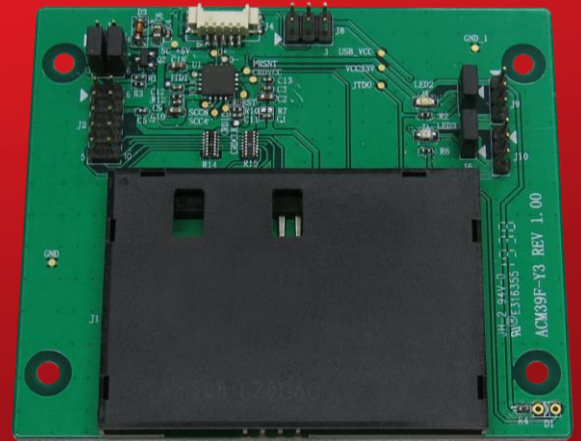




**Advanced Card Systems Ltd.**  
Card & Reader Technologies

# ACM39U-Y (CCID)カードリー ダーモジュール



リファレンスマニュアル V1.00



## 改定履歴

リリース日付	改訂説明	バージョン
2020/04/15	<ul style="list-style-type: none"><li>初回発布</li></ul>	1.00



## 目次

1.0.	紹介 .....	5
1.1.	参照ファイル .....	5
1.2.	シンボルと略語 .....	5
2.0.	特性 .....	6
3.0.	サポートしているスマートカード .....	7
3.1.	MCU カード .....	7
3.2.	メモリカード .....	7
4.0.	スマートカードインターフェース .....	8
4.1.	スマートカード電源 VCC (C1) .....	8
4.2.	プログラミング電圧 VPP (C6) .....	8
4.3.	カードタイプのセクション .....	8
4.4.	マイクロコントローラベースカードのためのインターフェース .....	8
4.5.	カード引き裂き保護 .....	8
5.0.	電源 .....	9
5.1.	LED .....	9
5.1.1.	状態 .....	9
5.1.2.	LED 動作 .....	10
6.0.	ハードウェアの接続 .....	11
6.1.	J2: 外部接続式カードコネクタ .....	12
6.2.	J3 : カード検出選択肢 (ノーマルショート/ノーマルオープン) .....	13
6.3.	J4: USB ケーブルコネクタ .....	14
6.4.	J8: USB ピンアウトコネクタ .....	14
6.5.	J9 : カードの接続信号コネクタ .....	15
6.6.	J10 : USB 電源信号コネクタ .....	15
7.0.	USB インターフェース .....	16
7.1.	通信パラメーター .....	16
7.2.	エンドポイント .....	16
8.0.	通信プロトコル .....	17
9.0.	メモリカードタイプの選択 .....	19
10.0.	コマンド .....	20
10.1.	CCID コマンドパイプ、バルクアウトメッセージ .....	20
10.1.1.	PC_to_RDR_lccPowerOn .....	20
10.1.2.	PC_to_RDR_lccPowerOff .....	20
10.1.3.	PC_to_RDR_GetSlotStatus .....	21
10.1.4.	PC_to_RDR_XfrBlock .....	21
10.1.5.	PC_to_RDR_GetParameters .....	21
10.1.6.	PC_to_RDR_ResetParameters .....	22
10.1.7.	PC_to_RDR_SetParameters .....	22
10.2.	CCID Bulk-IN メッセージ .....	25
10.2.1.	RDR_to_PC_DataBlock .....	25
10.2.2.	RDR_to_PC_SlotStatus .....	26
10.2.3.	RDR_to_PC_Parameters .....	27
10.3.	メモリカードのコマンドセット .....	28



10.3.1.	メモ리카ード – 1、2、4、8 および 16 kilobit I2C カード.....	28
10.3.2.	メモ리카ード – 32、64、128、256、512 および 1024 kilobit I2C カード.....	30
10.3.3.	メモ리카ード – SLE 4418/SLE 4428/SLE 5518/SLE 5528.....	32
10.3.4.	メモ리카ード – SLE 4432/SLE 4442/SLE 5532/SLE 5542.....	37
10.4.	PC_to_RDR_XfrBlock を介して、他のコマンドを実行する .....	41
10.4.1.	GET_READER_INFORMATION .....	41
10.5.	PC-to_RDR_Escape を介して、他のコマンドを実行する.....	42
10.5.1.	GET_READER_INFORMATION .....	42
Appendix A.	応答エラーコード.....	43

#### 図示一覧表

図示 1	: ACM39U-Y3 LED 配置.....	9
図示 2	: ACM39U-Y3 ハードウェアの接続.....	11
図示 3	: 外部接続式カードコネクタ.....	12
図示 4	: ACM39U-Y3 背面の抵抗ジャンパーを選択する.....	13
図示 5	: USB ケーブルコネクタ.....	14
図示 6	: USB ピンアウトコネクタ.....	14
図示 7	: カードの接続信号コネクタ .....	15
図示 8	: USB 電源信号コネクタ.....	15

#### チャート一覧表

表 1	: シンボルと略語 .....	5
表 2	: USB インターフェース配線.....	16
表 3	: 応答エラーコード .....	43

## 1.0. 紹介

ACM39U-Y モジュールカードリーダーは PC（個人のパソコン）とスマートカードの間の通信インターフェースです。異なるタイプのスマートカードは異なるコマンドと通信プロトコルを採用しているため、ほとんどの場合、スマートカードとコンピュータの間の直接通信を防止します。ACM39U-Y スマートカードリーダーと同じコアを共有 ACM39 モジュールカードリーダーは、コンピュータから多種多様なスマートカードへの統一されたインタフェースを確立します。スマートカードのたくさんの特性を持っているため、コンピュータ・ソフトウェア・プログラマがスマートカードの操作を詳しく了解する必要をなくなります。多くの場合、これらの操作の技術的詳細はスマートカードシステムの実装と関係がありません。

## 1.1. 参照ファイル

下記のファイルは [www.usb.org](http://www.usb.org) でダウンロードできます。

- 《ユニバーサル・シリアル・バス仕様 2.0》（即ち USB 仕様）、2000 年 4 月 27 日
- 《ユニバーサル・シリアル・バス共通クラス仕様 1.0》、1997 年 12 月 16 日
- 《ユニバーサル・シリアル・バス・デバイス・クラス：集積回路（S）カード・インターフェース・デバイス用のスマートカード CCID 仕様 1.1》、2005 年 4 月 22 日

下記のファイルは [www.usb.org](http://www.usb.org) でオーダーできます。

- 《ISO/IEC 7816-1：識別カード — 接点付きの集積回路（S）カード — パート 1：物理特性》
- 《ISO/IEC 7816-2：識別カード — 接点付きの集積回路（S）カード — パート 2：接点のサイズと場所》
- 《ISO/IEC 7816-3：識別カード — 接点付きの集積回路（S）カード — パート 3：電子信号および伝送プロトコル》

## 1.2. シンボルと略語

略語	説明
ATR	リセット応答（Answer-To-Reset）
CCID	チップ/スマートカードインターフェースデバイス（Chip/Smart Card Interface Device）
ICC	集積回路カード（Integrated Circuit Cards）
IFSC	プロトコル T=1 の集積回路カードの情報フィールドのサイズ（Information Field Sized for ICC for protocol T=1）
IFSD	プロトコル T=1 のチップ/スマートカードインターフェースデバイスの情報フィールドのサイズ（Information Field Sized for ICC for protocol T=1）
NAD	ノードの場所（Node Address）
PPS	プロトコルおよびパラメータセレクション（Protocol and Parameters Selection）
RFU	将来に使用のために保留（Reserved for Future Use） <sup>1</sup>
TPDU	転送プロトコルデータユニット Application Protocol Data Unit
USB	ユニバーサル・シリアル・バス（Universal Serial Bus）

**表1：シンボルと略語**

<sup>1</sup> 特別の説明がなければ、0 に設置しなければなりません。



## 2.0. 特性

- USB フルスピード・インターフェース
- プラグアンドプレイ——CCID 準拠、高い柔軟性を持っている
- スマートカードリーダー：
  - 接触式インターフェース
    - ISO 7816 クラス A、B、C の（5V、3V および 1.8V）カードサポート
    - 共通の特権カード（CAC）サポート
    - SIPRNET カードサポート
    - J-LIS カードサポート
    - T = 0 または T = 1 プロトコルのマイクロプロセッサ・カードをサポート
    - 様々なメモリカードサポート
    - プロトコルとパラメータの選択サポート（PPS）
    - 短絡保護保有
- アプリケーション プログラミング インターフェース
  - PC/SC サポート
  - CT- API サポート（PC / SC の上のラッパー経由で）
- Android™ 3.1 と以降のバージョンサポートしている2
- 以下の規格に準拠：
  - EN 60950/IEC 60950
  - ISO 7816
  - EMV™ Level 1 (接触式)
  - PC/SC
  - CCID
  - CE
  - FCC
  - RoHS
  - REACH
  - Microsoft® WHQL

