

pcProx[®] Enroll & Wiegand Converter

Configuration Utility User Manual



Thank You!

Congratulations on the purchase of your pcProx® 125 kHz Enroll, pcProx® 13.56 MHz Enroll, or Wiegand device(s). RF IDEas hopes you enjoy using the readers as much as we enjoyed creating and developing them. Configuration is easy, so you will be able to quickly take advantage of a more secure environment in your business, school, or organization.

Please call our Sales department if you have any questions or are interested in our OEM and Independent Developer's programs.

We look forward to your comments and suggestions for our product line! Please go to www.RFIDEas.com and follow the **Support** ⇨ **Learning Center** link for more details about our product line.

We are always discovering new applications for our product line(s). There are several software developer's licensing our technology so the solution you are looking for may already be developed.

Thank you,
The RF IDEas Staff

Need Assistance?

Ph: 847.870.1723
Fx: 847.483.1129
E: Sales@RFIDEas.com
TechSupport@RFIDEas.com

Contents

2	Thank You!	13	Chapter 3: Software
4	Chapter 1: The Basics	13	pcProx Software Installation
4	Wireless Identification Overview	15	pcProx Configuration Utility
5	ID Card Reader System	15	Connect Tab
5	pcProx Output Formats	17	Timing Tab
7	Credential Form Factors	18	Set Keystroke Data Tab
8	Card Compatibility	21	Advanced Tab
8	Reader Configuration Purposes	28	Card Formats Tab
9	Chapter 2: Hardware	28	About Tab
9	What's In Your Part Number?	29	HWG Files for Device Deployment
10	Interface (Connectors)	30	Open a .HWG File
11	USB Readers and Wiegand Converters	32	Chapter 4: Tips and Troubleshooting
11	RS-232 Readers and Converters	32	Troubleshooting
11	Minimum System Requirements	32	Precautions
12	Reader Set-Up Basics	33	Before You Call Technical Support
12	LED Beeper	33	Talking To The Technician
		34	Appendix
		54	Index
		55	Other Products and Accessories

Wireless Identification Overview

pcProx® Activated Identification

Employers are more security conscious than ever. More buildings, machines, systems, and applications require identification information to gain access. RF IDEas devices allow the building access cards to be used as a digital identifier throughout the workplace.

Various pcProx applications include:

- Card Enrollment
- Application log-on
- Form filler to existing software applications
- PC/LAN Log On
- Cafeteria Purchases/Vending
- Machine Access
- PLC and embedded controllers
- Time/Attendance

While magnetic stripe technology is very inexpensive, it is prone to read problems from wear and dirt. Cards have a limited life as they scratch easily. Readers pick up dirt and oils that damage the read head.

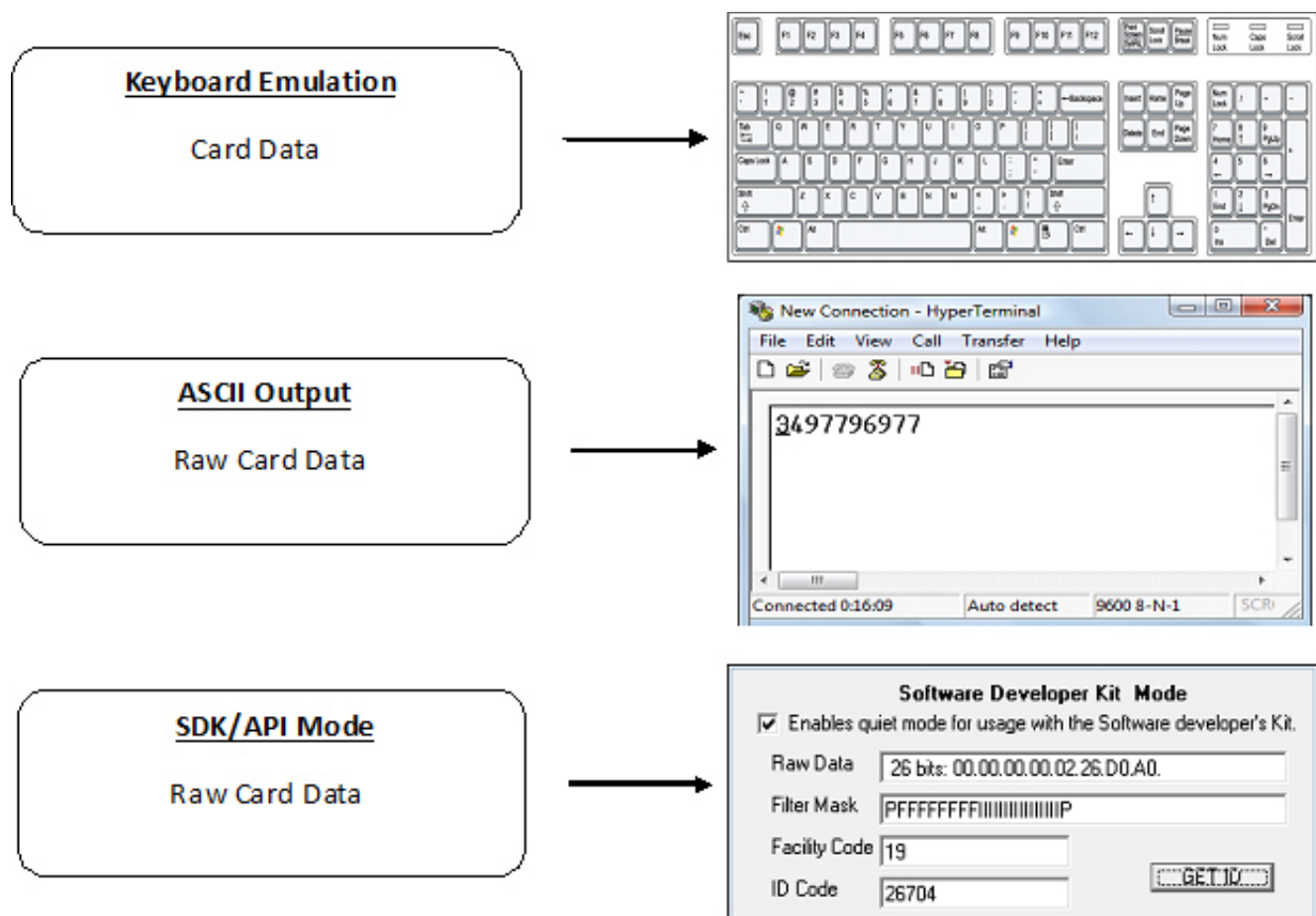
Proximity technology is based on a contactless interface and is not subject to those reliability issues. Our proximity device is easily configured to increase security and reliability. Companies using proximity technology for building access immediately benefit as their employee identification cards can also be used with the proximity device for additional authentication applications. Thus, the majority of deployment and enrollment costs are quickly recovered.

The diagram on the following page is a high level overview of how the reader works. The reader sends RF signals to the card and the card sends signals back to send data. The card data is output by the reader in keystrokes or ASCII characters. This card data can be configured to include delimiters to separate the data. This reader can be used as a standalone system or seamlessly integrated with other software applications using the optional Software Developer's Kit (SDK).

ID Card Reader System



Output Formats



pcProx Features

- Read all data from proximity cards
- Read configuration
- Write configurations
- Software Developer's Kit/API compatibility
- Output in decimal or hexadecimal
- User controls number of digits output

pcProx Functions

- Software Developer's Kit (SDK) USB
- Software Developer's Kit RS-232
- USB Keyboard
- RS-232
- Serial Virtual COM

pcProx Connectors

- USB Keyboard
- USB Virtual COM Port
- RJ45
- PS/2
- DB-9 RS-232
 - Power Plug - 2.1 mm
 - Power Plug - 2.5 mm
 - PS/2
 - DB9 - Pin 9 Power

Credential Form Factors

Credentials are inactive electronic devices that rely on readers to supply the required power for start-up and communication. The credential itself, consists of antennas that produce proximity or contactless frequencies. Proximity and contactless smart card technology cards allow users to effortlessly manage multiple applications through a single credential.

Data: The data on access cards are a string of binary numbers set with a fixed configuration and length.

Frequencies: RF IDEas' access control readers and credentials utilize the low-frequency 125 kHz (proximity) band and/or the high-frequency 13.56 MHz (contactless) band.

Credential Form Factors: With over 300 million physical access credentials in use worldwide, there are a variety of low and high frequency form factors customers can choose from to meet their particular needs.

The below illustrates some of the various form factors available.



CSN: Also known as the Card Serial Number, is part of the ISO 15693 standard for vicinity cards operating at the 13.56 MHz frequency.

UID: The User ID or User Identification, can be encoded as data on the card when a security key is needed.

Manufacturer/Vendor Card Compatibility

Please go to www.RFIDeas.com for specific device part numbers associated to card types.

Reader Configuration Purposes

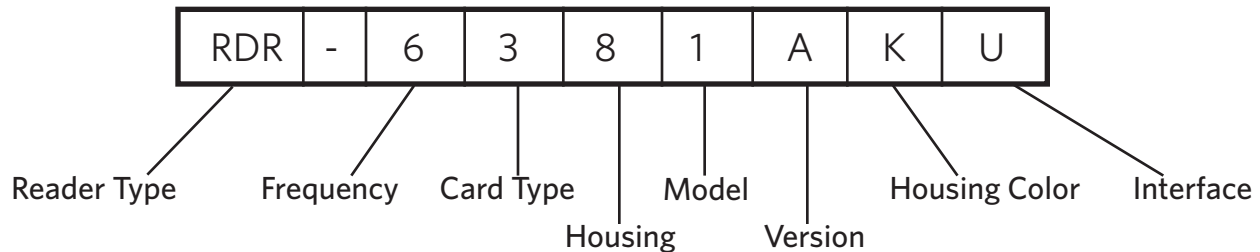
The method of encoding data on a card and transmitting data to the reader differs accordingly to each technology involved. The reader itself is not aware of the makeup of the card data format or access privileges for the cardholder. This information is only accessible through the configuration process of the reader utilizing the supplied software.

The reader is very flexible and may need to be configured in order to present an exact desired output for the user, such as, singling out FAC or ID, obtaining a desired base (i.e. decimal, lowercase, uppercase, hexadecimal).

What's In Your Part Number?

All RF IDEas reader part numbers follow a distinct system of categorization to allow for an ease of differentiation between products.

