール=3dot	00h(固定)
JAN短縮	1モジュール=2dot	00h~13h
	1モジュール=3dot	00h~09h
NW-7	1モジュール=2dot	00h~1Eh
	1モジュール=3dot	00h~1Ah
CODE39	1モジュール=2dot	00h~1Dh
	1モジュール=3dot	00h~18h

bYPos

印字Y座標

&H00 ~ &H40 印字するバーコードのY座標を指定(1mm/dev)

strData

印字データ

3~14バイト

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

以下のバーコード印字を行う。

バーコード種別	J A N 標準
	J A N 短縮
	N W - 7
	C O D E 3 9

ロイコ及び白濁リライトカードへのバーコード印字は保証出来ません。

<対象機種>

A B S - A 3 1 / A B S - S 3 1

3.27 CRWLDischargeCard (コマンドコード 80h)

カードを排出する。

< 構文 >

Function CRWLDischargeCard(ByRef baResponse As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

装置内のカードを排出する。

カード排出後、CRWLReadStatus() でカードが取り除かれたこと (baResponse(1) が &H30) を確認する必要がある。この確認を行わないと次のカードの取込み動作が行われない。

< 対象機種 >

全機種

3.2.8 CRWLDISchargeCardDispenser (コマンドコード 90h)

カードをディスペンサーに収納する。(回収機仕様のリーダライタのみ)

< 構文 >

Function CRWLDISchargeCardDispenser(ByRef baResponse As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードをディスペンサーに収納する。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ (回収機仕様のみ)

3.2.9 CRWLCleaning (コマンドコード A0h)

クリーニングを行う。

< 構文 >

Function CRWLCleaning(ByRef baResponse As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

関数呼出し後、カード挿入待ち状態となる。カードが挿入されると磁気ヘッド、サーマルヘッドのクリーニングを行い、カードを排出する。

カード排出後、CRWLReadStatus() でカードが取り除かれたこと (baResponse(1) が &H30) を確認する必要がある。この確認を行わないと次のカードの取込み動作が行われない。

< 対象機種 >

全機種

3.30 CRWLTakeInCardDispenser (コマンドコード B 0 h)

カードをディスペンサーから取込む。(発行機仕様のリーダライタのみ)

< 構文 >

Function CRWLTakeInCardDispenser(ByRef baResponse As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードをディスペンサーから取込む。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ (発行機仕様のみ)

3.3.1 CRWLTakeInCardDispenserS31 (コマンドコード B 0 h)

カードをディスペンサーから取込む。(発行機仕様のリーダライタのみ)

< 構文 >

Function CRWLTakeInCardDispenserS31(ByRef baResponse As Byte , ByVal bMode As Byte) As Long

< 引数 >

baResponse	レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)
bMode	カード取込み指定
&H31	カードディスペンサからカードを取り込む
&H32	カードディスペンサのカード有無確認のみ

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードをディスペンサーから取込む。

< 対象機種 >

C R - S 3 1 シリーズ (発行機仕様のみ)

3.3.2 CRWLControlLED (コマンドコード C 0 h)

L E D を制御する。

< 構文 >

Function CRWLControlLED(ByRef baResponse As Byte, ByVal bMode As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bMode

&h30	LED 消灯
&h31	LED 点灯
&h32	LED 低速点滅
&h33	LED 高速点滅

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

L E D を一時的に制御する。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.3.3 CRWLSetRetry (コマンドコード C1h)

リトライ数を設定する。

< 構文 >

Function CRWLSetRetry(ByRef baResponse As Byte, ByVal bCount As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

bCount リトライ回数指定
 &h31 リトライ無し
 &h32 リトライ MAX1 回
 ~
 &h39 リトライ MAX8 回

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

リード、ライト動作時のリトライ数を設定する。

< 対象機種 >

A B S - 1 2 3 1 L シリーズ

3.3.4 CRWLSetRtcDate (コマンドコード E 1 h)

時計を設定する。

< 構文 >

Function CRWLSetRtcDate(ByRef baResponse As Byte, ByVal strData As String) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData 日付・時刻を設定する配列の先頭要素
文字列で、" YYMMDDhhnnss " (年 年 月 月 日 日 時 時 分 分 秒 秒)
年は西暦の下 2 桁とする。
設定範囲は、2001/01/01 00:00:00 ~ 2099/12/31 23:59:59

