



Advanced Card Systems Ltd.
Card & Reader Technologies

ACR1011

SIMicro (CCID)

智能卡及 Micro SD 卡读写器



参考手册 V1.01



目录

1.0.	简介	3
1.1.	SIM 尺寸智能卡读写器.....	3
1.2.	存储设备.....	3
1.3.	非接触特性.....	3
1.4.	易于集成.....	3
2.0.	特性	4
3.0.	系统框图	5
4.0.	电源	6
4.1.	LED 状态指示灯.....	6
5.0.	智能卡接口	7
5.1.	智能卡电源 VCC (C1).....	7
5.2.	编程电压 VPP (C6).....	7
5.3.	卡片类型选择.....	7
5.4.	微控制器卡接口.....	7
5.5.	卡片插拔保护.....	7
6.0.	USB 接口	8
6.1.	通信参数.....	8
6.2.	端点.....	8
7.0.	通信协议	9
7.1.	发送至 ACR101 (CCID)的命令.....	11
7.1.1.	CCID 命令通道 Bulk-OUT 消息.....	11
7.1.2.	CCID Bulk-IN 消息.....	16
7.1.3.	其它通过 PC_to_RDR_XfrBlock 进行的命令.....	18
7.2.	大容量存储.....	19
	附录 A.支持的卡片类型	20
	附录 B.响应报文状态码	21

图目录

图 1	: ACR100I 系统框图.....	5
-----	---------------------	---



1.0. 简介

ACR101I SIMicro (CCID) 不仅是一款简单的 SIM 尺寸智能卡读写器。它将智能卡读写器和 Micro SD 卡槽集成在一个轻便的 USB Token 中，能够支持高度安全的移动式应用。另外由于内置了一块 MIFARE® Classic (1K) 芯片，它还可以用于逻辑和物理访问控制等多种非接触式应用。ACR101I SIMicro (CCID) 读写器还有 HID 版本，由于无需安装驱动，同样可以令您体验到即插即用的便利性。



1.1. SIM 尺寸智能卡读写器

ACR101I SIMicro (CCID) 是一款外形纤巧而功能强大的读写器，可以支持符合 ISO 7816 标准的微处理器智能卡。其中包括绝大多数存储卡及符合 T=0 和 T=1 协议的微处理器卡。

由于将确保应用的安全性作为首要任务，它使您能够将公钥基础设施 (PKI) 等高度安全的技术集成到多种应用当中，最大限度地保护了敏感数据的安全。

1.2. 存储设备

ACR101I SIMicro (CCID) 读写器不仅拥有读写 SIM 尺寸智能卡的功能，还可以用作存储设备。它的尺寸仅为 72.0 mm x 26.0 mm x 11.7 mm，由于通过 USB 供电并且不需要任何连接线，可以方便的随身携带使用。ACR101I SIMicro (CCID) 能够支持最高 8 GB 的可扩展 Micro SD 内存。

1.3. 非接触特性

由于内置了一块 MIFARE Classic 1K 芯片，ACR101I SIMicro (CCID) 具有非接触式卡的功能。这使得用户能够将这款功能强大的设备灵活应用于物理和逻辑访问控制等多种应用。

1.4. 易于集成

ACR101I SIMicro (CCID) 符合 CCID 芯片/智能卡接口设备标准和 PC/SC 标准，因此无需安装驱动就可以很容易地集成到计算机环境中并进行使用。另外它还可以在运行了 Android™ 3.1 及以上版本平台的移动设备上使用。

这一系列特性使得 ACR101I SIMicro (CCID) 可以在多种应用领域中使用，例如公钥基础设施 (PKI)、网络安全和 GSM 管理等。



2.0. 特性

- USB 复合设备 - 兼用作智能卡读写器和大容量存储设备
- USB 2.0 高速接口
- 总线供电 - 无需额外电源或电池
- 即插即用 - 支持 CCID 标准，具有高度的灵活性
- 伸缩式 USB 接口
- 智能卡读写器：
 - 支持 ISO 7816 A 类、B 类和 C 类（5 V、3 V、1.8 V）SIM 尺寸卡
 - 支持符合 T=0 或 T=1 协议的微处理器卡
 - 通过同步卡 APDU 支持存储卡
 - 支持符合 GSM 11.11 规范的卡
 - 支持 PPS（协议和参数选择）
 - 具有短路保护功能
- 应用程序编程接口：
 - 支持 PC/SC
 - 支持 CT-API（通过 PC/SC 上一层的封装）
- 闪存：
 - 支持 Micro SD 卡
 - 最高 8 GB 内存
- 非接触特性：
 - 内置 MIFARE Classic 1K 芯片
- 支持 Android™ 3.1 及以上版本¹
- 符合下列标准：
 - ISO 7816
 - CE
 - FCC
 - VCCI
 - PC/SC
 - CCID
 - Microsoft® WHQL
 - RoHS 2
 - REACH