---

2 PC/SC と CCID サポートに適用しません

## 3.0. サポートしているスマートカード

### 3.1. MCU カード

ACM39U-Y モジュールカードリーダーが PC/SC 基準を準じて、ISO 7816 A タイプ、B タイプと C タイプ（5V、3V と 1.8V）スマートカードをサポートします。全ての  $T = 0$  または  $T = 1$  プロトコルに準拠している MCU カードをサポートしている。

カードの ATR が専用の操作モードを指定すれば（ $T A 2$  が存在している； $T A 2$  のビット 5 は 0 でなければなりません）、しかし ACM39U-Y (CCID)がこのモードをサポートできない場合、ACM39U-Y (CCID) はカードをリセットして、交渉モードに設置します。交渉モードを設置できないと、ACR3901U-S1 がこのカードを拒否します。

カードの ATR が交渉のモード（ $T A 2$  が存在指定ない）および通信パラメータ（デフォルトパラメータじゃなくて）を指定すれば、ACM39U-Y (CCID)がその通信パラメータを使用して、PPS を実行します。ACR32 が PPS を拒否したら、デフォルトパラメータを使用する（ $F=372$ ,  $D=1$ ）。

上記のパラメータの意味について、ISO 7816-3 仕様を参照してください。

### 3.2. メモリカード

ACM39U-Y が様々のメモリカードをサポートしている、例：

- I2C バスプロトコルに準拠し、一回で 128 バイト/ページを書くことができるメモリカード（フリーメモリカード）、以下を含めて：
  - Atmel®：AT24C01/02/04/08/16/32/64/128/256/512/1024
  - SGS-Thomson：ST14C02C、ST14C04C
  - Gemplus：GFM1K、GFM2K、GFM4K、GFM8K
- 書き込み保護機能付インテリジェント 256 バイトの EEPROM カード、以下を含めて：
  - Infineon®：SLE4418、SLE4428、SLE5518 および SLE5528
- インテリジェント 256 バイトの EEPROM、書き込みのカードプロテクト機能付カード、以下を含めて：
  - Infineon®：SLE4432、SLE4442、SLE5532 および SLE5542

## 4.0. スマートカードインターフェース

ACM39U-Y と挿入されたカードの間のインターフェースが ISO 7816-3 仕様プロトコルに準拠して、ACM39U-Y の実用的な機能性を高めるために一定の制限や機能拡張をします。

### 4.1. スマートカード電源 VCC (C1)

挿入されたカードの消費電流は 50mA よりも高くしてはならない。

### 4.2. プログラミング電圧 VPP (C6)

ISO7816-3 仕様によると、スマートカードコンタクト C6 (VPP) がスマートカードにプログラミング電圧を供給する。市場内のすべてのスマートカードが EEPROM ベースであり、外部プログラミング電圧の供給の必要がないです。

ACM39U-Y のコンタクト C6 (VPP) が通常の制御信号として実装されました。このコンタクトの電気仕様は信号 RST (コンタクト C2) の電気仕様と同じです。

### 4.3. カードタイプのセレクション

制御 PC は、挿入されたカードをアクティブする前に、ACM39U-Y に適切なコマンドを送信してカードタイプを選択する必要があります。メモリカードと MCU ベースカードも含めている。

MCU ベースカードに対して、リーダーが T=0 または T=1 中から望ましいプロトコルを選びます。しかしながら、挿入されたカードは両方のプロトコルタイプをサポートできる場合は、リーダーが PPS を通じて、このセレクションを受け入れられて、実行します。MCU カードは 1 つだけのプロトコル (T=0 または T=1) をサポートする時に、アプリケーションがどのプロトコルを選ぶことと関係なくて、リーダーは自動的にこのプロトコルタイプを選択します。

### 4.4. マイクロコントローラベースカードのためのインターフェース

マイクロコントローラカードは C1 (VCC)、C2 (RST)、C3 (CLK)、C5 (GND) および C7 (I/O) これらのコンタクトだけ使用します。4 MHz の周波数が CLK 信号 (C3) に適用します。

### 4.5. カード引き裂き保護

電気入れる状態で、急に引き出されたカードを保護するために、ACM39U-Y がメカニズムを提供しています。カードが取り外されている時、ACM39U-Y とカード間の信号線への電力供給がすぐに非アクティブ化されます。原則として、電氣的な損傷を回避するために、パワーダウンしてから、カードをリーダーから除去されるべきである。

注意：ACM39U-Y は、決して挿入されたカードへの電源供給に切り替わりません。ホストから適切なコマンドが読者に送られ、明示的にこの操作を行う必要があります。



## 5.0. 電源

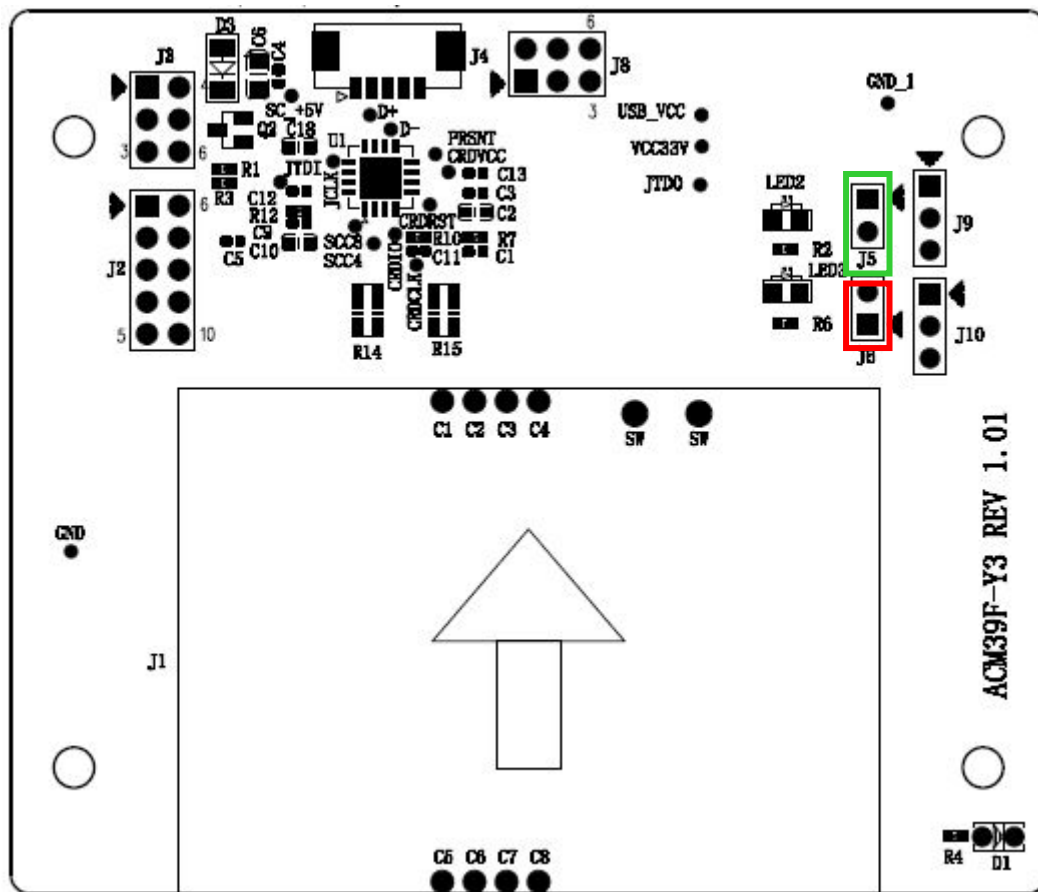
ACM39U-Y5V、100 mA の安定化電源の電圧が必要で、PC で電源を供給します（各タイプのリーダーと共に供給されたケーブルを介して）。詳細については、ACM39U-Y ユーザーマニュアルの LED 使用に関する説明を参照してください。