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

時計を設定する。

< 対象機種 >

A B S - A 3 1 / A B S - S 3 1

3.35 CRWLGetVersion (コマンドコード F 0 h)

バージョンを取得する。

< 構文 >

Function CRWLGetVersion(ByRef baResponse As Byte, ByVal strData As String) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData バージョン情報

バイト	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
名称	アプリケーションヘッダー			アプリ種別	ソフト管理				区切り文字	アプリバージョン				区切り文字	BIOSヘッダー		
Hex	41	5D	3A	53	*	*	*	*	2D	*	*	*	*	2C			
ASCII	A	P	:	S					-					,	O	S	:

	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
	BIOS種別	ソフト管理				区切り文字	BIOSバージョン				区切り文字	予備			
	53	*	*	*	*	2D	*	*	*	*	2C	20	20	20	20
	S					-					,				

< 戻り値 >

CRW_SUCCESS (0)

正常終了

CRW_ERR_HOOKUP

リーダライタが接続されていないか、電源が OFF である

CRW_ERR_INITIALIZE

ドライバが使用可能状態で無い

CRW_ERR_COMPORT

通信ポートにエラーが発生した

CRW_ERROR

その他のエラー

< 説明 >

バージョンを取得する。strData にバージョン情報を格納する。

< 対象機種 >

ABS - A31 / ABS - S31

3.3.6 CRWLGetRtcDate (コマンドコード F 1 h)

現在時刻を取得する。

< 構文 >

Function CRWLGetRtcDate(ByRef baResponse As Byte, ByVal strData As String) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

strData 日付・時刻データを格納する配列の先頭要素
 文字列で、" YYMMDDhhnnss " (年年月月日日時時分分秒秒)
 年は西暦の下 2 桁とする。
 取得範囲は、2001/01/01 00:00:00 ~ 2100/02/28 23:59:59

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

現在の日付・時刻を取得する。

< 対象機種 >

A B S - A 3 1 / A B S - S 3 1

3.37 CRWLBattCheck (コマンドコード F 5 h)

バッテリーの状態を取得する。

< 構文 >

Function CRWLBattCheck(ByRef baResponse As Byte) As Long

< 引数 >

baResponse レスポンスを格納する配列の先頭要素 (先頭位置からレスポンス 1 ~ 3 を格納する)

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

バックアップ用電池の電圧をチェックする。

レスポンス 2 が 4 0 h で返ってきたとき、バッテリー異常と判断する。

< 対象機種 >

A B S - S 3 1

3.3.8 CRWLTSTPrint

テスト印字を行う。

<構文>

Function CRWLTSTPrint(ByVal lngWait As Long, ByVal bMode As Byte) As Long

注意 テスト印字は外字登録番号1のデータ領域を使用する為、テスト印字後は外字登録番号1の外字データは消去されます。

<引数>

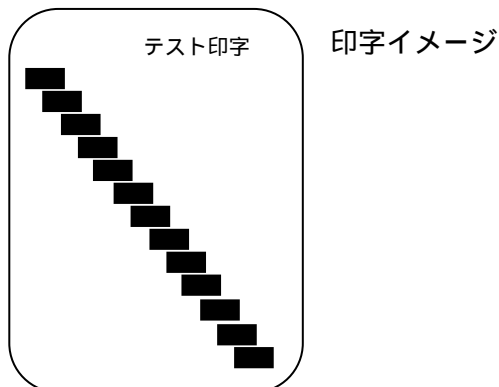
lngWait	タイムアウト時間 (秒) カード取り込み待ち時間 例) 10 秒 (10)
bMode	テスト印字方法指定
	&h30 印字データ以外の部分に以前書かれていた字を消去する
	&h31 印字データ以外の部分に以前書かれていた字を消去しない

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERR_TIMEOUT	カード取り込みタイムアウト 及び タイムアウト時間指定範囲外
CRW_ERROR	その他のエラー