¹不适用 PC/SC 和 CCID 支持

3.0. 系统框图

USB 集线器控制器通过 USB 端口连接建立计算机和智能卡 MCU 以及闪存之间的通信接口。终端用户有 1-GB 闪存的存储空间。在 Windows Explorer 下，检测到设备时显示为可移动磁盘。ACR101 (CCID)采用 USB 端口供电，没有其他外接电源。

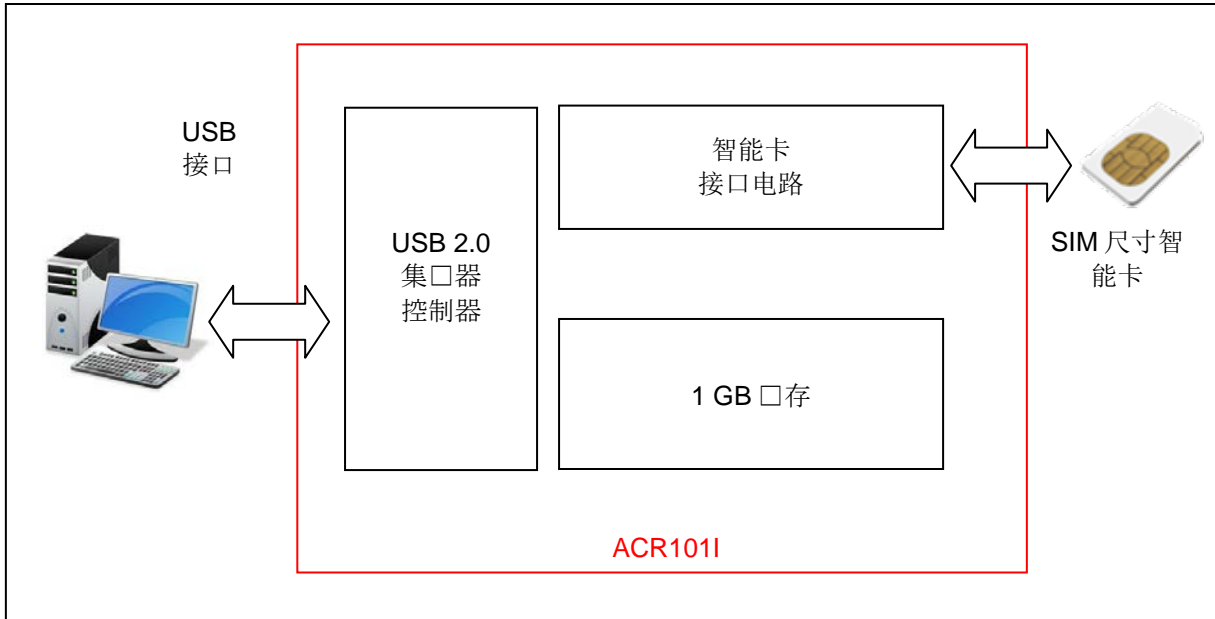


图1 : ACR100I 系统框图



4.0. 电源

ACR101 (CCID)需要 5 V, 100 mA 的直流稳压电源, 由计算机供电。

4.1. LED 状态指示灯

读写器前部的双色 LED 表示智能卡和闪存接口的激活状态。

绿色 LED:

缓慢闪烁 (每 2 秒钟开启 200 毫秒)

表示智能卡接口部分已上电并处于待机状态。智能卡尚未被插入, 或者智能卡尚未上电 (如果卡片已经插入)

长亮

表示已经开启向智能卡的供电, 即智能卡处于激活状态。

红色 LED:

长亮

表示 ACR101 (CCID)和闪存之间正在通信。



5.0. 智能卡接口

ACR101 (CCID)和所插入智能卡之间的接口符合 ISO 7816-3 标准。

5.1. 智能卡电源 VCC (C1)