### 5.1. LED

#### 5.1.1. 状態

ACM39U-Y3 には、動作状態を示す 2 つの LED が装備されています。

- **赤**は電源インジケータです
- **緑**はスマートカードの動作インジケータです



図示1 : ACM39U-Y3 LED 配置



### 5.1.2. LED 動作

次の表に、カードポーリング中の LED の動作を示します。

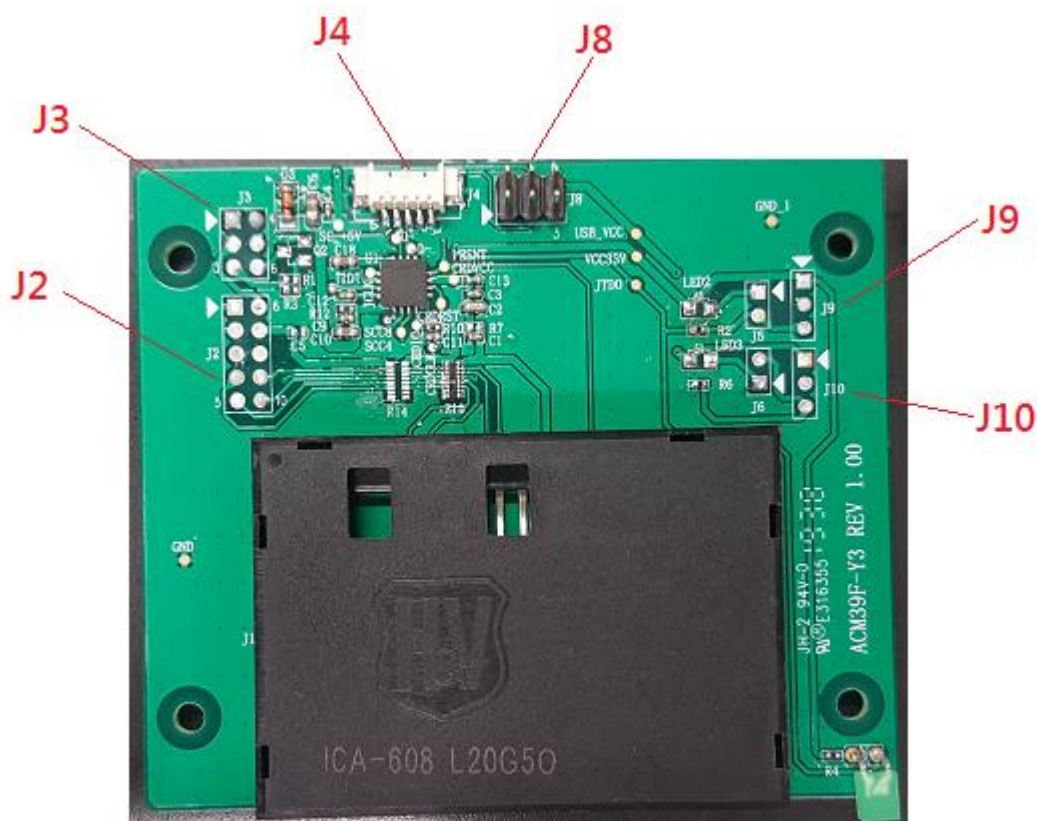
LED 色	操作	状態
赤	点灯	電源
緑	ゆっくり点滅 <sup>3</sup>	カード挿入していない/ カードパワーダウン
	点灯	カードパワーアップ
	速く点滅する	カード動作中

---

<sup>3</sup> LED が 2 秒ごとに 200ms オンする)

## 6.0. ハードウェアの接続

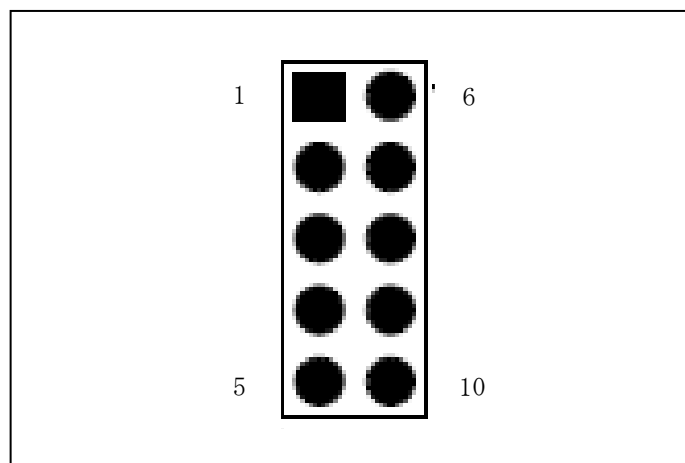
詳細については、ACM39U-Y ユーザーマニュアルのコネクタに関する説明を参照してください。



図示2 : ACM39U-Y3 ハードウェアの接続

ジャンパー	コネクタ
J2	外部接続式カードコネクタ
J3	カード検出選択肢（ノーマルショート/ ノーマルオープン）
J4	USB ケーブルコネクタ
J8	USB ピンアウトコネクタ
J9	カードの接続信号コネクタ
J10	USB 電源信号コネクタ

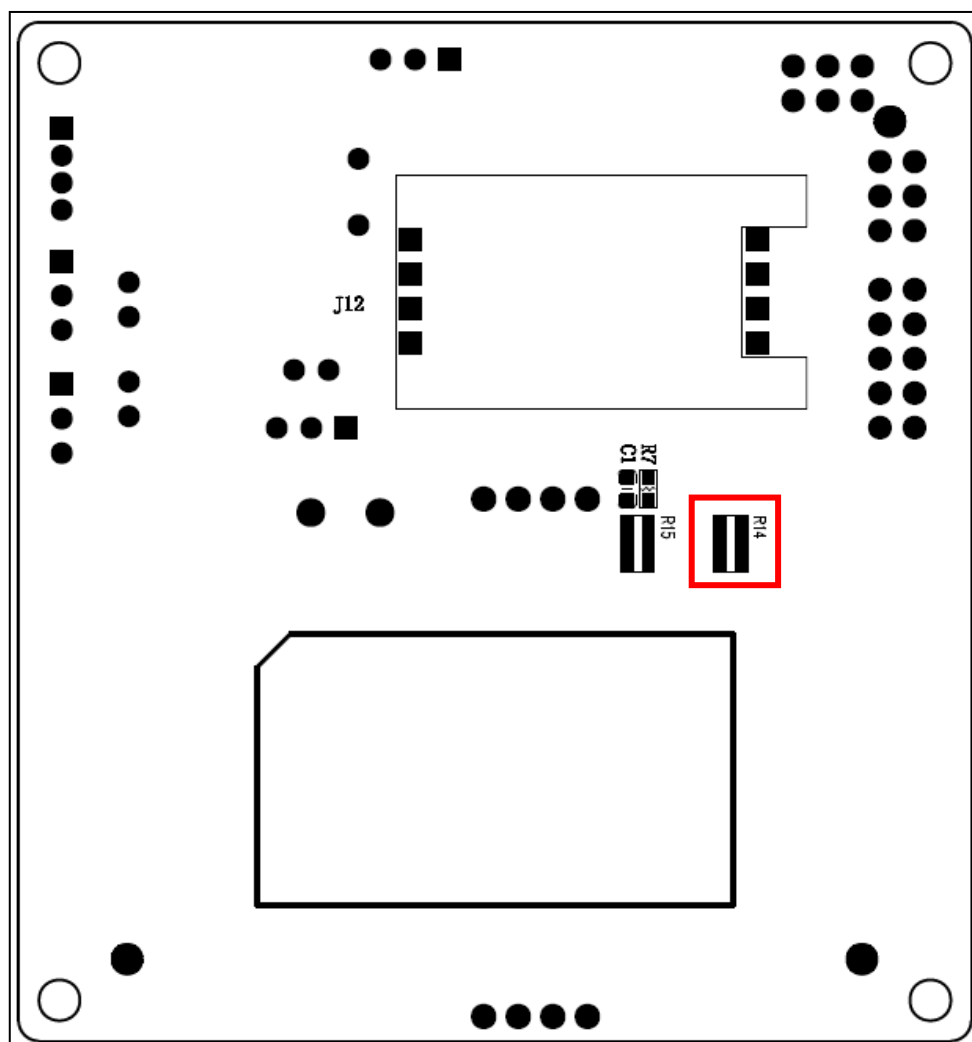
## 6.1. J2:外部接続式カードコネクタ



図示3 : 外部接続式カードコネクタ

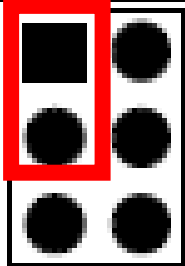
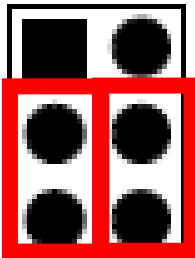
コネクタ	説明
1、2	C5:GND
3	C6:NC
4	C7:カード I/O (データ入力/輸出)
5	C8:カードのコネクタナンバー
6	カードの検出ピン
7	C1:カード VCC
8	C2:カードのリセットピン
9	C3 : カードのクロック信号
10	C4:カードのコネクタナンバー