<説明>

カード入力待ち状態からカードの取り込み、テスト印字、カードの排出を行う。
印字結果を目視することで、印字ヘッドの故障を発見できる。



<対象機種>
全機種

3.3.9 CRWLReadVariableLengthDataL

カードの取込み、可変長磁気データのリードを行う。

< 構文 >

```
Function CRWLReadVariableLengthDataL( ByRef baResponse As Byte, _  
                                     ByVal bMode As Byte, ByVal bBits As Byte, _  
                                     ByRef lngLen As Long, ByVal strData As String ) As Long
```

< 引数 >

プログラミング上の注意点を参照してください。

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ~ 3 を格納する）
bMode	リードモードの指定 &H30 標準（69 バイト）リード &H32 カード取込み &H33 可変長（最大 69 バイト）リード
bBits	データビット数の指定 &H30 7 b i t 磁気リード（パリティ有り） &H31 8 b i t 磁気リード（パリティ無し）
lngLen	読み込んだデータ数（バイト数）
strData	磁気データを読み込む変数（事前に 207 バイト以上の領域を確保しておくこと）

< 戻り値 >

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダーライターが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

< 説明 >

カードの取込み、磁気データのリードを行う。

カードが装置内に有る場合は磁気データのリードを行う。この場合は、strData に磁気データが格納される。

カードが装置内に無い場合は、baResponse(1)に &H30、baResponse(3)に &H34 を返し、カード挿入待ちとなる。**この後、通常 CRWLReadDataL() 関数実行後は CRWLReadStatus() 関数を呼び出しカードの状態を監視するが、本関数実行後は CRWLGetVariableLengthData() 関数でカードの状態監視、データの取得を行うこと。**

カード挿入待ちのキャンセルは、CRWLCancelI で行う。

< 対象機種 >

A B S - A 3 1 / A B S - S 3 1

3.40 CRWLGetVariableLengthData

リードした可変長磁気データを取得する。

<構文>

```
Function CRWLGetVariableLengthData( ByRef baResponse As Byte, _  
                                     ByRef lngLen As Long, ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

<引数>

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）
lngLen	読み込んだデータ数（バイト数）
strData	磁気データを読み込む変数（事前に 207 バイト以上の領域を確保しておくこと）

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダーライターが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERR_GET	CRWLReadVariableLengthDataL() 実行直後以外のときに呼ばれた
CRW_ERROR	その他のエラー

<説明>

CRWLReadVariableLengthDataL() 関数でリードした可変長磁気データを取得する。
ただし、CRWLReadVariableLengthDataL() 関数実行直後のみ動作する。

<対象機種>

ABS - A31 / ABS - S31

3.4.1 CRWLWriteVariableLengthDataL

可変長磁気データを書き込む。

<構文>

```
Function CRWLWriteVariableLengthDataL( ByRef baResponse As Byte,  
                                       ByVal bMode As Byte, _  
                                       ByVal bBits As Byte, _  
                                       ByVal lngLen As Long, ByVal strData As String ) As Long  
                                     プログラミング上の注意点を参照してください。
```

<引数>

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ~ 3 を格納する）
bMode	ライトモードの指定 &H30 標準（69 バイト）ライト &H33 可変長（最大 69 バイト）ライト
bBits	データビット数の指定 &H30 7 b i t 磁気ライト（パリティ有り） &H31 8 b i t 磁気ライト（パリティ無し）
lngLen	書き込むデータ数（バイト数）
strData	書き込む磁気データ

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

可変長磁気データの書き込みを行う。

<対象機種>

A B S - A 3 1 / A B S - S 3 1

3.4.2 CRWLEnq

レスポンスの再送要求を行う。

<構文>

```
Function CRWLEnq( ByRef baResponse As Byte,  
                  ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

<引数>

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）
strData	磁気データを読み込む変数（事前に 2 0 7 バイト以上の領域を確保しておくこと）

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダーライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

カード挿入待ちの時など、レスポンスを要求したい時に使用します。

<対象機種>

全機種

3.4.3 CRWLEnq2

レスポンスの再送要求を行う。

<構文>

```
Function CRWLEnq2( ByRef baResponse As Byte,  
                  ByVal strData As String ) As Long
```