Below is the basic part number scheme.



Reader Type: The reader type distinguishes between standard reader, OEM, converter, mag-stripe, or a kit.

Frequency: RF IDEas' access control readers are available in low-frequency 125 kHz (proximity) or high-frequency 13.56 MHz (contactless).

Card Type: The card type allows for the selection of over 35 different card types for reader compatibility (Please visit www.RFIDEas.com, choose a product and locate the Part Numbers tab for specific device part numbers associated to card types).

Housing: This option provides the user to select the form factor housing for the desired reader. The housings include; desktop, wall mount, USB dongle, PCMCIA, bare board, ExpressCard, or custom. (For more on form factors, please visit www.RFIDEas.com)

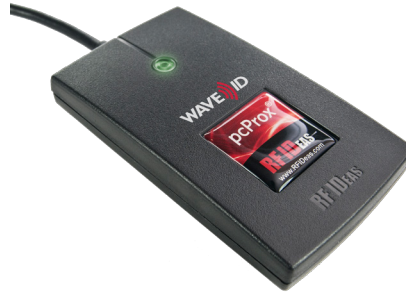
Model: The model selection corresponds to the type of reader, whether it is a standard, writer (13.56 MHz contactless only), playback (13.56 MHz contactless only), SDK, or analyzer.

Version: The version refers to the selection of either our standard or custom build.

Housing Color: The color category simply allows for the selection of either our black or pearl housings. (Applies only to desktop and wall mount housings)

Interface: This option specifies the type of connection for the reader (i.e. USB, RS-232, PCMCIA, etc).

Interface (Connectors)



CONNECTORS



USB
Keyboard



USB
Virtual COM Port



DB-9
RS-232



Female Power



RJ45



PS/2 Power

OUTPUT



or

RS-232 Data
0.012345679012

USB Readers and Wiegand Converters

The pcProx USB keystroke device operates in two primary modes:

- USB keyboard. It reads the card data and sends it as keystrokes as if the user typed the ID data on a keyboard.
- Under the application programmer interface (API) defined in the pcProx SDK. When it reads card data, the active application receives the entire card data.

RS-232 Readers and Converters

The RS-232, Ethernet, or virtual COM port device operates in two primary modes:

1. ASCII output device. In this mode the user card data is read and sent as a decimal or hexadecimal number in ASCII characters.
2. API defined in the pcProx SDK. The device attaches to a computer serial port. When it reads card data, the active application receives the entire card data.

Once the configuration settings are correctly configured and written to flash memory, the device can immediately be deployed.

Minimum System Requirements

Minimum System Requirements	
HARDWARE	Pentium class PC
MEMORY	32 MB RAM
DISK	25 MB hard disk space
I/O	1 available RS-232 or USB Port
OPERATING SYSTEM	Any operating system that supports a USB keyboard including Microsoft Windows 2000®, XP®, Vista®, 7®, Server 2003®, Server 2008®, Linux. Can be used for keystroke applications.

Note: The software does not perform any data validation checking. The data must be known before it is read to verify its validity.

Reader Set-Up Basics

Plug the connector into the workstation's (or available on any peripheral) open RS-232, USB or Ethernet plug.

Place the device next to the monitor, beside the workstation, or where appropriate.

The workstation should detect new hardware for USB connections. Verify the workstation recognizes this connection using Device Manager'.

Verify the correct COM port for RS-232 DB9 connections using 'Device Manager.'

When the software is installed, it should recognize these connections in order to configure the appropriate device. Once the device is configured and written to its flash memory, these settings will not have to be configured again.

LED Beeper

The desktop, USB dongle, wall mount, and bare board (OEM) model readers are all equipped with a light up LED on the front cover. The LED is configurable through the utility software (See LED and Beeper functions in the Software section) to allow the device to produce a beep upon light up of the LED when a credential is detected by the reader.

pcProx Software Installation

Install the pcProx installation program to the Desktop for quick access. This installer is digitally signed by RF IDEas to authenticate file integrity for your safety.

Open the www.RFIDEas.com website and click **Support** ⇒ **Software**.

Click **pcProxConfig.exe** to download the installation utility. This file is the pcProx installation program. Use this icon on the desktop to open the installation program to configure the device once the software is installed.

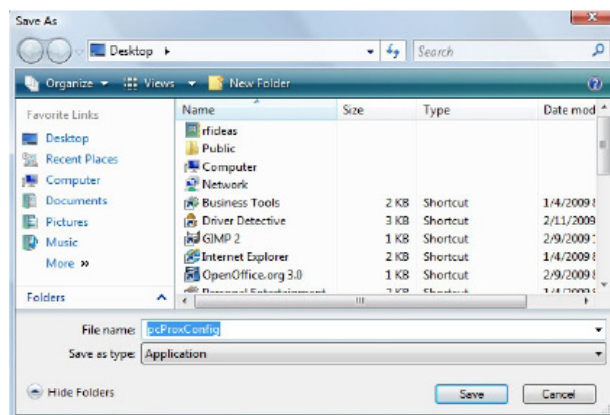
The File Download – Security Warning window displays.



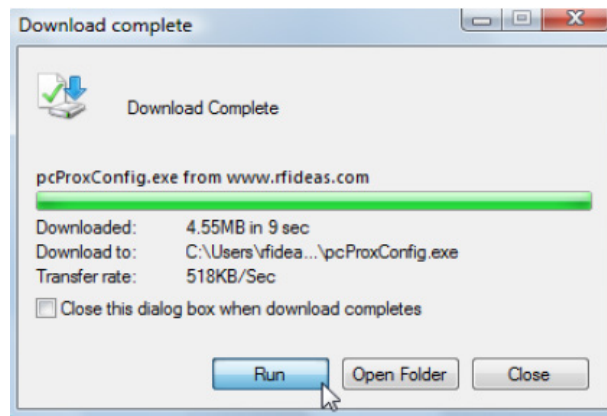
Click **Save** in the File Download – Security Warning window to save the installation program to the Desktop.

Note: Close the browser as the configuration software can be run from the location it was saved.

Click **Save** in the Save As window to accept the default file name

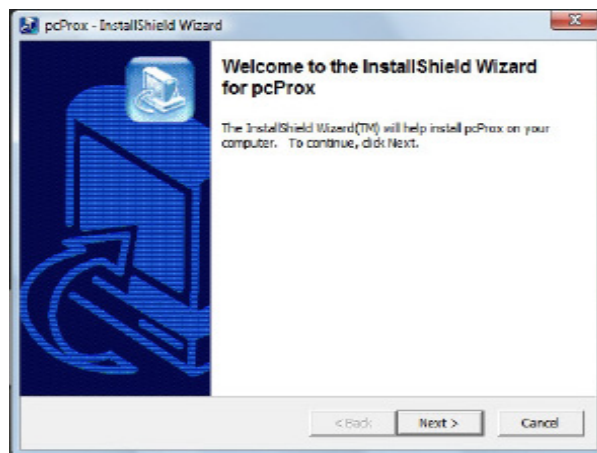


The configuration software downloads to the appropriate location.

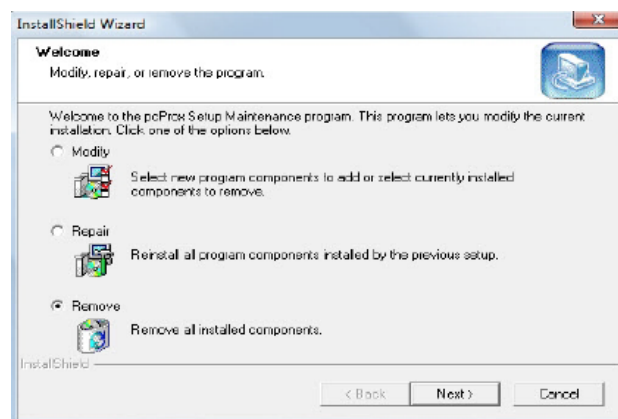


Click **Run** in the Download Complete window to open the configuration software and begin configuring the device.

Follow the prompts to install the configuration software.

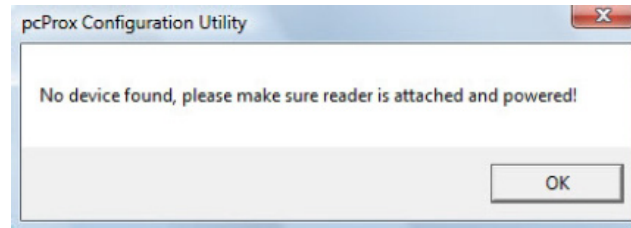


If you are installing an upgrade, click the **Remove** radio button to remove the old configuration software before installing the new files.



Note: Use **Control Panel** ⇒ **Add/Remove** Programs to successfully remove the setup program if there is a problem with installation or if this is an upgrade. Reboot and then reinstall the program.

If the pcProx utility is open and a device is not recognized or connected to the workstation, the following message displays:



There are two utilities to configure the device. The following section describes the pcProx configuration utility just installed.

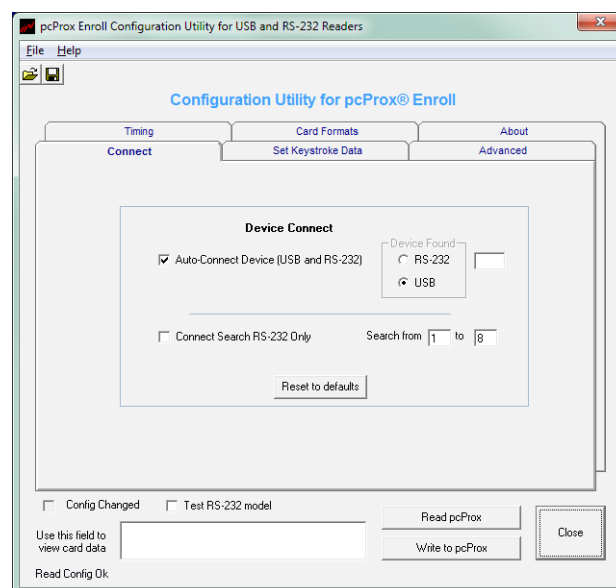
To proceed, you must first verify that the device is connected to the appropriate connector. It is best to configure one device at a time, plug in each device separately so that you can easily distinguish which device is being configured.