外部カードホルダーを使用する場合、抵抗ジャンパーを変更する必要があります。抵抗アレイは、R15ではなく（R14）にはんだ付けする必要があります。下記の図示のように。

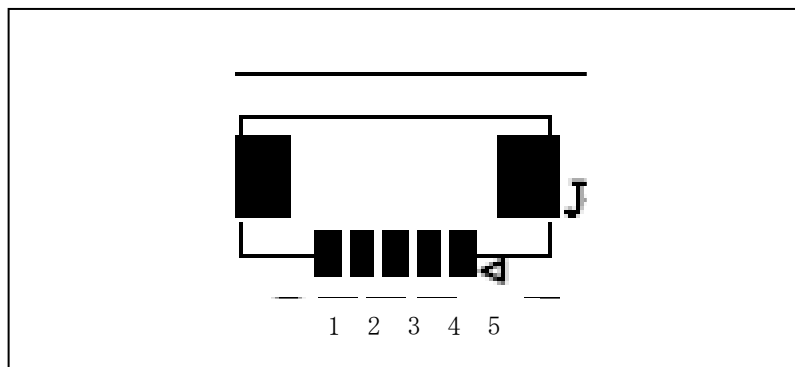


図示4 : ACM39U-Y3 背面の抵抗ジャンパーを選択する

## 6.2. J3 : カード検出選択肢（ノーマルショート/ノーマルオープン）

配置	結線図	説明
カードの検出回路がノーマルオープン		カードスロットの検出オプションがノーマルオープンである場合には、提供されたジャンパヘッドを使用して、（赤で示したように）2つのピンを接続します
カード検出回路をノーマルオフ		カードスロットの検出オプションがノーマルオフである場合には、提供されたジャンパヘッドを使用して、（赤で示したように）2つのピンを接続します

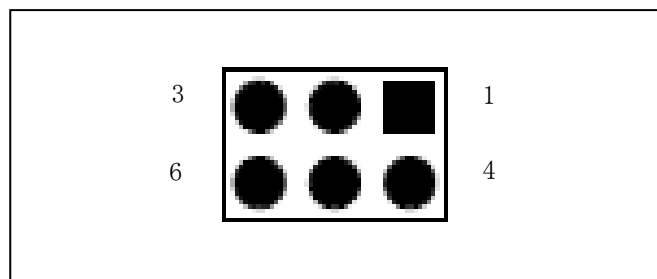
### 6.3. J4: USB ケーブルコネクタ



図示5 : USB ケーブルコネクタ

コネクタ	説明
1	USB VCC
2	USB D-
3	USB D+
4	USB GND
5	USB シールド

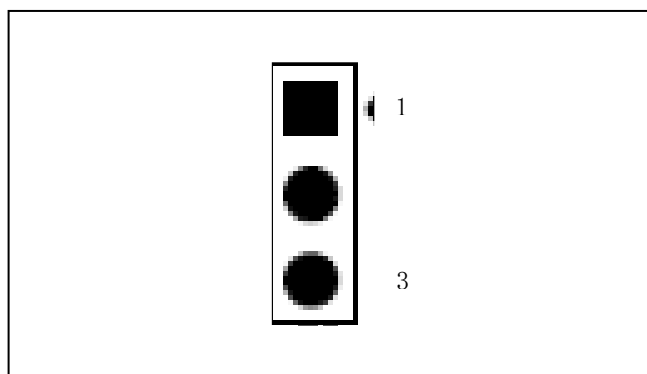
### 6.4. J8: USB ピンアウトコネクタ



図示6 : USB ピンアウトコネクタ

コネクタ	説明
1、3、4	USB GND
2	USB D+
5	USB D-
6	USB VCC

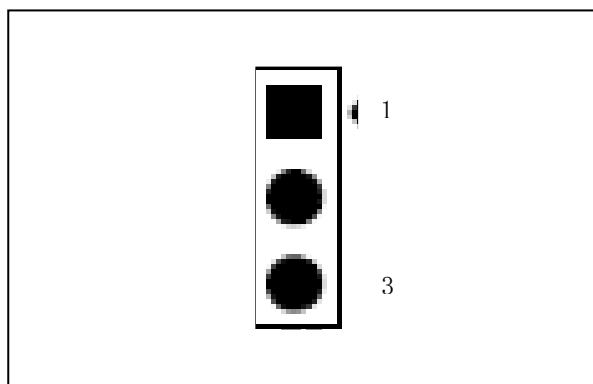
## 6.5. J9 : カードの接続信号コネクタ



図示7 : カードの接続信号コネクタ

コネクタ	説明
1	USB VCC
2	接続していない
3	MCU から接続された信号

## 6.6. J10 : USB 電源信号コネクタ



図示8 : USB 電源信号コネクタ

コネクタ	説明
1	USB VCC
2	接続していない
3	GND

## 7.0. USB インターフェース

### 7.1. 通信パラメーター

ACM39U-Y は USB 2.0 仕様の USB インターフェースを介してコンピュータに接続されます。フルスピードモードをサポートできて、12 mbps で働いています。USB インターフェースの詳細については、セクション 6 を参照してください。

ピン	信号	機能
1	V <sub>BUS</sub>	カードに+5 V の電源を供給
2	D-	ACM39U-Y と PC は差動信号でデータを転送します。
3	D+	ACM39U-Y と PC は差動信号でデータを転送します。
4	GND	電源用の参照電圧レベル

**表2** : USB インターフェース配線

**注意** : ACM39U-Y が USB インターフェースを介して、通常に動作させるように、ACS PC/SC ドライバをインストールしなければなりません。

### 7.2. エンドポイント

ACM39U-Y が下記のエンドポイントを介して、ホストの PC と通信します :

<b>エンドポイント制御 (Control Endpoint)</b>	設置と制御のため
<b>バルクアウト (Bulk OUT)</b>	ホストから ACM39U-Y に送信するコマンドに対して (データパケットサイズは 64 バイト)
<b>バルクイン (Bulk IN)</b>	ACM39U-Y からホストに返す応答に対して (データパケットサイズは 64 バイト)
<b>割り込み入力 (Interrupt IN)</b>	ACM39U-Y からホストに送信する状態メッセージに対して (データパケットサイズは 8 バイト)



## 8.0. 通信プロトコル

ACM39U-Y は USB を介して、ホストとのインターフェースを確立します。業界の規範 - CCID 標準は、USB チップ - スマートカードインタフェース装置に関わっているプロトコルを定義します。CCID 仕様はスマートカードを動作させるために必要な全てのプロトコルをカバーしています。

ACM39U-Y の USB エンドポイントの装置と使用は CCID 標準の Rev 1.0 のパート 3 に準拠するはずです。

概要を以下に要約されています：

1. **制御コマンド**制御パイプ（デフォルトのパイプ）で送信されます。特定な請求と USB 規格の請求を含めます。デフォルトのパイプで送信されたコマンドはデフォルトのパイプでホストにレポート情報を返します。
2. **CCID イベント**割り込みパイプで送信されます。
3. **CCID コマンド** BULK-OUT エンドポイントで送信されます。ACM39U-Y に送信された全てのコマンドは自分の関係エンディング応答を持っています。いくつかのコマンドは中間応答も持っています。
4. **CCID 応答** BULK-IN エンドポイントで送信されます。ACM39U-Y に送信された全てのコマンドは必ず同期に送信されます。（例：ACM39U-Y にとって、bMaxCCIDBusySlots は 01 h に相当です）。