プログラミング上の注意点を参照してください。

<引数>

baResponse	レスポンスを格納する配列の先頭要素（先頭位置からレスポンス 1 ～ 3 を格納する）
strData	磁気データを読み込む変数（事前に 2 0 7 バイト以上の領域を確保しておくこと）

<戻り値>

CRW_SUCCESS (0)	正常終了
CRW_ERR_HOOKUP	リーダーライタが接続されていないか、電源が OFF である
CRW_ERR_INITIALIZE	ドライバが使用可能状態で無い
CRW_ERR_COMPORT	通信ポートにエラーが発生した
CRW_ERROR	その他のエラー

<説明>

この関数は、B u s y レスポンスも返します。

- ・ CRWLEnq は、アプリケーションに Busy レスポンスを返しません。
例えば、アプリケーションがカード取込み待ちのとき、他の処理を何も行わなければ CRWLEnq を使用して下さい。
- ・ CRWLEnq2 は、アプリケーションにカード端末が返した全てのレスポンスを返します。
例えば、アプリケーションがカード取込み待ちのとき、他の処理を行う場合は CRWLEnq2 を使用して下さい。

クリーニング中を検出するために使用します。

クリーニング処理は、カード挿入待ちの時にキャンセルすることができます。

コマンドボタンを使用してキャンセルを行う場合、CRWLEnq を使用するとクリーニング中 (Busy レスポンス) はアプリケーションに戻ってきませんので、コマンドボタンを非表示にするタイミングが失われます。従って、アプリケーションではキャンセルができない仕様にする必要があります。

CRWLEnq2 を使用すると、クリーニング中 (Busy レスポンス) はアプリケーションに戻って来ますので、コマンドボタンを非表示にするタイミングが得られます。

<対象機種>

全機種

4 プログラミング

ここでは、本ドライバを利用してアプリケーションプログラムを作成する場合に参考になる情報を提供します。

4.1 プログラミング上の注意点

本ドライバを利用してアプリケーションを作成する場合に、以下の点に留意して作成してください。

データ領域の確保

引数の領域確保はアプリケーション側で行ってください。

各関数の baResponse や CRWLReadDataL 関数の strData などの引数は必要な領域をアプリケーション側で確保します。

CRWLWriteData 関数の strData などでは、以下のようなコーディングに注意してください

例 1) 領域を確保していない

```
Dim strData as String
strData = "1999/01/01 Test Card"
```

例 2) 207 バイト分の領域を確保

```
Dim strData as String
strData = Space(207)
Lset strData = "1999/01/01 Test Card"
```

引数の範囲

このドキュメントでは引数の説明箇所にそれぞれの引数の取りうる値を記述してありますが、その大部分はアプリケーションからの値をそのままリーダーライタに渡しています。したがって、範囲外の値が渡されたときの動作はリーダーライタの動作に依存します。誤って範囲外の値が渡されないように十分注意してください。

戻り値とレスポンスステータスの両方を監視する

各関数の戻り値はドライバの動作上発生するエラー情報のみを戻します。リーダーライタで発生したエラー（書き込みエラーなど）は、引数 baResponse にレスポンスとして情報が戻りますので、こちらの値を監視してください。

引数のデータ型

String 型の場合、バイナリデータを文字列変数に格納すると、変換により日本語文字列としての認識できないコードは削除され、正しく動作しなくなりますので、バイト型 (Byte) 変数及びバイト型配列を使用して下さい。

String に代入できない 16 進数

```
"128 (&h80)" ~ "159 (&h9F)"
"224 (&hE0)" ~ "252 (&hFC)"
代入した場合 "&h00" になってしまいます。
```

例) ABS1231Def.bas

```
Public Declare Function CRWLWriteData Lib "ABS1231.DLL" (ByRef baResponse As Byte, ByVal bMode As Byte, _
    ByVal bBits As Byte, ByVal strData As String) As Long
    (変更前)

Public Declare Function CRWLWriteData Lib "ABS1231.DLL" (ByRef baResponse As Byte, ByVal bMode As Byte, _
    ByVal bBits As Byte, ByRef strData As Byte) As Long
    (変更後)
```