Once connections have been verified, the device can then be configured to output the card data in the appropriate format. This configuration can be saved to quickly configure additional devices.

pcProx Configuration Utility

Connect Tab: Device Connect

Connect the configuration utility to the device's internal storage to allow reading, modifying, and writing to the device.

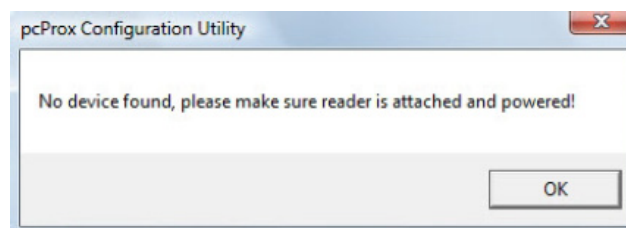


Note: Use the helpful information that displays when the cursor rests on a field or button.

Function	Description
Auto-Connect Device (USB and RS-232)	The utility auto searches the USB and RS-232 (1 to 8) ports for the connected device. Ports 1 to 8 is the default. The last ports entered display if the defaults were changed and the configuration written to flash memory.
Device Found	The radio button identifies the type of device found (USB or RS-232). The configuration program automatically checks for a USB device first. If it does not find one, it scans COM ports 1 to 8 for a serial port device. If the port numbers are changed and the configuration is written to flash memory, a ComSpecPort.txt file is saved to the computer.
Connect Search RS-232 Only (Serial)	Check to select a specific group of COM ports, or a specific COM port, for a serial, Ethernet, or CDC Ethernet device from RF IDEas.
Search from..to	Enter the appropriate ports to connect to a specific device.
Reset to Defaults	Click to reset the configuration settings back to the factory defaults. The factory settings are different for each device model.
Config Changed	The program selects this check box to indicate to the user any modifications made to the configuration which have not been written to the device.
Test RS-232 model	Check to view ASCII card data. Select this check box to disable the configuration from writing to flash memory. Deselect this check box to continue to configure the device. This only works for COM ports 1 to 8.
Read pcProx 125 kHz or pcProx 13.56 MHz	Click to read the current device configuration.
Write to pcProx 125 kHz or pcProx 13.56 MHz	Click to write the new configuration settings to flash memory.

Note: Use the test area to view card data if Notepad or Hyper Terminal is not available.

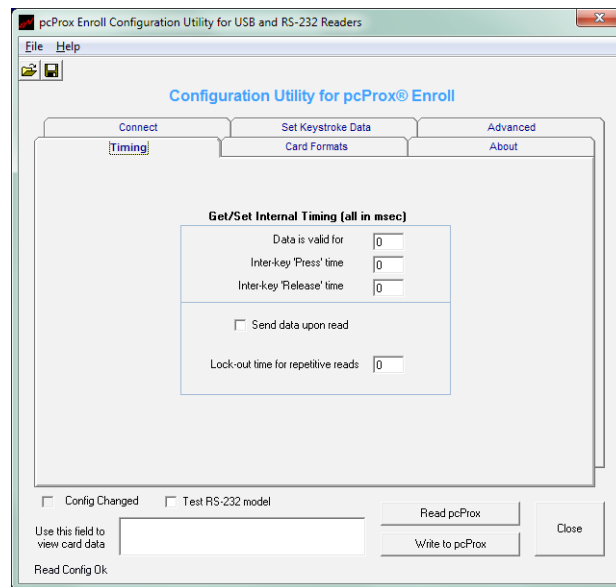
If the installation utility does not recognize a device as being connected to the workstation, the following error message displays:



To proceed, click **OK** to allow the utility to open. Check the appropriate **Device Connect** check box and connect.

Timing Tab: Get/Set Internal Timing (in msec)

Control the timing for the card data on the device here. You can regulate how quickly the next card data is read. The utility reads the current device configuration. Remember, the device is sending data as if it was typing on a keyboard.



Function	Description
Data is Valid For	Enter the time the card data remains valid in the device. The minimum value is 900. This is read in 50 msec increments. The default is 1000.
Inter-key 'Press' Time	Enter the length of time the key is held down. The minimum value is 4. The maximum is 1,000. The default is 20.
Inter-key 'Release' Time	Enter the time delay between keystrokes. The minimum value is 4. The maximum is 1,000. The default is 20.
Send Data Upon Read	If this is checked, the card data is repeatedly sent. Generally this check box is not checked. If left unchecked, the card data is only read once. Otherwise, the card data is continuously sent.
Lock-out Time for Repetitive Reads	This sets how long the device is locked and will not accept the data of the next card. This is read in 50 msec increments. The minimum value is 0. The maximum is 12,500.

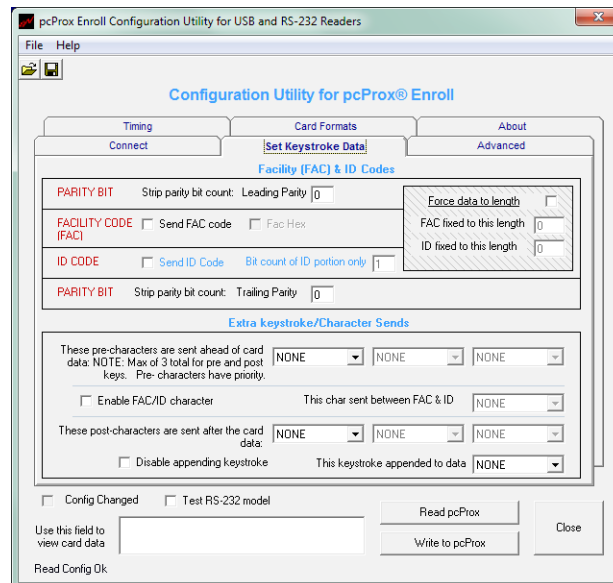
The below figure displays an example of these fields with values.

This figure shows a close-up of the 'Get/Set Internal Timing (all in msec)' section with the following values entered: 'Data is valid for' is 1000, 'Inter-key 'Press' time' is 20, 'Inter-key 'Release' time' is 20, the 'Send data upon read' checkbox is unchecked, and 'Lock-out time for repetitive reads' is 1200.

Set Keystroke Data Tab: Facility (FAC) & ID Codes

Enter the FAC and Card ID data here. The top section is organized consistent with the way the card data is laid out on generic proximity cards.

Note: If **Enables quiet mode for usage with the Software Developer's Kit** is checked on the Advanced tab, only the Parity Bit fields are enabled on this tab. If the SDK is being used, all keyboard related functions do not apply as keystroking is disabled.



Function	Description
Strip Parity Bit Count - Leading Parity	Enter the number of Leading Parity bit(s) to ignore. There are typically only 1 or 2 leading parity bits. The minimum is 0 and the maximum is 15.
Send FAC Code	Check to send the Facility/Site (FAC) code.

Note: If neither the **Send FAC** and **Send ID** are checked, all bits are converted to a single decimal number.

Function	Description
FAC Hex	Check to send this code in hexadecimal. This is set for KANTECH 10 proximity cards.
Send ID Code	Check to send only the ID portion of the card data. Check both the 'Send FAC Code' and 'Send ID Code' check boxes to send both data fields.
Bit Count of ID Portion Only	Enter the bit count of the ID portion.
Strip Trailing Parity Bit Count	Enter the number of trailing parity bit(s) to ignore. There is typically only 1 trailing parity bit. The minimum is 0 and the maximum is 15.

The following displays these fields with assigned values for a 26 bit format.

PARITY BIT	Strip parity bit count: Leading Parity	<input type="text" value="1"/>
FACILITY CODE (FAC)	<input checked="" type="checkbox"/> Send FAC code <input type="checkbox"/> Fac Hex	
ID CODE	<input checked="" type="checkbox"/> Send ID Code Bit count of ID portion only	<input type="text" value="16"/>
PARITY BIT	Strip parity bit count: Trailing Parity	<input type="text" value="1"/>

Function	Description
Force Data to Length	Check to enable the addition of zeros to the front of both FAC and card data to create a specific length. For example, if the data is 567 and the length is set to 6, the output is 000567.
FAC Fixed to this Length	Enter the number of zeros to add to the front the FAC data to create a specific length.
ID Fixed to this Length	Enter the number of zeros to add to the front of the card data to create a specific length.

Force Data to Length

The following figure displays these fields.

Force data to length	<input checked="" type="checkbox"/>
FAC fixed to this length	<input type="text" value="3"/>
ID fixed to this length	<input type="text" value="5"/>

If	Then
Neither Send FAC code or Send ID code is checked...	Both the FAC and ID bits are represented by one number
Send FAC Only is checked...	Only the FAC code bits represent the number sent
Send ID Only is checked...	Only the ID code bits represent the number sent
Both Send FAC code and Send ID code are checked...	The ID code bits are sent as one number The FAC code bits are sent as a different number

Pre and Post Keystrokes and Delimiters

Configure the device to send extra data in addition to the card data here. Keystrokes are sent for USB readers. ASCII characters are sent for RS-232 readers. Due to the device's internal limits, a maximum of 3 keystrokes/characters shared between leading or trailing keystrokes can be set. If a reader with more than 3 keystrokes/characters is required, please call our Sales Department.

For example:

1. Add 1 keystroke before the card data

OR

Add 1 keystroke between the card data and facility code

2. Add 2 trailing keystrokes.

The first field defines up to 3 pre-keystrokes or characters sent prior to retrieving any card data. Select the keystroke/characters from the drop-down list. There are 3 special characters (special1, special2, special3) that can be specified. These add an extra measure of security and make it more difficult to reproduce card data directly from the keyboard. This is useful when using the proximity device as a log on device.

Note: The pre and post keystroke drop-down menu list displays the keyboard characters for all USB devices. The drop-down list displays ASCII characters for serial devices.

Function	Description
Enable FAC/ID Character	Check to use a separator character between the FAC and card data.
This Char Sent Between FAC & ID	Select the delimiter character to place between the FAC and ID codes.

The next section of the Set Keystroke Data tab controls the 3 post-keystrokes or characters to be sent after the card data that is read. Three characters can be sent as post delimiters if none Select the keystroke/characters from the drop-down list.

Function	Description
Disable Appending Keystroke	Check so the device does not append a character to the end of the data.
This Keystroke Appended to Data	If the Disable appending keystroke check box is empty, the character selected from the drop-down list is appended to the card data.

The following figure displays these fields.