ACM39U-Y がサポートしている CCID 特性は下記のクラス記述子を参照します：

オフセット	データフィールド	大きさ	数値	説明
0	<i>bLength</i>	1		記述子のバイト。
1	<i>bDescriptorType</i>	1		CCID 機能記述子のタイプ。
2	<i>bcdCCID</i>	2		2 進化した 10 進数での CCID 仕様のリリース番号。
4	<i>bMaxSlotIndex</i>	1		ACM39U-Y が一つのカードスロットを持っている
5	<i>bVoltageSupport</i>	1		ACM39U-Y は 1.8V、3.0V と 5V のスロットサポート。
6	<i>dwProtocols</i>	4		ACM39U-Y は T=0 および T=1 プロトコルをサポートできる。
10	<i>dwDefaultClock</i>	4		デフォルトの ICC クロック周波数は 4.8 MHz です。
14	<i>dwMaximumClock</i>	4		ICC サポートできるクロック周波数は 4.8 MHz です。
18	<i>bNumClockSupported</i>	1		クロック周波数の手動設定をサポートしていません。
19	<i>dwDataRate</i>	4		デフォルトの ICC I/O ボーレートは 12918 bps です。
23	<i>dwMaxDataRate</i>	4		ICC I/O サポートできる最大のボーレートは 826 kbps です。
27	<i>bNumDataRatesSupported</i>	1		ボーレートの手動設定をサポートしていません。



オフ セット	データフィールド	大き さ	数値	説明
28	<i>dwMaxIFSD</i>	4		ACM39U-Y T1 サポートできる最大 IFSD は 247 です。
32	<i>dwSynchProtocols</i>	4		ACM39U-Y は同期カードをサポートできません。
36	<i>dwMechanical</i>	4		ACM39U-Y は特別な機械的特性をサポートできません。
40	<i>dwFeatures</i>	4		ACM39U-Y が下記の特性をサポートできます： <ul style="list-style-type: none"><li>パラメーターに基づいて、自動的に ICC のクロック周波数を変更します</li><li>周波数と FI、DI パラメーターに基づいて、自動的ボーレートを変更します</li><li>ACM39U-Y との TPDU レベル変更</li></ul>
44	<i>dwMaxCCIDMessageLength</i>	4		ACM39U-Y が受け入れられる最大なメッセージの長さは 271 バイトです。
48	<i>bClassGetResponse</i>	1		TPDU レベル交換に影響しません。
49	<i>bClassEnvelope</i>	1		TPDU レベル交換に影響しません。
50	<i>wLCDLayout</i>	2		LCD なし。
52	<i>bPINSupport</i>	1		PIN 認証サポート。
53	<i>bMaxCCIDBusySlots</i>	1		同じ時間に一つだけのスロットが動作状態になります。

## 9.0. メモリカードタイプの選択

メモリカードに他のコマンドを実行する前に、SELECT\_CARD\_TYPE を実行しなければなりません。このコマンドはカードリーダーに挿入されて、選択したカードにパワーダウン/アップを実行する。同時にリセットを実行する時に使われる。SCardConnect( ) API によって確立されたロジックなスマートカードリーダーが通信した後に使用しかできません。

**注：** SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。メモリカードのコマンド設定については、本書の **10.3** 節を参照してください。。

下記のプログラムコードスニペットは ACM39U-Y でメモリカードのタイプの選択方法を示します：

```
SCARDCONTEXT hContext;
SCARDHANDLE hCard;
unsigned long dwActProtocol;
SCARD_IO_REQUEST ioRequest;
DWORD size = 64, SendLen = 6, RecvLen = 255, retCode;
byte cardType;
//Establish PC/SC Connection
retCode = SCardEstablishContext (SCARD_SCOPE_USER, NULL, NULL,
&hContext);
//List all readers in the system
retCode = SCardListReaders (hContext, NULL, readerName, &size);
//Connect to the reader
retCode = SCardConnect(hContext, readerName, SCARD_SHARE_SHARED,
SCARD_PROTOCOL_T0, &hCard, &dwActProtocol);
//Select Card Type
unsigned char SendBuff[] = {FF,A4,00,00,01,cardType};
retCode = SCardTransmit( hCard, &ioRequest, SendBuff, SendLen, NULL,
RecvBuff, &RecvLen);
//Disconnect from the reader
retCode = SCardDisconnect(hCard, SCARD_UNPOWER_CARD);
//End the established context
retCode = SCardReleaseContext(hContext);
```

## 10.0. コマンド

### 10.1. CCID コマンドパイプ、バルクアウトメッセージ

ACM39U-Y は、1.0 バージョンの CCID 仕様のセクション 4.1 で Bulk-OUT メッセージに従わなければなりません。また、この仕様はいくつかの追加機能用の拡張コマンドを定義しています。

本節は ACM39U-Y がサポートしている CCID タイプの Bulk-OUT メッセージをリストします。

#### 10.1.1. PC\_to\_RDR\_lccPowerOn

このコマンドはスロットを活性化して、カードから ATR を返すために使われます。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	62h	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
2	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
5	<i>bSeq</i>	1		コマンドのシーケンス番号。
6	<i>bPowerSelect</i>	1		ICC に印加される電圧： 00h = 自動電圧選択 01h = 5 V 02h = 3 V
7	<i>abRFU</i>	2		保留して将来使います。

このコマンドメッセージの応答は *RDR\_to\_PC\_DataBlock* 応答メッセージです。返したデータはリセット応答（ATR）です。

#### 10.1.2. PC\_to\_RDR\_lccPowerOff

スロットの活性化をキャンセルする時、このコマンドを使います。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	63h	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>abRFU</i>	3		保留して将来使います。

このメッセージの応答 *RDR\_to\_PC\_SlotStatus* 応答メッセージです。

### 10.1.3. PC\_to\_RDR\_GetSlotStatus

現在のスロットの状態情報を取得する時に、このコマンドを使います。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	65h	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>abRFU</i>	3		保留して将来使います。

このメッセージの応答 *RDR\_to\_PC\_SlotStatus* 応答メッセージです。

### 10.1.4. PC\_to\_RDR\_XfrBlock

ICC にデータブロックを転送する時にこのコマンドを使用します。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	6Fh	
1	<i>dwLength</i>	4		このメッセージの <i>abData</i> データフィールドのサイズ
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>bBWI</i>	1		現在転送している CCIDs ブロックの待機タイムアウトを拡張するために使用します。「この数値にブロックの待機時間を掛け」の期限が切れた後、CCID はブロックをタイムアウトにします。
8	<i>wLevelParameter</i>	2	0000h	RFU (TPDU 交換レベル)。
10	<i>abData</i>	バイト配列		CCID に送信されるデータブロック。データは ICC に「そのまま」送信されます (TPDU 交換レベル)

このメッセージの応答 *RDR\_to\_PC\_DataBlock* 応答メッセージです。

### 10.1.5. PC\_to\_RDR\_GetParameters

スロットのパラメーターを取得する時にこのコマンドを使用します。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	6Ch	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>abRFU</i>	3		保留して将来使います。

このメッセージの応答 *RDR\_to\_PC\_Parameters* 応答メッセージです。

### 10.1.6. PC\_to\_RDR\_ResetParameters

スロットのパラメーターをデフォルト値に戻す時ニコのコマンドを使用します。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	6Dh	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>abRFU</i>	3		保留して将来使います。

このメッセージの応答 *RDR\_to\_PC\_Parameters* 応答メッセージです。

### 10.1.7. PC\_to\_RDR\_SetParameters

スロットのパラメーターを設置する時にこのコマンドを使用します。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	61h	
1	<i>dwLength</i>	4		メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		このコマンドのスロット番号を識別します。
6	<i>bSeq</i>	1		コマンドのシーケンス番号。
7	<i>bProtocolNum</i>	1		下記は指定されたプロトコルのデータ構造です。 00h = T=0 プロトコル構造 01h = T=1 プロトコル構造 下記の値を保留して将来使います： 80h = 2 線プロトコルの構造 81h = 3 線プロトコルの構造 82h = I2C プロトコル構造
8	<i>abRFU</i>	2		保留して将来使います。
10	<i>abProtocolDataStructure</i>	バイト配列		プロトコルのデータ構造



T=0 プロトコルのデータ構造 (dwLength=00000005h)

オフセット	データフィールド	大きさ	数値	説明
10	<i>bmFindexDindex</i>	1		B7-4 – FI – ISO/IEC 7816-3:1997 中のテーブル 7 をインデックスして、クロックレートの変換係数を選択します B3-0 – DI - ISO/IEC 7816-3:1997 中のテーブル 8 をインデックスして、ボーレートの変換係数を選択します
11	<i>bmTCCCKST0</i>	1		B0 – 0b, B7-2 – 0000000b B1 – 使用している約束 (b1=0 : 直接 ; b1=1 : 逆) <b>注 :</b> CCID がこのビットを無視します。
12	<i>bGuardTimeT0</i>	1		2 文字間の余分な GuardTime。正常な GuardTime (12 ETU) に 0–254 ETU を追加します。FFh と 00h が同じです。
13	<i>bWaitingIntegerT0</i>	1		T=0 の場合 WI が WWT を定義する時に使われます
14	<i>bClockStop</i>	1		ICC クロック停止サポート 00h = クロックを停止することは許可されていません 01h = クロック信号が低い時に停止されます 02h = クロック信号が高い時に停止されます 03h = クロック信号が低い時または高い時に停止されます