上記のように変更することで使用可能になります。

なお、Windows2000以降、WindowsAPIの文字列扱いが全てUnicodeとなったため、VisualBasicランタイム環境内でも文字列の扱いがUnicodeとなっています。開発環境によっては、文字列をShiftJISで扱うものがあり、DLL側がパラメータとしてUnicodeを期待しているために不整合が生じることがあります(リードデータを取得できない等の問題が発生する可能性があります)。

このような問題が起こった場合には、上記のようにバイト型に変更して扱うか、またはアプリケーション側で、shiftJIS<->Unicodeの変換ルーチンを準備し、この変換ルーチンを通して、アプリケーションで定義したShiftJIS文字列をUnicode文字列に変更しDLLにパラメータとして渡す、またパラメータ読み出し時には、DLLから受けた文字列をshiftJISに逆変換して読み出す等の対応をする必要があります。

コマンドの戻り値による要因・対策

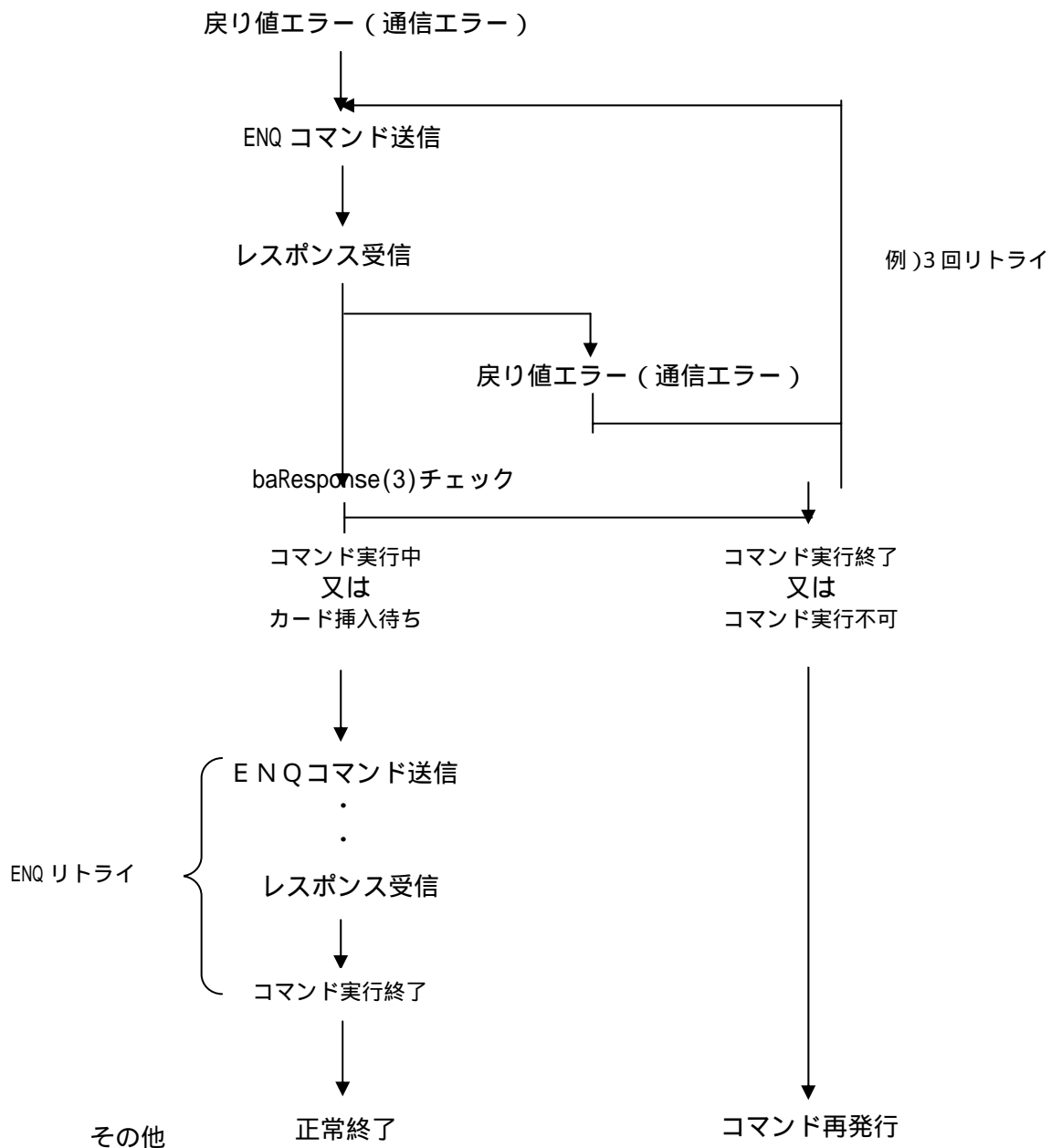
表 コマンド戻り値エラー詳細一覧

番号	エラー	要因	対策
1	CRW_ERR_HOOKUP(-20)	コマンド送信時 CTS ON 監視タイムアウト	・PC - 端末間のケーブルがクロスケーブルが確認する。
2	CRW_ERR_INITIALIZE(-30)	通信ポートの初期化未完了	・CRWInitializeにより通信ポートの初期化を行ってください。
3	CRW_ERR_COMPORT(-40)	送受信タイムアウト 受信データエラー 受信データ長エラー	・COMポート、通信速度、パリティビット、ストップビット、タイムアウト時間を確認してください。 ・電源ノイズや静電気の影響がないか確認してください。 ・ケーブル抜けがないか確認してください。
4	CRW_ERROR(-999)	レスポンスのデータ部レンジエラー 通信リトライ回数範囲外	・COMポート、通信速度、パリティビット、ストップビット、タイムアウト時間を確認してください。 ・電源ノイズや静電気の影響がないか確認してください。 通信リトライ回数に範囲内の値を設定しているか確認してください。

戻り値エラー発生時の処理例

戻り値エラーは主に通信異常が起った時に発生します。

発生時の処理例を以下に示します。



本ドキュメントのほかに、三和ニューテック株式会社発行の製品（納入）仕様書を参考にしてください。アプリケーション開発時の留意点などの記述があります。

4.2 コーディングサンプル

標準的なアプリケーションのコーディングサンプルを示します。

なお、すべてのサンプルはVisual Basicのコードとなります。また、Visual Basicのプロジェクトに標準モジュールのABS1231Def.Basが追加されているものとします。

なお、初期化处理、終了処理は共通部分として以下に示します。

```
' -----  
' フォームロード  
Private Sub Form_Load()  
    Dim baResponse(3) As Byte  
    Dim lngRet As Long  
  
    ' 初期化处理  
    lngRet = CRWInitialize(baResponse(1), 1, 9600, 1, 1, 10000)  
    If lngRet <> 0 Or baResponse(2) <> &H30 Then  
        ' エラー処理を記述する  
    End  
    End If  
End Sub  
  
' -----  
' フォームアンロード  
Private Sub Form_Unload(Cancel As Integer)  
    ' 終了処理  
    CRWTerminate  
End Sub
```

4.2.1 カードの新規発行例

‘ カードの新規発行

```
Private Sub Command1_Click()  