Extra keystroke/Character Sends

These pre-characters are sent ahead of card data: NOTE: Max of 3 total for pre and post keys. Pre-characters have priority.

Pre-character 1: a | NONE | NONE

☒ Enable FAC/ID character This char sent between FAC & ID: COLON

These post-characters are sent after the card data:

Post-character 1: z | NONE | NONE

☐ Disable appending keystroke This keystroke appended to data: ENTER

Note: The trailing characters are only available on Firmware 5.0.0 and above. These fields will not be enabled on any Firmware below 5.0.0.

Go to www.ASCII.org to view the pre and post characters that display for serial readers.

Advanced Tab

Use this tab to control the advanced features of the device.

pcProx Enroll Configuration Utility for USB and RS-232 Readers

File Help

Configuration Utility for pcProx® Enroll

Timing | Card Formats | About

Connect | Set Keystroke Data | **Advanced**

LED - Beep Control

☐ SDK Controls LED ☐ Beep

☐ Red ☐ Green

USB Reader(s) on this Machine 1

Enter LUID 0-32767: 9329

List of unique USB reader addresses: 8080

Software Developer Kit Mode

☐ Enables quiet mode for usage with the Software developer's Kit.

Raw Data: []

GET ID

Characters Sent When Card is Removed

First character: []

Second character: []

☐ Reverse Wiegand Bits ☐ Read only cards with this bit count: 26 ☐ Euro KeyPad

☐ Enable output as Hexadecimal ☐ Reverse Bytes ☐ Emulate ProxPro

☐ Invert Wiegand Data (pcProxH only) ☐ Ignore H/W data inversion override ☐ Enable 64 bit math

☐ Config Changed ☐ Test RS-232 model

Use this field to view card data: []

Read Config OK

Read pcProx Write to pcProx Close

Note: If **Enables quiet mode for usage with the Software Developer's Kit** is checked, only the Parity Bit fields are enabled on the Set Keystroke Data tab. If the SDK is being used, all keyboard/character related fields do not apply.

LED - Beep Control

Function	Description
SDK Controls LED	Check to assign LED control to the device. If unchecked, the device controls the LED. Please see the following tables.
Beep	Check so the device beeps when a card data is read. If unchecked, the device will not beep.

Function	Description
Red	Check to display a red LED.
Green	Check to display a green LED.

The following table further explains controlling the LED.

If	Then
SDK Controls LED is checked Red is unchecked Green is unchecked	The LED is off.
SDK is checked Red is checked Green is unchecked	The LED displays Red
SDK is checked Red is unchecked Green is checked	The LED displays Green
SDK is checked Red is checked Green is checked	The LED displays Amber
If unchecked, the device controls the LED	Red = Device Idle Green = Valid Card Data

The following figure displays these fields.

LED - Beep Control

☐ SDK Controls LED ☒ Beep

☐ Red

☐ Green

USB Device(s) on this Machine

Note: This section does not apply to the RS-232 readers since these are attached to individual COM ports.

This field displays the number of readers connected to the workstation. Up to 16 different pcProx Enroll USB readers can be connected to the on their workstation at one time.

Determine the card data that various readers display by configuring a leading character to insert in front of the card data.

Attach one device at a time in order to preset each specific device's leading characters.

Function	Description
Enter LUID 0 - 32767	<p>Enter the logical unit ID of the device presently connected. This allows the developer to talk to each individual device to determine which one is reading the card data.</p> <p>Once a unique address is written to the device, it will selectively communicate with each device to determine which one sent the card data.</p> <p>SDK users enter the address for the attached device here. Only 1 device may be attached when this feature is used.</p>
List of Unique USB Reader Addresses	<p>This drop down list displays all attached readers that have a unique address. If there are more readers on the workstation than USB ports, a unique 2 byte number between 0 and 65,535 can be assigned to the reader to identify the device. The logical unique ID (LUID) can be set by the user.</p>

The following figure displays these fields.

The screenshot shows a software interface titled "USB Reader(s) on this Machine". It contains three input fields: a small box with the number "1", a text box labeled "Enter LUID 0-32767" containing the value "0", and a dropdown menu labeled "List of unique USB reader addresses" also containing the value "0".

Software Developer Kit Mode

Software developer's kit (SDK) is a software application interface that assists software developers, authors and IT professionals to integrate card data to a physical device or database/software program. The SDK enables cards data to be read and delivered via your software to a device or program that it can then use to perform a specific function. Our SDK is leading edge and works for almost every type of manufacturers credentials for more flexibility when marketing your product.

If you are a software developer and have purchased an SDK, you can change the mode of operation of the proximity device. This does not apply to the 82 Series readers. This means from 'keyboard' to SDK for the USB device. This means from direct ASCII output asynchronously to the SDK's synchronous mode of polling for the RS-232 device.

In the SDK mode, all keystroke or serial send data can be inhibited. The card data can be read using functions included in RF IDEas' SDK. The SDK is sold separately.

SDK tightly integrates your software with the card reader. Please visit www.RFIDEas.com for more details.

Function	Description
Enables Quiet Mode for Usage with the Software Developer's Kit	If this check box is checked, the device will not send card data as USB keystrokes or asynchronous ASCII.
Raw Data	This field displays the number of bits read as well as the card data from the card less the parity bits in hexadecimal format.
Filter Mask	If the Filter Mask field is empty, the device tries to 'guess' the card format. There are literally thousands of formats and this is just a guess - unless the format, facility and ID codes are known. You can also enter your own mask using a string of characters made up of 'F', 'I', and 'P'.
Facility Code	This represents the decimal equivalent of the bits defined by 'F' in the Filter Mask.
ID Code	This represents the decimal equivalent of the bits defined by 'I' in the Filter Mask.

The following figure displays these fields.

Software Developer Kit Mode

☒ Enables quiet mode for usage with the Software developer's Kit.

Raw Data

Filter Mask

Facility Code

ID Code

GET ID

Use GET ID to determine the card format.

Note: Present the card on the device and click GET ID at the **same** time for the results to display

Any stripped parity bits do not display using the GET ID function. For example, if there is 1 leading and 1 trailing parity bit set to be stripped for a 26 bit card on the Set Keystroke Data tab, the GET ID returns 24 bits.

The button calls an SDK function and shows how the application would use and see data using SDK.

For example: The filter for a 26 bit Wiegand card format is: "PFFFFFFFFFFFFFFFFFFFFF". This has:

- 2 "P" - parity bits,
- 8 "F" - FAC code bits, and
- 16 "I" - card code bits.

This will split the bit-stream into the FAC and card data codes.

Click **GET ID** immediately after the card is read or the following message displays;

Software Developer Kit Mode

☒ Enables quiet mode for usage with the Software developer's Kit.

Raw Data

Filter Mask

Facility Code

ID Code

Characters Sent When Card is Removed

The RS-232 device is capable of sending two 'Card Gone' characters whenever the card is removed. Use the pull-down character selector boxes to choose the desired ASCII characters. These characters will be sent once the data in the entry field has elapsed. Please see the Timing Tab.

Function	Description
Reverse Wiegand Bits	Check to reverse the order of Wiegand bits. This is primarily used for Card Key proximity cards.
Enable Output as Hexadecimal	Check to select a hexadecimal keystroke data output instead of the standard decimal.
Invert Wiegand Data (pcProxH only)	Check if using a legacy application that requires the Wiegand data to be inverted.

Wiegand Formatting

The following figure displays these fields.

☐ Reverse Wiegand Bits

☐ Enable output as Hexadecimal

☒ Invert Wiegand Data (pcProxH only)

Function	Description
Read Only Cards with this Bit Count	No data is sent from the device unless the bit count is matched. The total bits received from the card must match this bit count, parity bits included. If checked and 26 is entered in the field, the device will only respond to 26 bit cards.
Reverse Bytes	Check to read the card data in a reverse byte order. Use this feature with MIFARE CSN readers. For 56 bit cards, go to the 'Set Key Stroke Data' tab and set the Bit Count of ID Portion to 56.
Ignore H/W Data Inversion Override	Check to override the invert the Wiegand data lines feature. Be sure to check this unless your data is inverted.

The following figure displays these fields.

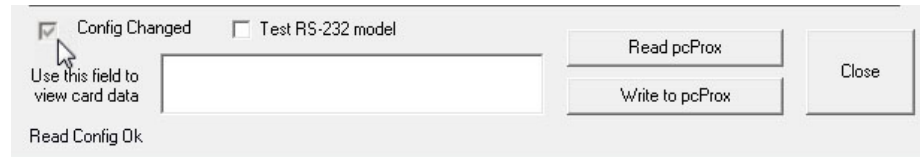
☒ Read only cards with this bit count 26
☐ Reverse Bytes
☒ Ignore H/W data inversion override

Function	Description
Euro Key Pad	Check to make the device compatible to Europe's AZERTY keyboard. The USB key codes for the numeric key pad instead of the top number row on the keyboard are enabled to overcome the AZERTY keyboard differences.
Emulate ProxPro	<p>This feature is for the pcProx RS-232 model only. The HID ProxPro device has a specific data output:</p> <p>For example the string is: CCDDDDDDDDDDDDXX<CR><LF> C = HID Specific Code D = Card Data (valid numbers are 0000000000 to 1FFFFFFFFF) X = Checksum. This is calculated by converting the ASCII characters to hex values and adding the six bytes CC..DD ..DD together and taking the least significant 8bits of the result. CR = 0D Hex (ASCII code for a carriage return). LF = 0A Hex (ASCII code for a line feed).</p> <p>Check so the RS-232 produces this output. No other selected options will have any effect until this feature is deselected and written to flash memory.</p>
Enable 64 Bit Math	<p>Check so the device uses a 64 bit binary to decimal conversion to calculate the card number. This is available for firmware version 5.6 and above. In previous versions, the device would only convert 32 bits at a time and concatenate when larger bit length ID numbers were encountered.</p> <p>Check this to display the true representation of the number or if the card is over 32 bits.</p>

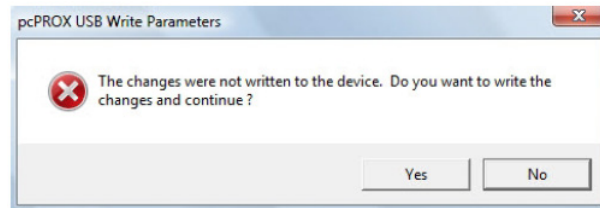
The following figure displays these fields.