插入的智能卡的电流消耗不得大于 50 mA。

5.2. 编程电压 VPP (C6)

根据 ISO 7816-3 的规定，由智能卡上的触点 C6 (VPP) 为智能卡提供编程电压。但由于市面上的智能卡大多数基于 EEPROM，不需要为其提供外部编程电压，ACR101 (CCID)的触点 C6 (VPP) 已被实现为普通的控制信号。此触点的电气规格与 RST 信号 (触点 C2) 的规格相同

5.3. 卡片类型选择

每次激活插入的卡片前，处于控制地位的计算机都要向 ACR101 (CCID)发送适当的命令来选择卡片类型。这些卡片包括存储卡和基于 MCU 的卡。对于基于 MCU 的卡片来说，读写器允许从 T=0 或 T=1 中选择首选的协议。但是只有当插入读写器的卡片对这两种协议类型都支持时，读写器才可以协议与参数选择 (PPS) 接受并执行这样的选择。如果 MCU 卡仅支持 T=0 或 T=1，则读卡器会自动采用该协议类型，而不管应用程序选择哪一种。

5.4. 微控制器卡接口

微控制器卡只使用触点 C1 (VCC)、C2 (RST)、C3 (CLK)、C5 (GND) 和 C7 (I/O)。时钟信号 (C3) 的频率为 4 MHz。

5.5. 卡片插拔保护

ACR101 (CCID)提供了一种机制来保护在上电状态下被突然拔出的卡片。当卡片被移出时，卡片的电源以及 ACR101 (CCID)与卡之间的信号线路会立即被取消激活。但是作为惯例，只应在断电后才应从读卡器移出卡片，这样可以避免电气损伤。

注：ACR101 (CCID)从不主动为插入的卡片上电。此操作必须由处于控制地位的计算机向读卡器发送适当的命令来进行。



6.0. USB 接口

ACR101 (CCID)通过符合 USB 标准的 USB 接口与计算机建立连接。

6.1. 通信参数

ACR101 (CCID)通过符合 USB 1.1 规范的 USB 端口与计算机建立连接，ACR101 (CCID)支持 USB 全速模式，闪存读写速率为 480 Mbps。

引脚	信号	功能
1	V _{BUS}	为读写器提供+5V 的电源
2	D-	ACR101 (CCID)和计算机间以微分信号传输数据
3	D+	ACR101 (CCID)和计算机间以微分信号传输数据
4	GND	参考电压等级

6.2. 端点

ACR101 (CCID)通过如下端点与主计算机进行通信：

Control Endpoint – 用于设置和控制

Bulk OUT Endpoint 1 – 用于从主机发送至大容量存储接口的命令（数据包大小为 64 字节）

Bulk IN Endpoint 1 – 用于从大容量存储接口发送至主机的响应（数据包大小为 64 字节）

Bulk OUT Endpoint 2 – 用于从主机发送至 CCID 接口的命令（数据包大小为 16 字节）

Bulk IN Endpoint 2 – 用于从 CCID 接口发送至主机的响应（数据包大小为 16 字节）

7.0. 通信协议

ACR101 (CCID)通过 USB 与主机建立连接。它是一个复合设备，包括两个接口：芯片卡接口设备和大容量存储。

ACR101 (CCID)智能卡接口的内核与 ACR38 CCID 智能卡接口的内核一样。获取读写器固件后，智能卡接口将被标识为“ACR38-112c”。CCID 涵盖了操作智能卡和 PIN 所需的全部协议。

注：后面章节将讨论与此相关的命令。

ACR101 (CCID) 智能卡接口的 USB 端点配置和使用应当符合 CCID 标准第 3 部分的规定。概述总结如下：

1. **控制命令**通过控制通道（缺省通道）发送,其中包括类特定请求和 USB 标准请求。由缺省通道发送的命令会通过缺省通道向主机反馈信息。
2. **CCID 事件**通过中断通道发送。
3. **CCID 命令**经由 *BULK-OUT* 端点发出。发送至智能卡读写器的每个命令都有一个相关的最终响应，一些命令也可以有中间响应。
4. **CCID 响应**由 *BULK-IN* 端点发送。所有发送至智能卡读写器的命令都必须同步发送。（即：*bMaxCCIDBusySlots* 等同于 1）。