    Dim baResponse(3) As Byte  
    Dim lngRet As Long  
    Dim lngLoop As Long  
    Dim strData As String  
  
    ' 領域確保  
    strData = Space(207)  
    ' カード挿入要求  
    lngRet = CRWLReadDataL(baResponse(1), &H32, &H30, strData)  
    ' 新規の場合は&H32を指定する  
    If lngRet <> 0 Or baResponse(2) <> &H30 Then  
        ' エラー処理を記述する  
        Exit Sub  
    End If  
  
    ' カード挿入のチェック  
    For lngLoop = 1 To 10000 ' カードが挿入されるのを待つ  
        ' 100ms 以上間隔を開けてコマンド送信を行う  
        ' レスポンスの再送要求  
        lngRet = CRWLEnq(baResponse(1), strData)  
        If lngRet <> 0 Or baResponse(2) <> &H30 Then  
            ' エラー処理を記述する  
            Exit Sub  
        End If  
  
        ' ステータスチェック  
        If baResponse(1) = &H31 And baResponse(3) = &H30 Then  
            ' カードが挿入された  
            Exit For  
        End If  
    Next  
  
    If baResponse(1) <> &H31 Then  
        ' カードが挿入されなかった  
        Exit Sub  
    End If  
    ' 磁気データの設定を行う  
    ' 磁気面書き込み  
    ' このとき strData が 69 バイト以上あること  
    lngRet = CRWLWriteDataL(baResponse(1), &H30, &H30, strData)  
  
    ' 書き込み失敗  
    If lngRet <> 0 Or baResponse(2) <> &H30 Then  
        ' エラー処理を記述する  
        Exit Sub  
    End If
```

```

'印字データ設定
strData = "A B S 1 2 3 1"
IngRet = CRWLPrintL(baResponse(1), &H30, &H30, &H1, strData)

'印字面書き込み失敗
If IngRet <> 0 Or baResponse(2) <> &H30 Then
    ' エラー処理を記述する
    Exit Sub
End If

'カード排出
IngRet = CRWLDischargeCard(baResponse(1))      'カード排出
If IngRet <> 0 Or baResponse(2) <> &H30 Then
    ' エラー処理を記述する
    Exit Sub
End If
Do
    ' 100ms 以上間隔を開けてコマンド送信を行う
    IngRet = CRWLEnq(baResponse(1),strData)
    If IngRet <> 0 Or baResponse(2) <> &H30 Then
        ' エラー処理を記述する
        Exit Sub
    End If

    If baResponse(1) = &H30 Then '装置内カード無し
        Exit Do
    End If
Loop

End Sub