☐ Euro KeyPad
☐ Emulate ProxPro
☐ Enable 64 bit math

If the Read Config OK message displays and there is a check in the Config Changed check box, the new configuration has NOT been saved. Click **Write to Prox** to write the new configuration to flash memory.



The following message displays if the new configuration has not been written/flushed to the device.



To avoid writing the changes to the device, click **NO** to close the message window. Click **X** to close the pcProx Enroll Configuration Utility window.

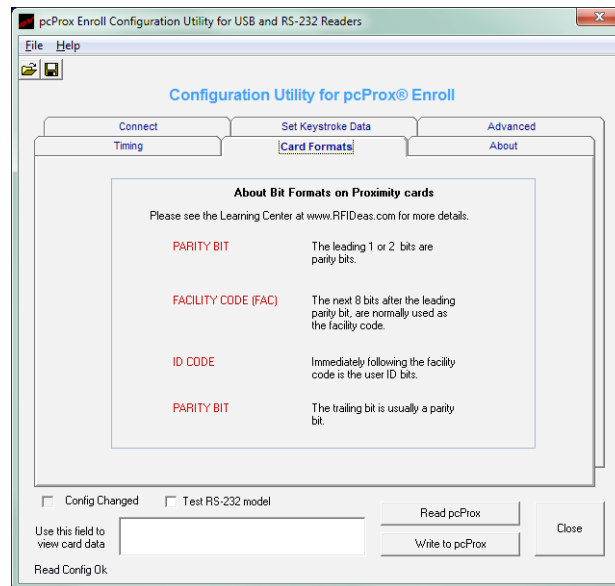
OR

Click the **Read pcProx** button if changes were made and you want to return to the configuration that was previously written to the device and abandon all changes made to the configuration.



Card Formats Tab: About Bit Formats on Proximity Cards

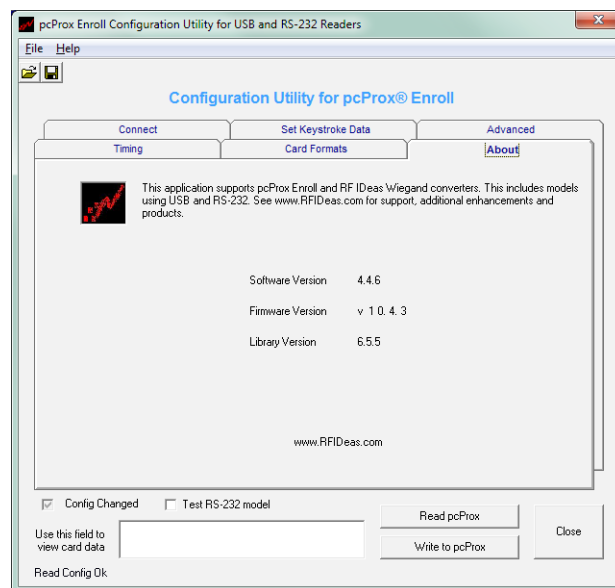
This tab defines the bit formats on proximity cards.



Please go to www.RFIDeas.com and follow the **Support** ⇒ **Learning Center** path for more details.

About Tab

This tab displays the software, firmware, and library versions.

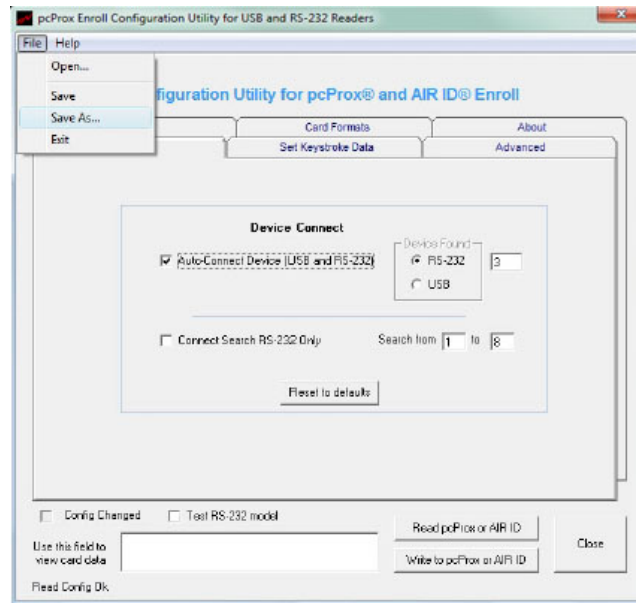


HWG Files for Device Deployment

If there are readers that require several unique configurations, create each configuration file and save each with a .HWG file name extension to the RF IDEas folder.

For example, a HWG file could identify a specific device configuration. Be aware that if the device is reading cards while you are configuring the device, the device might type the card data directly into the active document/window.

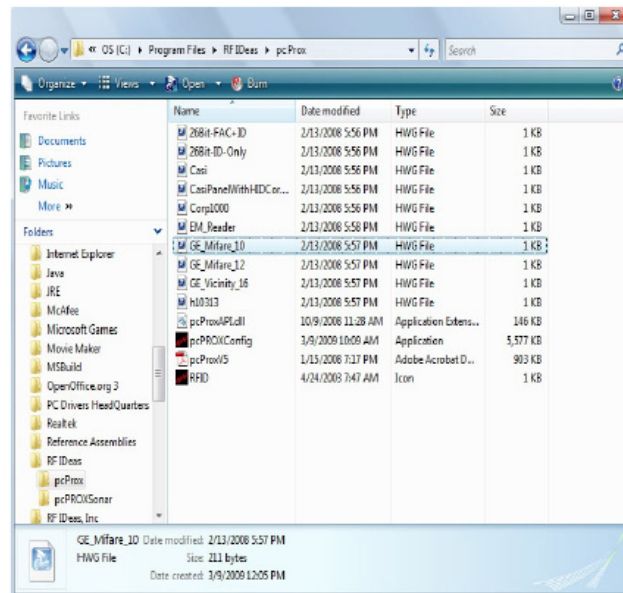
Click **File** ⇒ **Save As** to save the configuration.



Note: The operating system will not recognize the file name extension .HWG until the file association is made.

Save all different device configurations to a .HWG file. If you are using different card types, the .HWG file can quickly display the specific configuration for each card type.

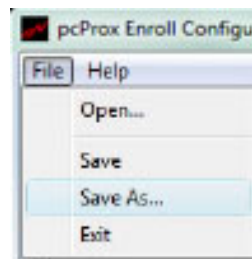
Click **Open** to save the .HWG file in the RFIDeas/pcProx folder. The **Open** button in the Save As window saves the file.



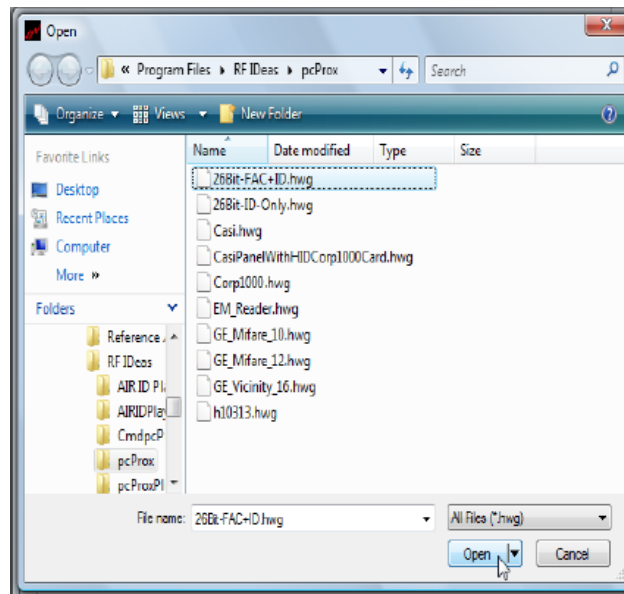
Note: ALWAYS click **Write to pcProx** to write the new configuration to flash memory of the device. Verify the Written OK message displays in the lower left corner of the configuration utility window. Place the card over the device to verify the new configuration was correctly read before saving a .HWG file.

Open a .HWG File

Click File ⇒ Open to load the configuration in the utility to configure additional readers.



Navigate to the C:\Program Files\RFIDeas\pcProx folder to open a specific configuration file. There are also predefined files in this directory that are configurations for different card types.

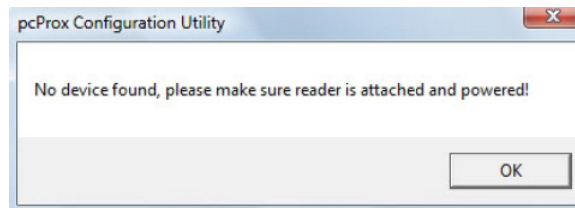


Click Open to display this configuration in the utility. Click **Write to pcProx** to save this configuration.



Troubleshooting

If the device is not working the following error message displays:



1. Check to be sure the device is plugged into the USB or RS-232 port. When the workstation is on and no card is being read, the LED is red. A valid proximity card causes the LED to turn green, provided the configuration is not set to only read certain bit lengths.
2. Only one COM port application can own the RS-232 port at a time. Make sure there is not another COM port application running. This prevents our software from seeing the device.
3. Verify the correct model and the software configuration screen agrees with the device attached.
4. Verify the port agrees with the workstation connector.
5. If the device still does not work, unplug it, remove 'General USB Device' using Windows 'Control Panel' ⇒ 'Add/Remove' Hardware. Then reboot the workstation. When the workstation boots up, re-attach the device USB and the OS should re-install the Windows driver automatically.

Change the release time to 1000 on the Timing tab for USB keystrokes to slow down the device. Open Notepad or Word and swipe a card to display the card data to see the actions of any nonprintable symbols.

If the device does not read the card, contact the card manufacturer/vendor to verify that the card type is compatible with the device model.

Precautions

Do not mount the device directly on a metal surface. This could interfere with the RF signal and the operation of the device.

The device may not recognize valid cards in the presence of high RF fields. If current readings are

erratic, take the following step:

- Move the equipment from any known transmitters nearby.

Contact Technical Support at 866.439.4884 for more information.

Before You Call Technical Support

Please make sure you've identified your reader model and credential type being used. Have this information ready so that your call will be routed to the correct specialist.