ACR101 (CCID)智能卡接口支持的 CCID 功能由其类别描述符定义：

偏移	字段	大小	值	说明
0	<i>bLength</i>	1	36h	这个描述符的字节大小。
1	<i>bDescriptorType</i>	1	21h	CCID 功能描述符类型。
2	<i>bcdCCID</i>	2	0100h	CCID 以二进制编码的十进制指定的版本号。
4	<i>bMaxSlotIndex</i>	1	00h	具有一个卡槽
5	<i>bVoltageSupport</i>	1	07h	该款智能卡读写器支持 1.8 V、3.0 V 和 5.0 V 的槽位电压。
6	<i>dwProtocols</i>	4	00000003h	该款智能卡读写器支持 T=0 和 T=1 协议
10	<i>dwDefaultClock</i>	4	00000FA0h	默认 ICC 时钟频率为 4MHz
14	<i>dwMaximumClock</i>	4	00000FA0h	ICC 支持的最大时钟频率为 4MHz
18	<i>bNumClockSupported</i>	1	00h	不支持手动设置时钟频率
19	<i>dwDataRate</i>	4	00002A00h	默认 ICC I/O 波特率为 10752 bps
23	<i>dwMaxDataRate</i>	4	0001F808h	ICC I/O 支持的数据传输速率为 344 Kbps
27	<i>bNumDataRatesSupported</i>	1	00h	不支持手动设置数据传输速率
28	<i>dwMaxIFSD</i>	4	00000Feh	T=1 协议下，该智能卡读写器支持的最大 IFSD 为 254。



偏移	字段	大小	值	说明
32	<i>dwSynchProtocols</i>	4	00000000h	该款智能卡读写器不支持同步卡
36	<i>dwMechanical</i>	4	00000000h	该款智能卡读写器不支持特殊机制特性
40	<i>dwFeatures</i>	4	00010030h	该款智能卡读写器支持以下功能： <ul style="list-style-type: none">• 自动根据参数改变 ICC 时钟频率• 自动根据频率和 FI、DI 参数改变波特率• 与智能卡读写器进行 TPDU 级别交换
44	<i>dwMaxCCIDMessageLength</i>	4	0000010Fh	智能卡读写器可接受的最大报文长度为 271 字节
48	<i>bClassGetResponse</i>	1	00h	对 TPDU 级别的交换没有影响
49	<i>bClassEnvelope</i>	1	00h	对 TPDU 级别的交换没有影响
50	<i>wLCDLayout</i>	2	0000h	没有 LCD
52	<i>bPINSupport</i>	1	00h	无 PIN 校验
53	<i>bMaxCCIDBusySlots</i>	1	01h	同一时间内只能有 1 个槽位处于工作状态

7.1. 发送至 ACR101 (CCID)的命令

正常操作中，ACR101 (CCID) 在计算机和读写器通讯过程中充当从设备。通迅过程包括一连串的命令-应答：计算机发送一条命令给读卡器，并在命令执行后收到读卡器的应答。计算机只有在收到前一条命令的回应后才能给 ACR101 (CCID) 发送下一条命令。

没有收到计算机命令时，读写器能直接传输的数据只有读写器复位消息和卡片状态消息。

7.1.1. CCID 命令通道 Bulk-OUT 消息

ACR101 (CCID) 应遵循 CCID 协议第 4 部分定义的 CCID 类 *Bulk-OUT* 消息。CCID 还定义了操作附加功能的扩展命令。此节列举了 ACR101 (CCID) 支持的 CCID 类 *Bulk-OUT* 消息。

7.1.1.1. PC_to_RDR_lccPowerOn

激活卡槽并返回卡片的 ATR。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	62h	
1	<i>dwLength</i>	4	00000000h	此消息的额外字节的大小
2	<i>bSlot</i>	1		标识命令的插槽号
5	<i>bSeq</i>	1		命令的序号
6	<i>bPowerSelect</i>	1		用于 ICC 的电压 00h – 自动电压选择 01h – 5 伏 02h – 3 伏
7	<i>abRFU</i>	2		保留为将来使用

此命令消息的响应是 *RDR_to_PC_DataBlock* 响应消息，返回的是复位应答 (ATR) 数据。

7.1.1.2. PC_to_RDR_lccPowerOff

取消激活卡槽。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	63h	
1	<i>dwLength</i>	4	00000000h	此消息的额外字节的大小
5	<i>bSlot</i>	1		标识命令的插槽号
6	<i>bSeq</i>	1		命令的序号
7	<i>abRFU</i>	3		保留为将来使用