T=1 プロトコルのデータ構造 (dwLength=00000007h)

オフセット	データフィールド	大きさ	数値	説明
10	<i>bmFindexDindex</i>	1		B7-4 – FI – ISO/IEC 7816-3:1997 中のテーブル 7 をインデックスして、クロックレートの変換係数を選択します B3-0 – DI – ISO/IEC 7816-3:1997 中のテーブル 8 をインデックスして、ボーレートの変換係数を選択します
11	<i>bmTCKKST1</i>	1		B7-2 – 000100b B0 – チェックサムタイプ (b0=0 : LRC ; b0=1 : CRC) B1 – 使用している約束 (b1=0 : 直接 ; b1=1 : 逆) <b>注 :</b> CCID がこのビットを無視します。
12	<i>bGuardTimeT1</i>	1		余計な GuardTime (2 文字間の余分な GuardTime は 0 – 254etu) は FFh である場合、GuardTime を 1 etu 減らします。
13	<i>bWaitingIntegerT1</i>	1		B7-4 = BWI 値 0-9 有効 B3-0 = CWI 値 0-Fh 有効
14	<i>bClockStop</i>	1		ICC クロック停止サポート 00h = クロックを停止することは許可されていません 01h = クロック信号が低い時に停止されます 02h = クロック信号が高い時に停止されます 03h = クロック信号が低い時または高い時に停止されます
15	<i>bIFSC</i>	1		交渉された IFSC の大きさ
16	<i>bNadValue</i>	1	00h	NAD = 00h だけサポートできます

このメッセージの応答 *RDR\_to\_PC\_Parameters* 応答メッセージです。



## 10.2. CCID Bulk-IN メッセージ

Bulk-IN メッセージは Bulk-OUT メッセージに応答するため使われます。ACM39U-Y は CCID プロトコルのパート 4.2 が定義している Bulk-IN メッセージを準拠するはずですが。

本節は ACM39U-Y がサポートしている CCID タイプの Bulk-IN メッセージをリストします。

### 10.2.1. RDR\_to\_PC\_DataBlock

このコマンドは ACM39U-Y によって送信されて、*PC\_to\_RDR\_IccPowerOn* と *PC\_to\_RDR\_XfrBlock* メッセージに対する応答です。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	80h	CCID によってデータブロックを送信しています。
1	<i>dwLength</i>	4		メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		Bulk-OUT メッセージ中の値と同じです。
6	<i>bSeq</i>	1		Bulk-OUT メッセージ中の値と同じです。
7	<i>bStatus</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
8	<i>bError</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
9	<i>bChainParameter</i>	1	00h	RFU (TPDU 交換レベル)。
10	<i>abData</i>	バイト配列		このデータフィールドは CCID から返したデータを含めています。

### 10.2.2. RDR\_to\_PC\_SlotStatus

このコマンドは ACM39U-Y によって送信されて、PC\_to\_RDR\_IccPowerOff と PC\_to\_RDR\_GetSlotStatus メッセージに対する応答です。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	81h	
1	<i>dwLength</i>	4	00000000h	メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		Bulk-OUT メッセージ中の値と同じです。
6	<i>bSeq</i>	1		Bulk-OUT メッセージ中の値と同じです。
7	<i>bStatus</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
8	<i>bError</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
9	<i>bClockStatus</i>	1		数値 : 00h = クロック動作中 01h = クロックが低い状態に止まっている 02h = クロックが高い状態に止まっている 03h = 不明な状態に止まっている 残された値を保留して将来使います。

### 10.2.3. RDR\_to\_PC\_Parameters

このコマンドは ACM39U-Y によって送信されて、PC\_to\_RDR\_GetParameters、

PC\_to\_RDR\_ResetParameters および PC\_to\_RDR\_SetParameters メッセージに対する応答です。

オフセット	データフィールド	大きさ	数値	説明
0	<i>bMessageType</i>	1	82h	
1	<i>dwLength</i>	4		メッセージの余分なバイトのサイズ。
5	<i>bSlot</i>	1		Bulk-OUT メッセージ中の値と同じです。
6	<i>bSeq</i>	1		Bulk-OUT メッセージ中の値と同じです。
7	<i>bStatus</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
8	<i>bError</i>	1		CCID 規格 (Rev 1.0) 4.2.1 節に定義されたスロットステータスレジスタ
9	<i>bProtocolNum</i>	1		<p>下記は指定されたプロトコルのデータ構造です。</p> <p>00h = T=0 プロトコルの構造</p> <p>01h = T=1 プロトコルの構造</p> <p>下記の値を保留して将来使います：</p> <p>80h = 2 線プロトコルの構造</p> <p>81h = 3 線プロトコルの構造</p> <p>82h = I2C プロトコル構造</p>
10	<i>abProtocolDataStructure</i>	バイト配列		プロトコルのデータ構造

### 10.3. メモリカードのコマンドセット

ここでは、ACM39U-Y のメモリーカードコマンドセットを紹介します。

#### 10.3.1. メモリカード – 1、2、4、8 および 16 kilobit I2C カード

##### 10.3.1.1. SELECT\_CARD\_TYPE

このコマンドはカードリーダーに挿入されて、選択したカードにパワーダウン/アップを実行する。同時にリセットを実行する時に使われる。

**注釈：** SCardConnect( ) API によって確立されたロジックなスマートカードリーダー通信後に使用しかできません。

SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Card Type
FFh	A4h	00h	00h	01h	01h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

##### 10.3.1.2. SELECT\_PAGE\_SIZE

このコマンドはスマートカードを読み取られるページサイズを選択する。デフォルト値は 8 バイトの書き込みページ。カードを取り外してリーダーの電源をオフにすると、カードはデフォルト値にリセットされます。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Page Size
FFh	01h	00h	00h	01h	

その中：

Page size

- = 03h : 8 バイトの書き込みページ
- = 04h : 16 バイトの書き込みページ
- = 05h : 32 バイトの書き込みページ
- = 06h : 64 バイトの書き込みページ
- = 07h : 128 バイトの書き込みページ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.1.3. READ\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	Byte Address		MEM_L
		MSB	LSB	
FFh	B0h			

その中 :

**Byte Address**      メモリカードのメモリアドレス位置  
**MEM\_L**              メモリカード中の読み取られていないデータの長さ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

BYTE 1	...	...	BYTE N	SW1	SW2

その中 :

**BYTE x**      メモリカードから読み出されたデータ  
**SW1 SW2**      = 90 00h (エラーなしの場合)

### 10.3.1.4. WRITE\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte n
		MSB	LSB					
FFh	D0h							

その中 :

**Byte Address**      メモリカードのメモリアドレス位置  
**MEM\_L**              メモリカードから読み出されていないデータの長さ  
**BYTE X**              メモリカードに書き入れているデータ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中 :

**SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.2. メモリカード – 32、64、128、256、512 および 1024 kilobit I2C カード

#### 10.3.2.1. SELECT\_CARD\_TYPE

このコマンドはカードリーダーに挿入されて、選択したカードにパワーダウン/アップを実行する。同時にリセットを実行する時に使われる。

**注釈：** SCardConnect( ) API によって確立されたロジックなスマートカードリーダー通信後に使用しかできません。SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Card Type
FFh	A4h	00h	00h	01h	02h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.2.2. SELECT\_PAGE\_SIZE

このコマンドはスマートカードを読み取られるページサイズを選択する。デフォルト値は 8 バイトの書き込みページ。カードを取り外してリーダーの電源をオフにすると、カードはデフォルト値にリセットされます。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Page size
FFh	01h	00h	00h	01h	

その中：

**Data**                      カードに送信されていない TPDU

**Page size**                = 03h : 8 バイトの書き込みページ  
                               = 04h : 16 バイトの書き込みページ  
                               = 05h : 32 バイトの書き込みページ  
                               = 06h : 64 バイトの書き込みページ  
                               = 07h : 128 バイトの書き込みページ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.2.3. READ\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	Byte Address		MEM_L
		MSB	LSB	
FFh				