For Assistance:

Ph: 847.870.1723

E: TechSupport@RFIDeas.com

Talking To The Technician

Provide the reader model and credential type being used to the Technical Support Specialist.

Explain your problem to the specialist.

Be prepared to provide the following information:

- Error messages displayed on the computer
- What you were doing when the problem occurred
- What steps you have taken to resolve the problem, including results from each steps

Listen and follow the steps provided by the specialist. Let the specialist know what happens when you perform the steps.

Standard 26 Bit Format Structure

There are several bits constructed together that comprise data sent from the proximity card to the device. There are numerous bit formats and lengths for proximity cards. The most popular is a 26 bit card format. The typical layout for this format is 24 bits of usable information as the first and last are parity bits to ensure data integrity.

The 26 bit format consists of 255 possible facility codes. Within each facility code there is a total of 65,535 unique card numbers.

The standard 26 bit Wiegand format is H10301. It is binary encoded data. The format consists of 2 parity bits, 8 bit facility code (F) and 16 bit card number fields (B). This format displays below.

```
PFFFFFFFFBBBBBBBBBBBBBBBP  
XXXXXXXXXXXXX.....  
.....XXXXXXXXXXXXXO
```

Bit Coding

P = Parity

O = Odd Parity

E = Even Parity

X = Parity mask

F = Facility code, range = 0 to 255

B = Card Number, range = 0 to 65,535

In general, the 26 bit format is the industry standard format. Primary benefits of this include:

- Open format
- Convenient to order
- Universal access control panel acceptance

The sale of this format is not limited to any one company yet the range of card numbers available in this format is limited. There is a potential for card numbers to be duplicated.

Please go to www.RFIDeas.com and follow the **Support ⇒ Learning Center ⇒ Proximity Card Formats** link for more details. The card manufacturer may also have additional details about the card format.

Determine Card Format

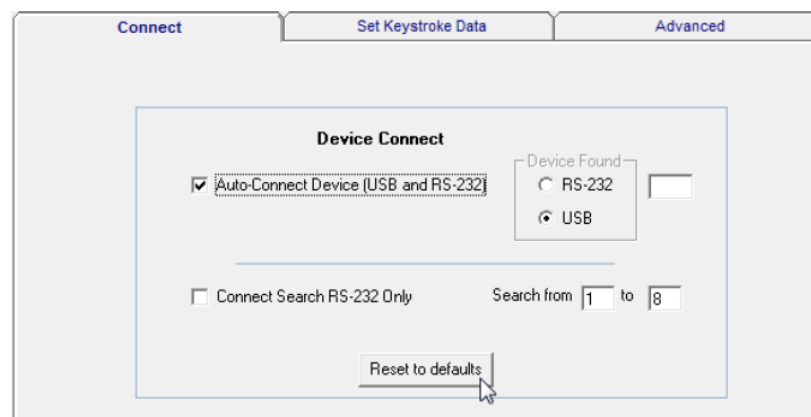
It is important to determine the card data format to locate the ID and FAC bits on the card. Cards use many different bit formats. It is necessary to determine any leading and trailing parity bits and exactly how the ID and FAC are configured. This data must be correctly interpreted to properly read card data.

Below is an example to determine a 26 bit card format.

Connect the pcProx device to the computer.

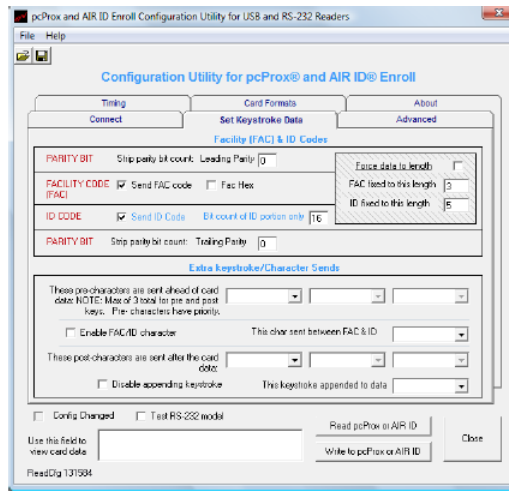
Open the pcProx Configuration Utility.

Click the **Connect** tab, click **Reset to defaults**.



Configure the Set Keystroke Data tab as shown below:

Function		Description
Leading Parity	0	
Send FAC Code	Check	
Send ID Code	Check	
ID Fixed to this Length	16	
Trailing Parity	0	

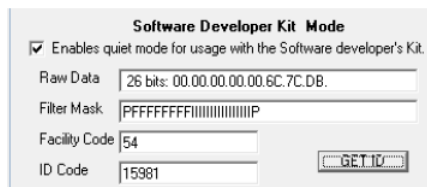


Check **Enable quiet mode for usage with Software Developers Kit** on the Advanced tab. Click **Write to pcProx**.

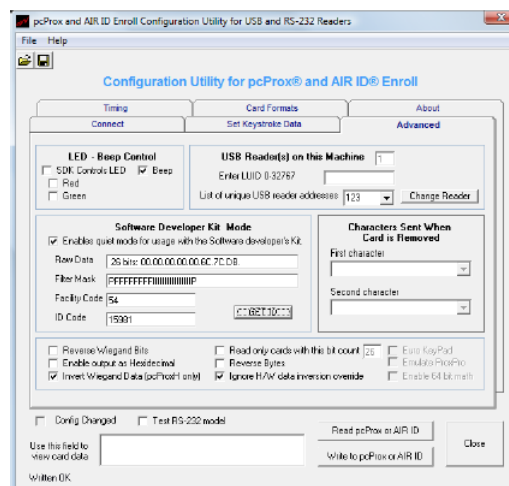
Present the card to the reader, when the LED is **green** click **GET ID**.

Note: Click “Get ID” at the same time the LED is **green**.

Click **GET ID** immediately after the card is read or the following message displays:



The card data displays in the test area. Screen should look similar to the one below.



Card data format in binary is typically: parity + FAC + ID + parity

There can be leading or trailing zeros to many parity bits.

In this example the card is 26 bit number is 00.00.00.00.00.6C.7C.DB.

Enter the above hexadecimal number into the Windows calculator and use the scientific mode to find the binary equivalent.

MSB	LSB
00011011000111110011011011	
001101100011111001101101	
(FAC54)	(ID15981)

Enter the decimal number you believe is on the card into the Windows calculator and use the scientific mode to find the binary equivalent. In the example above, a FAC of 54 and ID of 15981 were known values.

The leading parity bits are the most significant. The trailing parity bits are least significant.

The reader can now be configured to output the desired data.

Use the pcProx Device for Password Security - Complex Passwords

It is possible with certain limitations, to use the proximity token as a password for an application or operating system log on. The unique card bit-stream converted to either decimal or hexadecimal becomes the entire or a portion of the password. Enroll this card data to the password of the operating system application for the user.

Since the proximity token has no read/write memory there is no way go change this or write alphanumeric characters such as a user name to the proximity token. Some examples are shown below. Please see RF IDEas pcProx Playback Starter Kit or call the Sales Department if this capability is needed.

Several companies have adopted a policy that requires users to change their password every xx number of days to increase security. The PIN is the portion of the password the user changes every xx number of days. Since the card data is completely numeric, any alpha and upper/lower case letter constraints are handled in the user supplied PIN.

A two-factor authentication system is made up of:

1. Card ID data
2. Personal Identification Number (PIN)

The device may be configured to allow operation under either a one or two-factor authentication system.

One-Factor

In a one-factor system, the user simply scans the ID card. The device may be configured to add TAB keystrokes ahead of the data as well as a TAB or ENTER keystroke after the card data.

Two-Factor

The two-factor approach is especially useful when insisting on password construction rules or periodic changing of passwords.

In a two-factor system, the user may enter the PIN either before or after the card data. If the user adds the PIN before the card data, the device may be configured to append the ENTER keystroke.

Pre and Post Characters

There are some additional measures that can be taken to make it more difficult for unauthorized users to reproduce passwords.

Adding additional keystroke characters to the card information, that is difficult to re-produce, while configuring the data. These additional characters are labeled as Sp1, Sp2, and Sp3 on the delimiters tab menu selections.

pcProx Configuration Examples

The pcProx device can be configured to enter a password, log on to a system or unlock a workstation.

Change Password Configuration

To use the reader as a password device, first change the password. This operation is performed differently for each operating system and application. Please refer to the appropriate operating system user's manual for details.

Enter the old password and scan the ID card to enter the new password. Press Enter. Repeat, if applicable, to enter the new password into the Confirm Password field. If the new password is made up of the card data and a user supplied PIN, use card and type this PIN in both the New Password and Confirmation Password fields.

If the pcProx device is configured to append the Enter keystroke, re-configure the device for this Change Password operation to deselect this Enter keystroke. Once the password change is flashed to memory, add this post Enter keystroke.

Log On Device

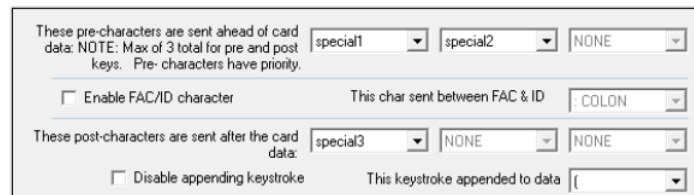
The operating system loads the USB driver prior to log on for the device to function as a log on device.

Make sure the cursor is in the Password field. Scan the ID card over the device to enter the password. If the password consists of the card data and a user supplied PIN, add this PIN. Click OK or press Enter.



No Software Log On

The Set Keystrokes Data tab configures the device to insert non-printable keystroke characters at the beginning, in the middle, and at the end of the card's unique data.



The device factory default, without the special configuration, displays: 15202752, in any application.

In pcProx and most text-based applications, the following will display: ??0007423?0000448?

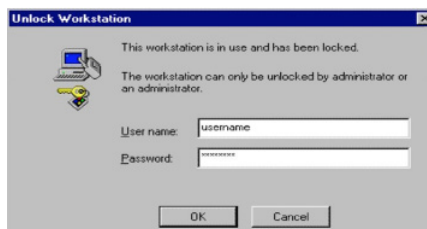
In Microsoft Word or Notepad it appears on screen as: ☺ ☹ ♥kv=007744110077441100b
Or depending on settings, it appears as: 00009750062336

This makes it much more difficult to reproduce a password.