此消息的响应是 *RDR_to_PC_SlotStatus* 消息。

7.1.1.3. PC_to_RDR_GetSlotStatus

获取卡槽的当前状态。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	65h	
1	<i>dwLength</i>	4	00000000h	此消息的额外字节的大小
5	<i>bSlot</i>	1		标识命令的插槽号
6	<i>bSeq</i>	1		命令的序号
7	<i>abRFU</i>	3		保留为将来使用

此消息的响应是 *RDR_to_PC_SlotStatus* 消息。

7.1.1.4. PC_to_RDR_XfrBlock

向 ICC 传输数据块。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	6Fh	
1	<i>dwLength</i>	4		此消息的 <i>abData</i> 数据域的大小
5	<i>bSlot</i>	1		标识命令的插槽号
6	<i>bSeq</i>	1		命令的序号
7	<i>bBWI</i>	1		用于为当前传输延长 CCID 块的超时等待时间。“该数值乘以块等待时间”的时间段过去后，CCID 将超时该块。
8	<i>wLevelParameter</i>	2	0000h	RFU (TPDU 交换级别)
10	<i>abData</i>	字节型数组		发送给 CCID 的数据块。信息是“按原样”发送至 ICC (TPDU 交换级别)

此消息的响应是 *RDR_to_PC_DataBlock* 消息。

7.1.1.5. PC_to_RDR_GetParameters

获取卡槽参数。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	6Ch	
1	<i>DwLength</i>	4	00000000h	此消息的额外字节的大小

偏移	字段	大小	值	说明
5	<i>BSlot</i>	1		标识命令的插槽号
6	<i>BSeq</i>	1		命令的序号
7	<i>AbRFU</i>	3		保留为将来使用

此消息的响应是 *RDR_to_PC_Parameters* 消息。

7.1.1.6. PC_to_RDR_ResetParameters

重置卡槽参数为默认值。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	6Dh	
1	<i>DwLength</i>	4	00000000h	此消息的额外字节的大小
5	<i>BSlot</i>	1		标识命令的插槽号
6	<i>BSeq</i>	1		命令的序号
7	<i>AbRFU</i>	3		保留为将来使用

此消息的响应是 *RDR_to_PC_Parameters* 消息。

7.1.1.7. PC_to_RDR_SetParameters

设置卡槽参数。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	61h	
1	<i>dwLength</i>	4		此消息的额外字节的大小
5	<i>bSlot</i>	1		标识命令的插槽号
6	<i>bSeq</i>	1		命令的序号
7	<i>bProtocolNum</i>	1		指定后面的协议数据结构。 00h = T=0 协议结构 01h = T=1 协议结构 以下值保留为将来使用 80h = 2线协议结构 81h = 3线协议结构 82h = I2C 协议结构
8	<i>abRFU</i>	2		保留为将来使用
10	<i>abProtocolDataStructure</i>	字节型数组		协议数据结构

T=0 协议的协议数据结构 (*dwLength=00000005h*)

偏移	字段	大小	值	说明
10	<i>bmFindexDindex</i>	1		B7-4 – FI – ISO/IEC 7816-3:1997 中表 7 的索引, 选择一个时钟频率转换因子。 B3-0 – DI – ISO/IEC 7816-3:1997 中表 8 的索引, 选择一个波特率转换因子
11	<i>bmTCKKST0</i>	1		B0 – 0b, B7-2 – 000000b B1 – 使用的约定 (b1=0: 正向约定; b1=1: 反向约定) 注: CCID 忽略该位。
12	<i>bGuardTimeT0</i>	1		两个字符间的额外保护时间。在通常的保护时间 (12 etu) 基础上增加 0-254 个 etu。FFh 与 00h 相同。
13	<i>bWaitingIntegerT0</i>	1		T=0 时 WI 用于定义 WWT
14	<i>bClockStop</i>	1		支持 ICC 时钟停止 00h = 不允许停止时钟 01h = 时钟信号为低时停止 02h = 时钟信号为高时停止 03h = 时钟信号为高或为低时停止

T=1 协议的协议数据结构(*dwLength=00000007h*)