その中：

**INS** = B0h : 32、64、128、256 および 512 kilobit IIC カード  
 = 1011 000\*b : 1024 kilobit IIC カード,  
 その中 \* はアドレッシング 17 ビットの MSB を示している。

**Byte Address** メモリカードのメモリアドレス位置

**MEM\_L** メモリカード中の読み取られていないデータの長さ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

BYTE 1	...	...	BYTE N	SW1	SW2

その中：

**BYTE x** メモリカードから読み出されたデータ

**SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.2.4. WRITE\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte n
		MSB	LSB					
FFh								

その中：

**INS** = D0h : 32、64、128、256 および 512 kilobit IIC カード  
 = 1101 000\*b : 1024 kilobit IIC カード,  
 その中 \* はアドレッシング 17 ビットの MSB を示している。

**Byte Address** メモリカードのメモリアドレス位置

**MEM\_L** メモリに書き入れているデータの長さ

**Byte x** メモリカードに書き入れているデータ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.3. メモリカード – SLE 4418/SLE 4428/SLE 5518/SLE 5528

#### 10.3.3.1. SELECT\_CARD\_TYPE

このコマンドはカードリーダーに挿入されて、選択したカードにパワーダウン/アップを実行する。同時にリセットを実行する時に使われる。

**注釈：** SCardConnect( ) API によって確立されたロジックなスマートカードリーダー通信後に使用しかできません。  
SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Card Type
FFh	A4h	00h	00h	01h	05h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.3.2. READ\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	Byte Address		MEM_L
		MSB	LSB	
FFh	B0h			

その中：

**MSB Byte Address** = 0000 00A<sub>9</sub>A<sub>8</sub>b はメモリカードのメモリアドレス位置である

**LSB Byte Address** = A<sub>7</sub>A<sub>6</sub>A<sub>5</sub>A<sub>4</sub> A<sub>3</sub>A<sub>2</sub>A<sub>1</sub>A<sub>0</sub>b はメモリカードのメモリアドレス位置である

**MEM\_L** = メモリカード中の読み取られていないデータの長さ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

BYTE 1	...	...	BYTE N	SW1	SW2

その中：

**BYTE x** = メモリカードから読み出されたデータ

**SW1 SW2** = 90 00h (エラーなしの場合)



### 10.3.3.3. READ\_PRESENTATION\_ERROR\_COUNTER\_MEMORY\_CARD (SLE 4428 と SLE 5528)

このコマンドがプレゼンテーションエラーカウンタを読み取る時に使われる。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	P1	P2	MEM_L
FFh	B1h	00h	00h	03h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

ERRCNT	DUMMY 1	DUMMY 2	SW1	SW2

その中：

- ERRCNT** エラー カウンター。FFh は最後の検証が正しいことを示している。00 h はパスワードがロックされていることを示している（最大再試行回数を超過した）。他の値は最後の認証が失敗したことを示している。
- DUMMY** 从卡片读取的 2 个字节的虚拟数据
- SW1 SW2** = 90 00h (エラーなしの場合)

### 10.3.3.4. READ\_PROTECTION\_BIT

コマンドフォーマット (*PC\_to\_RDR\_XfrBlock* 中の *abData* データフィールド)

Pseudo-APDU				
CLA	INS	Byte Address		MEM_L
		MSB	LSB	
FFh	B2h			

その中：

**MSB Byte Address** = 0000 00A<sub>9</sub>A<sub>8</sub>b はメモ리카ードのメモリアドレス位置である

**LSB Byte Address** = A<sub>7</sub>A<sub>6</sub>A<sub>5</sub>A<sub>4</sub> A<sub>3</sub>A<sub>2</sub>A<sub>1</sub>A<sub>0</sub>b はメモ리카ードのメモリアドレス位置である

**MEM\_L** カードから読み出される保護ビットの長さ (8ビットの倍数で、最大は 32 です)

。MEM\_L = 1 + INT [(number of bits – 1)/8]  
例えば、メモリ 0010H から始まりの 8 保護ビットを読み取るために、下記の pseudo-APDU を発行する必要がある：  
FF B1 00 10 01h

応答データフォーマット (*RDR\_to\_PC\_DataBlock* 中の *abData* データフィールド)

PROT 1	...	...	PROT L	SW1	SW2

その中：

**PROT y** 保護ビットが含まれているバイト

**SW1、SW2** = 90 00h (エラーなしの場合)

PROT バイト中で、保護ビットは以下のように並べている：

PROT 1								PROT 2								...								
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	..	P18	P17

その中：

**Px** は応答データの BYTE x の保護ビットです。

‘0’バイトが書き込み保護されている

‘1’バイトは書き込むことができる

### 10.3.3.5. WRITE\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte N
		MSB	LSB					
FFh	D0h							

その中：

**MSB Byte Address**= 0000 00A<sub>9</sub>A<sub>8</sub>b はメモ리카ードのメモリアドレス位置である

**LSB Byte Address** = A<sub>7</sub>A<sub>6</sub>A<sub>5</sub>A<sub>4</sub> A<sub>3</sub>A<sub>2</sub>A<sub>1</sub>A<sub>0</sub>b はメモ리카ードのメモリアドレス位置である

**MEM\_L** メモリに書き入れているデータの長さ

**Byte x** メモ리카ードに書き入れているデータ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1**、**SW2** = 90 00h (エラーなしの場合)

### 10.3.3.6. WRITE\_PROTECTION\_MEMORY\_CARD

コマンドで指定された各バイトは、内部でカードに指定されたアドレス中のデータと比べます。一致した場合、対応している保護ビットが不可逆的に“0”にプログラムされている。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	Byte Address		MEM_L	Byte 1	....	....	Byte N
		MSB	LSB					
FFh	D1h							

その中：

**MSB Byte Address**= 0000 00A<sub>9</sub>A<sub>8</sub>b はメモ리카ードのメモリアドレス位置である

**LSB Byte Address** = A<sub>7</sub>A<sub>6</sub>A<sub>5</sub>A<sub>4</sub> A<sub>3</sub>A<sub>2</sub>A<sub>1</sub>A<sub>0</sub>b はメモ리카ードのメモリアドレス位置である

**MEM\_L** メモリに書き入れているデータの長さ

**Byte x** バイト値が *Byte Address* から始まるカード内のデータと比較されます。BYTE 1 と *Byte Address* 中のデータを比べる ; BYTE N と (*Byte Address* + N - 1) 中のデータが比べる。

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.3.7. PRESENT\_CODE\_MEMORY\_CARD (SLE 4428 および SLE 5528)

SLE4428 と SLE5528 に書き込む操作を有効にするために、メモ리카ードにシークレットコードを提出する時に、このコマンドを使用します。以下の操作を実行する：

1. プレゼンテーションエラーカウンタにビット‘1’を検索して、‘0’に変更する。
2. 指定されたシークレットコードをカードに提出する。
3. プレゼンテーションエラーカウンタを消去する。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU						
CLA	INS	P1	P2	MEM_L	CODE	
					Byte 1	Byte 2
FFh	20h	00h	00h	02h		

その中：

**CODE** 2 バイトのシークレットコード (PIN)

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2 ErrorCnt
90h	

その中：

**SW1** = 90h

**SW2 (ErrorCnt)** = エラー カウンター。FFh は認証が成功したことを示している。00 h はパスワードがロックされていることを示している（最大再試行回数を超過した）。他の値は現在の認証が失敗したことを示している。

### 10.3.4. メモリカード – SLE 4432/SLE 4442/SLE 5532/SLE 5542

#### 10.3.4.1. SELECT\_CARD\_TYPE

このコマンドはカードリーダーに挿入されて、選択したカードにパワーダウン/アップを実行する。同時にリセットを実行する時に使われる。

**注釈：** SCardConnect( ) API によって確立されたロジックなスマートカードリーダー通信後に使用しかできません。  
SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU					
CLA	INS	P1	P2	Lc	Card Type
FFh	A4h	00h	00h	01h	06h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.4.2. READ\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	P1	Byte Address	MEM_L
FFh	B0h	00h		

その中：

**Byte Address** = A<sub>7</sub>A<sub>6</sub>A<sub>5</sub>A<sub>4</sub> A<sub>3</sub>A<sub>2</sub>A<sub>1</sub>A<sub>0</sub> はメモリカードのメモリアドレス位置である