```

4.2.2 カードの更新例

‘ カードのデータ更新

```
Private Sub Command1_Click()  
    Dim baResponse(3) As Byte  
    Dim lngRet As Long  
    Dim lngLoop As Long  
    Dim strData As String  
  
    ' 領域確保  
    strData = Space(207)  
    ' カード挿入要求  
    lngRet = CRWLReadDataL(baResponse(1), &H30, &H30, strData)  
    If lngRet <> 0 Or baResponse(2) <> &H30 Then  
        ' エラー処理を記述する  
        Exit Sub  
    End If  
  
    ' カード挿入のチェック  
    For lngLoop = 1 To 10000 ' カードが挿入されるのを待つ  
        ' 領域確保  
        strData = Space(207)  
        ' 100ms 以上間隔を開けてコマンド送信を行う  
        ' レスポンスの再送要求  
        lngRet = CRWLEnq(baResponse(1), strData)  
        If lngRet <> 0 Or baResponse(2) <> &H30 Then  
            ' エラー処理を記述する  
            Exit Sub  
        End If  
  
        ' ステータスチェック  
        If baResponse(1) = &H31 And baResponse(3) = &H30 Then  
            ' カードが挿入された  
            Exit For  
        End If  
    Next  
  
    If baResponse(1) <> &H31 Then  
        ' カードが挿入されなかった  
        Exit Sub  
    End If  
  
    ' レスポンスの再送要求により取得した磁気データ(strData)を参照し、正しいカードを  
    ' リードしたか検査する  
    ' レスポンスの再送要求により取得した磁気データ(strData)の更新を行う  
  
    ' 磁気面書き込み このとき strData が 69 バイト以上あること  
    lngRet = CRWLWriteDataL(baResponse(1), &H30, &H30, strData)  
  
    ' 書き込み失敗  
    If lngRet <> 0 Or baResponse(2) <> &H30 Then  
        ' エラー処理を記述する
```

```

Exit Sub
End If

'印字データ設定
strData = " A B S 1 2 3 1 "
IngRet = CRWLPrintL(baResponse(1), &H30, &H30, &H1, strData)

'印字面書き込み失敗
If IngRet <> 0 Or baResponse(2) <> &H30 Then
    ' エラー処理を記述する
    Exit Sub
End If

'カード排出
IngRet = CRWLDischargeCard(baResponse(1))      'カード排出
If IngRet <> 0 Or baResponse(2) <> &H30 Then
    ' エラー処理を記述する
    Exit Sub
End If
Do
    ' 100ms 以上間隔を開けてコマンド送信を行う
    IngRet = CRWLEnq(baResponse(1), strData)
    If IngRet <> 0 Or baResponse(2) <> &H30 Then
        ' エラー処理を記述する
        Exit Sub
    End If

    If baResponse(1) = &H30 Then '装置内カード無し
        Exit Do
    End If
Loop
End Sub

```