偏移	字段	大小	值	说明
10	<i>bmFindexDindex</i>	1		B7-4 – FI – ISO/IEC 7816-3:1997 中表 7 的索引, 选择一个时钟频率转换因子。 B3-0 – DI – ISO/IEC 7816-3:1997 中表 8 的索引, 选择一个波特率转换因子
11	<i>BmTCKKST1</i>	1		B7-2 – 000100b B0 – 校验和的类型 (b0=0: LRC; b0=1: CRC) B1 – 使用的约定 (b1=0: 正向约定; b1=1: 反向约定) 注: CCID 忽略该位。
12	<i>BGuardTimeT1</i>	1		额外保护时间 (两个字符间为 0-254 个 etu) 若值为 FFh, 则保护时间减少 1 个 etu。
13	<i>BwaitingIntegerT1</i>	1		B7-4 = BWI 值 0-9 有效 B3-0 = CWI 值 0-Fh 有效



偏移	字段	大小	值	说明
14	<i>bClockStop</i>	1		支持 ICC 时钟停止 00h = 不允许停止时钟 01h = 时钟信号为低时停止 02h = 时钟信号为高时停止 03h = 时钟信号为高或为低时停止
15	<i>bIFSC</i>	1		商定的 IFSC 的大小
16	<i>bNadValue</i>	1	00h	只支持 NAD = 00h

此消息的响应是 *RDR_to_PC_Parameters* 消息。

7.1.2. CCID Bulk-IN 消息

Bulk-IN 消息用于对 *Bulk-OUT* 消息做出响应。ACR101 (CCID) 应遵循 CCID 协议第 4 部分定义的 CCID 类 *Bulk-IN* 消息。此节列举了 ACR101 (CCID) 支持的 CCID 类 *Bulk-IN* 消息。

7.1.2.1. RDR_to_PC_DataBlock

此消息由智能卡读写器发出，是对 *PC_to_RDR_IccPowerOn*、*PC_to_RDR_XfrBlock* 和 *PC_to_RDR_Secure* 消息的响应。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	80h	表示 CCID 正在发送一个数据块。
1	<i>dwLength</i>	4		此消息的额外字节的大小
5	<i>bSlot</i>	1		与 <i>Bulk-OUT</i> 消息中的值相同
6	<i>bSeq</i>	1		与 <i>Bulk-OUT</i> 消息中的值相同
7	<i>bStatus</i>	1		CCID 规范 4.2.1 节定义的插槽状态寄存器
8	<i>bError</i>	1		CCID 规范 4.2.1 节和本规范 5.2.8 节定义的插槽错误寄存器
9	<i>bChainParameter</i>	1	00h	RFU (TPDU 交换级别)
10	<i>abData</i>	字节 型数 组		本数据域包含由 CCID 返还的数据

7.1.2.2. RDR_to_PC_SlotStatus

此消息由智能卡读写器发出，是对 *PC_to_RDR_IccPowerOff*、*PC_to_RDR_GetSlotStatus* 和 *PC_to_RDR_Abort* 消息，以及类特定 ABORT 请求的响应。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	81h	
1	<i>dwLength</i>	4	00000000h	此消息的额外字节的大小
5	<i>bSlot</i>	1		与 <i>Bulk-OUT</i> 消息中的值相同
6	<i>bSeq</i>	1		与 <i>Bulk-OUT</i> 消息中的值相同
7	<i>bStatus</i>	1		CCID 规范 4.2.1 节定义的插槽状态寄存器
8	<i>bError</i>	1		CCID 规范 4.2.1 节和本规范 5.2.8 节定义的插槽错误寄存器
9	<i>bClockStatus</i>	1		值= 00h 时钟运行 01h 时钟停于 L 状态 02h 时钟停于 H 状态 03h 时钟停止于未知状态 所有其他值保留为将来使用。



7.1.2.3. RDR_to_PC_Parameters

此消息由智能卡读写器发出，是对 *PC_to_RDR_GetParameters*、*PC_to_RDR_ResetParameters* 和 *PC_to_RDR_SetParameters* 消息的响应。

偏移	字段	大小	值	说明
0	<i>bMessageType</i>	1	82h	
1	<i>dwLength</i>	4		此消息的额外字节的大小
5	<i>bSlot</i>	1		与 Bulk-OUT 消息中的值相同
6	<i>bSeq</i>	1		与 Bulk-OUT 消息中的值相同
7	<i>bStatus</i>	1		CCID 规范 4.2.1 节定义的插槽状态寄存器
8	<i>bError</i>	1		CCID 规范 4.2.1 节和本规范 5.2.8 节定义的插槽错误寄存器
9	<i>bProtocolNum</i>	1		指定后面的协议数据结构。 00h = T=0 协议的结构 01h = T=1 协议的结构 以下值保留为将来使用 80h = 2 线协议结构 81h = 3 线协议结构 82h = I2C 协议结构
10	<i>abProtocolDataStructure</i>	字节 型数 组		协议数据结构如 5.2.3 节汇总