Unlock a Work Station

Note: The following window displays slightly different for each operating system.

If this dialog box displays and the cursor is in the User name field, press **Tab**, or define one of the pre-card data keystrokes in the device to send the 'Tab' keystroke to move the cursor to the Password field.



Click **OK**, press **Enter**, or configure the device to add the Enter keystroke as an appended keystroke.

ASCII Command Protocol Overview

ASCII Command Protocol (ACP) allows the user to talk directly to the device without a DLL or special application. The serial Prox communicates using ASCII commands. Printable ASCII commands at 9600 baud, no parity 1 stop bit, and no echo, can be sent to the device.

Note: USB devices that are virtual COM port do not need the baud rate set. The input is buffered by the device and executed when a carriage return (CR) or line feed (LF) is typed. The unit then parses the command and performs the operation, and displays the results or error code. “\r\nRF IDEas>” where \r represents a CR and \n represents a LF that displays on the command line.

All commands begin with the prefix rfid: and end with a Return key, CR or LF.

Determine the COM Port

Windows

Use device manager to display the COM ports. Open the serial COM port. If it is a CDC virtual port, open the newly installed device that was created.

Linux

Most Linux distributions include Minicom. Download putty (www.putty.org) to communicate with the serial device if Minicom is not available.

After the USB CDC device is enumerated on the Linux machine a device of either /dev/ttyACM0 or /dev/ttyACM1 is found in the /dev/directory. Minicom users may have to create a symbolic link from /dev/ttyACM0 to /dev/modem using the command `ln -s /dev/ttyACM0 /dev/modem` or `ln -s /dev/ttyACM1 /dev/modem`.

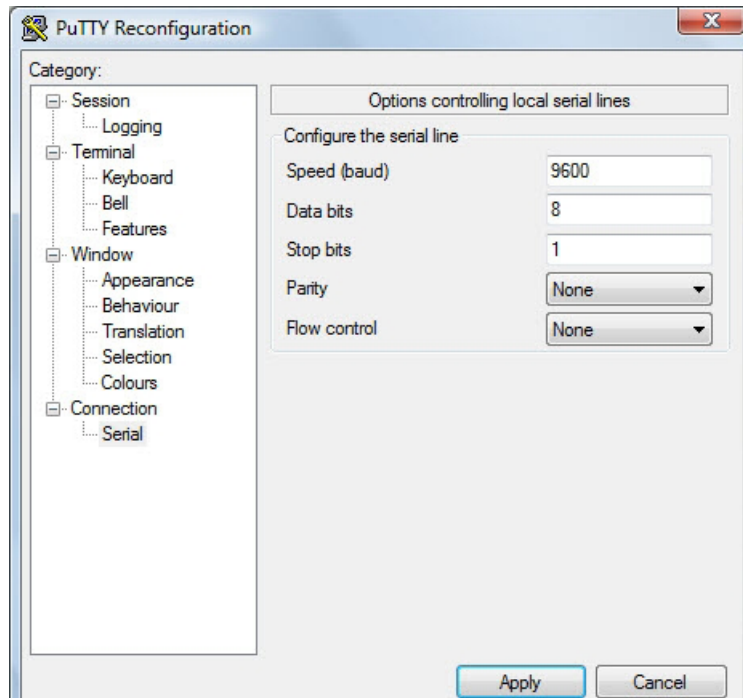
Mac OS X

The /dev/cu.usbmodemfa211 device is found on a Mac OS X . Use putty to communicate with this device.

Connect Serial Communications Program

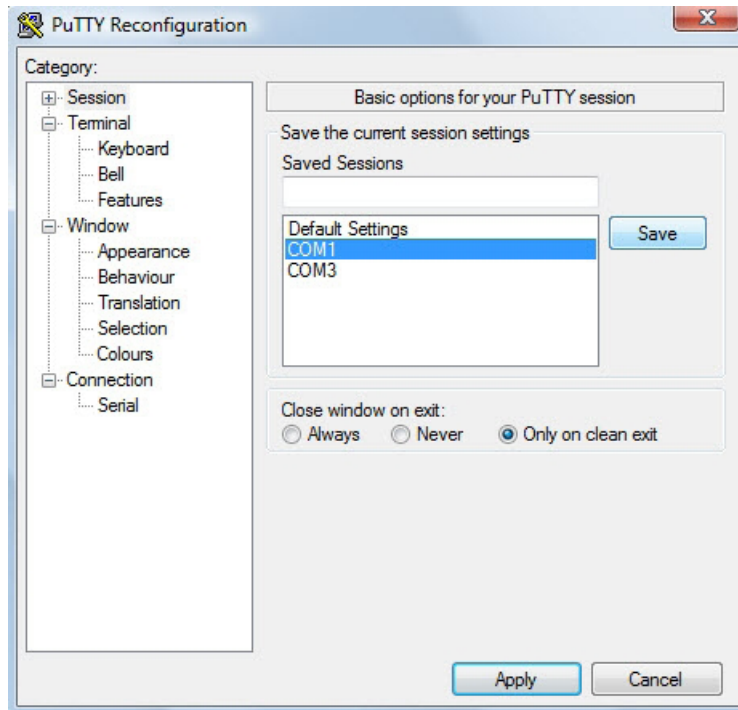
Open **putty.exe**. Click **Connection** ⇒ **Serial** and set the **Speed (baud)** to 9600, **Data bits** to 8, **Stop bits** to 1, and **Parity** to None. **Flow control** is not needed as there is no software or hardware handshaking.

Click **Session**.



Note: Use Hyper Terminal instead of putty with the XP operating system.

Highlight the appropriate session. Click **Save**. Click **Apply**.



PuTTY opens. Type **rfid:help** and press **Enter**. The Help command output displays. The complete list of Help command output is found in this section of the Configuration User Manual.

Command Structure

Commands are not case sensitive. Characters assigned to variables are case sensitive.

- All commands begin with a prefix string followed by one or more token strings with a period delimiter character between multiple tokens.
- Functions must end with a CR or LF.
- Variables can be assigned a value with an equal sign followed by the value or queried with a question mark.
- Any control characters other than CR, LF, and backspace terminate the command.
- The Escape key cancels a command.

The general syntax is:

PREFIX TOKEN { DELIMITER TOKEN } { { =Value } | { ? } }

The prefix string is rfid:

Command structure falls into one of three groups:

1. Perform a function.
2. Assign a variable.
3. Query a variable.

Perform a Function

A function performs an operation that may or may not display any results. A function may not be queried. An example of a function is to write the variable settings to flash memory using **rfid:cfg.write CR**.

Certain functions that display a value or series of values display the string between curly braces for easy parsing. For example, the **rfid:qid** function output displays:

{0x00BB,1,0x0000,80;0x000000801CD1931B2F14}

Assign a Variable

There are three types of variables:

1. Boolean
2. Integer
3. Character

Examples of Boolean Assignments

```
rfid:op.beep=0  
rfid:op.beep=true  
rfid:op.beep=False  
rfid:op.beep=F
```

Examples of Integer Assignment

```
rfid:out.led=0003  
rfid:out.led=3
```

Note: Some integer values require a 16 bit hexadecimal entry.

For Example: pcProx Plus card types: rfid:cfg.card.type=0xFFFF

Examples of Character Assignment

```
rfid:Delim.Chr.fac=':' CR  
rfid:Delim.Chr.fac='x3a' CR
```

Query a Variable

Query a single variable to display its current value.

- Booleans display as true or false.
- Integers display as 0..255 with leading zero suppression.
- Characters display as single quoted printable ASCII characters in the range 0x20..0x7E.
- Values from 0x00 .. 0x1F and 0x7F..0xFF will be with a leading backslash lowercase x and the two digit upper case hex number.
- The output of the variable displays between curly braces.

For example: **RF IDEas>rfid:out.led?**
 {3}

Variables are set and stored in RAM and are lost when the utility is closed. Use the **cfg.write** function to write the RAM configuration to flash memory. Use the **cfg.read** function to read the flash memory.

Help Command

Help displays the commands followed by its data type and expected syntax. The table below displays the Help command output. The functions display in blue.

HELP COMMAND OUTPUT	
rfid:beep.now	rfid:disp.fac.hex
‡ rfid:cfg	rfid:disp.fac.send
‡ rfid:cfg.card.hipri	rfid:disp.fac.strip
‡ rfid:cfg.card.list (Function)	rfid:disp.hex.lower
‡ rfid:cfg.card.type	rfid:disp.id.digits
rfid:cfg.read (Function)	rfid:disp.id.hex
rfid:cfg.reset (Function)	rfid:help (Function)
rfid:cfg.write (Function)	rfid:op.beep
rfid:chr.1	rfid:op.cont
rfid:chr.2	rfid:op.sdk
rfid:chr.3	rfid:out.led
rfid:chr.count.lead	rfid:qid (Function)
rfid:chr.count.trail	rfid:qid.hold
rfid:chr.eol	rfid:qid.id
rfid:chr.fac	rfid:qid.id.hold
rfid:chr.gone.1	rfid:time.hold
rfid:chr.gone.2	rfid:var (Function)
rfid:cmd.echo	rfid:wieg.id.bits
rfid:cmd.prompt	rfid:wieg.inv.bits
rfid:dev.luid	rfid:wieg.qual
rfid:dev.part (Function)	rfid:wieg.qual.bits
rfid:dev.ver (Function)	rfid:wieg.rev.bits
rfid:disp.64bit	rfid:wieg.rev.bytes
rfid:disp.fac.64bit	rfid:wieg.strip.lead.bits
rfid:disp.fac.digits	rfid:wieg.strip.trail.bits

‡ Applies only to the pcProx Plus

Help Command Summary

rfid:cfg.read

This function tells the device to read the flash memory to RAM.

rfid:cfg.reset

This function resets the flash memory to the factory settings.

rfid:cfg.write

This function tells the device to write the RAM to flash memory.

rfid:dev.part

This function displays the part number of the device

rfid: var

This function tells the device to display the variable command output. This is similar to a .HWG file.

QID

The **rfid:qid** function exists in four forms:

- rfid:qid (Function)
- rfid:qid.hold (Function)
- rfid:qid.id (Function)
- rfid:qid.id.hold (Function)

Each **qid** function returns the same queued ID. The last 3 items control what is cleared after the function displays the output. The top line below is an example output string. The bottom line displays how this example is formatted.