**MEM\_L** = メモリカード中の読み取られていないデータの長さ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

BYTE 1	...	BYTE N	PROT 1	PROT 2	PROT 3	PROT 4	SW1	SW2

その中：

**BYTE x** = メモリカードから読み出されたデータ

**PROT y** = 保護ビットが含まれているバイト

**SW1 SW2** = 90 00h (エラーなしの場合)

PROT バイト中で、保護ビットは以下のように並べている：

PROT 1								PROT 2								...								
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9								P18	P17

その中：

**Px** は応答データの BYTE x の保護ビットです。

‘0’バイトが書き込み保護されている

‘1’バイトは書き込むことができる

#### 10.3.4.3. READ\_PRESENTATION\_ERROR\_COUNTER\_MEMORY\_CARD (SLE 4442 と SLE 5542)

このコマンドがプレゼンテーションエラーカウンタを読み取る時に使われる。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	P1	P2	MEM_L
FFh	B1h	00h	00h	04h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

ERRCNT	DUMMY 1	DUMMY 2	DUMMY 3	SW1	SW2

その中：

- ERRCNT** エラー カウンター。07h は最後の検証が正しいことを示している。00H はパスワードがロックされていることを示している（最大再試行回数を超過した）。他の値は最後の認証が失敗したことを示している。
- DUMMY** 从卡片读取的 3 个字节的虚拟数据
- SW1 SW2** = 90 00h（エラーなしの場合）

#### 10.3.4.4. READ\_PROTECTION\_BITS

このコマンドは始めの 32 バイトの保護ビットを読み取る時に使われる。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	P1	P2	MEM_L
FFh	B2h	00h	00h	04h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

PROT 1	PROT 2	PROT 3	PROT 4	SW1	SW2

その中：

- PROT y** 保護ビットが含まれているバイト
- SW1、SW2** = 90 00h（エラーなしの場合）

PROT バイト中で、保護ビットは以下のように並べている：

PROT 1								PROT 2								...								
P8	P7	P6	P5	P4	P3	P2	P1	P16	P15	P14	P13	P12	P11	P10	P9	..	..	..	..	..	..	..	P18	P17

その中：

- Px** は応答データの BYTE x の保護ビットです。
- ‘0’バイトが書き込み保護されている
- ‘1’バイトは書き込むことができる

#### 10.3.4.5. WRITE\_MEMORY\_CARD

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	P1	Byte Address	MEM_L	Byte 1	....	....	Byte N
FFh	D0h	00h						

その中：

**Byte Address** =  $A_7A_6A_5A_4A_3A_2A_1A_0b$  はメモ리카ードのメモリアドレス位置である

**MEM\_L** メモリに書き入れているデータの長さ

**Byte x** メモ리카ードに書き入れているデータ

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.4.6. WRITE\_PROTECTION\_MEMORY\_CARD

コマンドで指定された各バイトは内部でカードに指定されたアドレス中のデータと比べる。一致した場合、対応している保護ビットが不可逆的に“0”にプログラムされている。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU								
CLA	INS	P1	Byte Address	MEM_L	Byte 1	....	....	Byte N
FFh	D1h	00h						

その中：

**Byte Address** =  $000A_4A_3A_2A_1A_0b$  (00h - 1Fh) はメモ리카ードの保護メモリアドレス位置である

**MEM\_L** メモリに書き入れているデータの長さ

**Byte x** バイト値がバイトアドレスから始まるカード内のデータと比較される。BYTE 1 と Byte Address 中のデータを比べる；BYTE N と (Byte Address + N - 1) 中のデータが比べる。

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)

#### 10.3.4.7. PRESENT\_CODE\_MEMORY\_CARD (SLE 4442 および SLE 5542)

SLE4442 と SLE5542 に書き込む操作を有効にするために、メモ리카ードにシークレットコードを提出する時に、このコマンドを使用します。以下の操作を実行する：

1. プレゼンテーションエラーカウンタにビット‘1’を検索して、‘0’に変更する。
2. 指定されたシークレットコードをカードに提出する。
3. プレゼンテーションエラーカウンタを消去する。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU							
CLA	INS	P1	P2	MEM_L	CODE		
					Byte 1	Byte 2	Byte 3
FFh	20h	00h	00h	03h			

その中：

**CODE** 3 バイトのシークレットコード (PIN)

応答データフォーマット (*RDR to PC DataBlock* 中の *abData* データフィールド)

SW1	SW2 ErrorCnt
90h	

その中：

**SW1** = 90h

**SW2 (ErrorCnt)** = エラー カウンター。07h は検証が正しいことを示している。00H はパスワードがロックされていることを示している（最大再試行回数を超過した）。他の値は現在の認証が失敗したことを示している。

#### 10.3.4.8. CHANGE CODE MEMORY CARD (SLE 4442 と SLE 5542)

指定されたデータを新しいシークレットコードとして、カードに書き入れる時に、このコマンドを使用します。  
PRESENT CODE コマンドでカードに現在のシークレットコードを提出してから、このコマンドを実行します。

コマンドフォーマット (PC to RDR XfrBlock 中の abData データフィールド)

Pseudo-APDU							
CLA	INS	P1	P2	MEM_L	CODE		
					Byte 1	Byte 2	Byte 3
FFh	D2h	00h	01h	03h			

応答データフォーマット (RDR to PC DataBlock 中の abData データフィールド)

SW1	SW2

その中：

**SW1 SW2** = 90 00h (エラーなしの場合)



## 10.4. PC\_to\_RDR\_XfrBlock を介して、他のコマンドを実行する

### 10.4.1. GET\_READER\_INFORMATION

このコマンドはリーダーのファームウェアのバージョンを取得する時に使われます。

**注釈：** SCardConnect( ) API によって確立された T=0 協議のロジックなスマートカードリーダー通信後に使用しかできません。SCardConnect( ) API についての詳しい説明は PC/SC 基準を参照してください。

コマンドフォーマット (PC\_to\_RDR\_XfrBlock 中の abData データフィールド)

Pseudo-APDU				
CLA	INS	P1	P2	Lc
FFh	09h	00h	00h	11h

応答データフォーマット (RDR\_to\_PC\_DataBlock 中の abData データフィールド)

FIRMWARE										

その中：

**FIRMWARE**    11 バイトのファームウェアバージョン番号

## 10.5. PC-to\_RDR\_Escape を介して、他のコマンドを実行する

### 10.5.1. GET\_READER\_INFORMATION

このコマンドはリーダーのファームウェアのバージョンを取得する時に使われます。

注：このコマンドは、ファームウェア 003R 以降を搭載した ACM39U でのみ使用できます。

Get Reader Information フォーマット (5 バイト)

コマンド	CLA	INS	P1	P2	Lc
Get Firmware Version	E0h	00h	00h	19h	00h

Get Reader Information 応答フォーマット (5 バイト+ファームウェア情報の長さ)

応答	CLA	INS	P1	P2	Le	データ出力
結果	E1h	00h	00h	00h	受信していないバイトの数量	ファームウェアのバージョン番号

例：

応答 = E1 00 00 00 0C 41 43 52 33 39 55 2D 30 2E 30 33 52

ファームウェアのバージョン番号 (HEX) = 41 43 52 33 39 55 2D 30 2E 30 33 52

ファームバージョン (ASCII) = "ACR39U-0.03R"



## Appendix A. 応答エラーコード

下記のテーブルは ACM39U-Y (CCID)が返す可能なエラーコードをまとめています：

エラーコード	状態
FFh	SLOTERROR_CMD_ABORTED
FEh	SLOTERROR_ICC_MUTE
FDh	SLOTERROR_XFR_PARITY_ERROR
FCh	SLOTERROR_XFR_OVERRUN
FBh	SLOTERROR_HW_ERROR
F8h	SLOTERROR_BAD_ATR_TS
F7h	SLOTERROR_BAD_ATR_TCK
F6h	SLOTERROR_ICC_PROTOCOL_NOT_SUPPORTED
F5h	SLOTERROR_ICC_CLASS_NOT_SUPPORTED
F4h	SLOTERROR_PROCEDURE_BYTE_CONFLICE
F3h	SLOTERROR_DEACTIVATED_PROTOCOL
F2h	SLOTERROR_BUSY_WITH_AUTO_SEQUENCE
E0h	SLOTERROR_CMD_SLOT_BUSY

表3：応答エラーコード

Android は Google Inc.の商標です。

EMV™は EMVCo LLC の商標です。

Atmel は Atmel Corporation また子会社がアメリカとまたはほかの国の登録商標です。

Infineon は Infineon Technologies AG の登録商標です。

Microsoft は Microsoft Corporation がアメリカとまたはほかの国の登録商標です。