7.1.3. 其它通过 PC_to_RDR_XfrBlock 进行的命令

7.1.3.1. GET_READER_INFORMATION

此命令返回特定智能卡读写器型号的相关信息和当前运行状态，例如固件版本号、命令及其响应的最大数据长度。支持的卡片类型、是否有卡片插入以及卡片是否上电。

注：只有使用 SCardConnect() API 建立逻辑智能卡读写器通信之后才可以使用此命令。对于 SCardConnect() API 的详细说明参见 PC/SC 规范。

Pseudo-APDU				
CLA	INS	P1	P2	Lc
FFh	09h	00h	00h	10h

命令格式 (PC_to_RDR_XfrBlock 中的 abData 数据域)

FIRMWARE										MAX_C	MAX_R	C_TYPE	C_SEL	C_STAT	

响应数据格式 (RDR_to_PC_DataBlock 中的 abData 数据域)

- FIRMWARE** 10 字节的固件版本号
- MAX_C** 命令数据字节的最大个数。
- MAX_R** 能请求响应传输的最大字节数。
- C_TYPE** 该款智能卡读写器支持的卡片类型。该数据域是一个位图，其中每一位表示一个特定的卡片类型。位设置为‘1’表示相应卡片类型是读写器支持的，并且能够用 SELECT_CARD_TYPE 命令选中。位分配如下：

字节	1										2																				
卡片类型	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0															

Appendix A.1 描述了位与卡片类型的对应关系。

- C_SEL** 当前选定的卡片类型。00h 表示未选择卡片类型。
- C_STAT** 标识卡片是否已经插入读写器，以及卡片是否已经上电：
 - 00h: 没有插入卡片
 - 01h: 插入卡片，但是没有上电
 - 03h: 卡片已上电



7.2. 大容量存储

大容量存储设备类标准定义了主机（计算机）和存储设备间进行数据传输所需的所有协议。ACR101 (CCID) 智能卡接口的 USB 端点配置和使用应当符合 USB 大容量存储设备规范第 3 部分（协议代码）中关于海量存储设备类 Bulk-Only 传输协议的规定。访问以下网站可获取该文档：www.usb.org.

该规范概述总结如下：

1. **控制命令**通过控制通道（缺省通道）发送,与 CCID 接口共享。
2. **数据输出**命令协议使用 *BULK-OUT* 端点传输从主机到设备的数据。
3. **数据输入**命令协议使用 *BULK-IN* 端点传输来自设备的数据或返回设备状态。



附录A. 支持的卡片类型

下表列出了 `GET_READER_INFORMATION` 命令返回卡片类型代码相应的卡片类型：

卡片类型代码	卡片类型
00h	自动选择 T=0 或 T=1 通信协议
01h	I2C 存储卡 (1、2、4、8 和 16 kilobits)
02h	I2C 存储卡 (32、64、128、256、512 和 1024 kilobits)
03h	Atmel® AT88SC153 安全存储卡
04h	Atmel® AT88SC1608 安全存储卡
05h	Infineon® SLE4418 和 SLE4428
06h	Infineon® SLE4432 和 SLE4442
07h	Infineon® SLE4406、SLE4436 和 SLE5536
08h	Infineon SLE4404
09h	Atmel® AT88SC101、AT88SC102 和 AT88SC1003
0Ch	使用 T=0 通信协议的 MCU 卡
0Dh	使用 T=1 通信协议的 MCU 卡



附录B. 响应报文状态码

下表列出了 ACR38 可能返回的错误代码：

错误代码	状态
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

Android 是 Google Inc.的商标。

Atmel 是 Atmel 公司或其子公司在美国及其他国家的注册商标。

Infineon 是英飞凌科技公司的注册商标。

Linux®是林纳斯·托瓦兹（Linus Torvalds）在美国和其他国家的注册商标。

Microsoft 是 Microsoft Corporation 在美国及/或其他国家的注册商标。

MIFARE 和 MIFARE DESFire 是 NXP B.V.的注册商标，由 NXP B.V.授权许可后才可使用。