EXAMPLE Output String: {0x1000,2,0x0000,80;0x000000801DD1910B2F04}

FORMAT of Output String: {AGE,OVERRUN,LOCKOUT,BITCOUNT;ID}

AGE is the time in 48ms ticks that counts how long ago a card was scanned. This value count from 0 through 65535 displays in hex with "0x" hex notation. After 52.5 minutes the counter maxes out at 65535. The card data above shows this card was read 4,096 (0x1000 hex) x .048 = 196.608 seconds which equals 3 minutes and 16 seconds. The AGE counts until 65,535 (0xFFFF hex) and then maxes out. It will not roll over to zero. Use the qid.id function to clear the age counter.

OVERRUN is a counter from 0 through 255 displaying the number of cards scanned and over writes unread buffer contents. The device buffers one card. When a second card is read, the first card data is lost and the counter is set to one, meaning one card has overrun the buffer. The card data example above displays that 2 cards were read and the data from those cards was not transferred before reading this card.

LOCKOUT TIME is the number of 48ms ticks remaining until another card can be scanned. The card data above displays 10 times .048 which equals .48 seconds until the next card can be read.

BIT COUNT is the number of bits that follow 26 .. 255 and display as hex after the ';'. Notice the use of commas and semicolons. The card data example above shows that the ID contains 80 bits.

ID The card data above has 80 bits and is 0x000000801DD1910B2F04.

QID.hold

This reads the card data as above and resets the hold lockout timer. Once the card data displays, a second card can be read immediately after without waiting for the lock out time period to expire.

QID.id

This reads the card data and also clears the age,overrun, and bit count after the values display.

QID.id.hold

This reads the card data and clears the ID variables and hold timer like both combined functions above.

SDK Command

The `rfid:op.sdk=False` tells the device to display card data every time a card is scanned. If true, no card data displays. In the SDK mode, all keystroke or serial send data can be inhibited. The card data can be read using function `rfid:qid`.

Variable Command

The `var` command displays all variables. The command output can be captured and played back into the device. There must be a delay of several milliseconds after each character or the pcProx serial input buffer overflows.

VARIABLE COMMAND OUTPUT	
<code>rfid:beep.now</code>	<code>rfid:disp.fac.hex=False</code>
<code>rfid:cfg=1</code>	<code>rfid:disp.fac.send=False</code>
<code>rfid:cfg.card.hipri=True</code>	<code>rfid:disp.fac.strip=True</code>
<code>rfid:cfg.card.type=0xEF04</code>	<code>rfid:disp.hex.lower=</code>
<code>rfid:chr.1</code>	<code>rfid:disp.id.digits=0</code>
<code>rfid:chr.2</code>	<code>rfid:disp.id.hex=False</code>
<code>rfid:chr.3</code>	<code>rfid:op.beep=True</code>
<code>rfid:chr.count.lead=0</code>	<code>rfid:op.cont=False</code>
<code>rfid:chr.count.trail=0</code>	<code>rfid:op.sdk=False</code>
<code>rfid:chr.eol='\x0D'</code>	<code>rfid:out.led=255</code>
<code>rfid:chr.fac='\x00'</code>	<code>rfid:time.hold=20</code>
<code>rfid:chr.gone.1='\x00'</code>	<code>rfid:wieg.id.bits=16</code>
<code>rfid:chr.gone.2='\x00'</code>	<code>rfid:wieg.inv.bits=True</code>
<code>rfid:cmd.echo=True</code>	<code>rfid:wieg.qual=False</code>
<code>rfid:cmd.prompt=True</code>	<code>rfid:wieg.qual.bits=26</code>
<code>rfid:dev.luid=0x0000</code>	<code>rfid:wieg.rev.bits=False</code>
<code>rfid:disp.64bit=False</code>	<code>rfid:wieg.rev.bytes=False</code>
<code>rfid:disp.fac.64bit=False</code>	<code>rfid:wieg.strip.lead.bits=1</code>
<code>rfid:disp.fac.digits=0</code>	<code>rfid:wieg.strip.trail.bits=1</code>

These five variables work together to display leading and trailing (pre and post) card data delimiters.

1. `rfid:chr.1='\x00'`
2. `rfid:chr.2='\x00'`
3. `rfid:chr.3='\x00'`
4. `rfid:chr.count.lead=0`
5. `rfid:chr.count.trail=0`

The first three commands identify the pre delimiter characters that can display. Three characters may be divided up as pre and/or post delimiters. `Count.lead` identifies how many of the three characters (`chr.1` .. `chr.3`) display before the card data. For example, if `count.lead` is set to 1, only one character displays before the card data and `chr.2` and `chr.3` can be set as post delimiters. Then `count.trail` can have a value of 0, 1, or 2. If `count.lead` is 2, `chr.1` and `chr.2` are set as leading delimiters. Then only `chr.3` can be set as a trailing delimiter. The same character can not be used for both a leading and trailing delimiter.

rfid:chr.eol='\x0D'

This command sends the End Of Line (EOL) character at the end of the card data. Typically a carriage return (CR) (0x0D) is used.

rfid:chr.fac=':'

This command sets a delimiter between the FAC and card data.

rfid:chr.gone.1='\x0A' and rfid:chr.gone.2='@'

These commands prompt the device to send the characters 'x0A' and '@' when the ID card is removed if they are not '00'.

rfid:cmd.echo=True

This command echoes user input when true and controls if backspace sends a space, backspace, space to erase the last character typed. If false, it is turned off for computer control. This value can be written to flash memory using `cfg.write`. It defaults to true on `cfg.reset`.

rfid:cmd.prompt=True

This command displays the prompt when true. If false, the prompt does not display. This value can be written to flash memory using `cfg.write`. It defaults to true on `cfg.reset`.

rfid:dev.luid=0x1234

This command sets the logical unit ID. A user-defined 2 byte value to identify this unit.

rfid:disp.64bit=False

This command uses 64 bit math to computer 64 bit decimal digits. This should always be kept on. If true, it uses 64 bit math.

rfid:disp.fac.digits=3

This command truncates or sets the FAC display leading zero.

rfid:disp.fac.hex=False

This command sends the FAC code in hex when true. If false, the FAC code is sent in decimal.

rfid:disp.fac.send=False

This command sends the FAC code if true. If false, the FAC code does not display.

rfid:disp.fac.strip=False

This command separates the FAC from the card data when true so it can be independently formatted for display. If false, the FAC code is not separated from the card data.

rfid:disp.id.digits=16

This command sets the digits so the left most significant digits will be truncated. For example, if the card data is 1234 and `id.digits=3`, then only 234 displays. If the card data = 8 formats the display width by truncating digits or adding leading zeros.

rfid:disp.id.hex=False

This command displays the card data as hexadecimal when true. If false, the card data displays as decimal.

rfid:op.beep=True

This command sets the device to beep on a successful card read when true. If false, the device will not beep even if the card is successfully read.

rfid:op.cont=False

This command sets the device to continuously read when true. This tells the device to read the same card data over and over while the card is on the device. If false, the device only sends the card data once.

rfid:op.sdk=False

This command stops the device from displaying the card data when true, so the qid or SDK API call must be used to get the card data. When true the device will send the data via keystrokes or serial depending on device type/model.

rfid:out.beep=False

This command makes the device beep when true. If false, the device will not beep. This is only available on OEM converter boards.

rfid:out.led=255

This command sets the variable and also sets the output LED color in RAM. Use cf.write to write this change to flash memory to persist across power cycles.

LED Value	Description
0	OFF
1	Red
2	Green
3	Amber
4..254	Reserved
255	Controlled by the device

Rfid.out.relay= True

This command sets the output driver to ON (active low) when true. This is only available on OEM converter boards.

rfid:time.hold=20

This command sets how long in 48ms ticks the data is held for the active ID. This also controls how long the device keeps the LED green in 48 msec ticks. The default time is $20 * 0.048 = 0.960$ seconds.

Note: The quid.hold resets the internal timer this value initializes.

rfid:time.lo=24

This command sets how long in 48ms ticks the card device has to wait for no card in the RF field to begin accepting new card data. This prevents the same card data from being read over and over. If op.cont is true this value has no effect. The default time is $24 * 0.048 = 1.15$ seconds.

Note: The quid.hold resets the internal timer this value initializes, so that a new card can be read as

soon as the data is transferred to the host computer.

rfid:wieg.id.bits=80

This command sets byte reversal and also defines the FAC bit size.

rfid:wieg.inv.bits=True

This command sets all ones to become zero in the Wiegand data. If false, all zeros are set to become a one.

rfid:wieg.qual=False

This command sets card reading filter to off. If true, card reading filter is on. This is related to the next command, .qual.bits=80.

rfid:wieg.qual.bits=80

This command sets the device to read only cards with this many bits. All other size cards are filtered out.

rfid:wieg.rev.bits=False

This command does not reverse all bits. If true, the least significant bits are swapped with the most significant bits.

rfid:wieg.rev.bytes=False

This command does not reverse all the bytes in the id.bits size field. If true, all bytes are reversed.

rfid:wieg.strip.lead.bits=1

This command strips 0 .. 15 bits from the most significant bits.

rfid:wieg.strip.trail.bits=1

This command strips 0 .. 15 bits from the least significant bits.

ACP Error Codes

Function	Display String	Description
1	{Error#1}	Illegal command. Wrong or Missing Prefix (rfid:).
2	{Error#2}	Input buffer exceeded. Too many character were typed without a CR or LF.
3	{Error#3}	Illegal operation, such as trying to query or assign a variable to a function or trying to use a variable as a function.
4	{Error#4}	Range Error. The value assigned to the variable does not make sense for its data type, such as try to assign 257 to a byte value.

201 - 245 Bit: FASC - N With Appended Expiration Date

This output format consists of the 200 bit FASC - N followed by the expiration date. The expiration date is 8 BCD digits long (YYYYMMDD). A field separator is placed between the FASCN and the expiration date. The data stream is terminated with an end sentinel and an LRC.

Example:

Agency Code = 1341

System Code = 0001

Credential Number = 987654

Credential Series (Cred) = 1

Individual Credential Issue (Indiv Cred) = 1

Person Identifier (PI) = 1234567890

Organizational Category (Org) = 1

Organizational Identifier = 1341

Person/Organizational Identifier (Per/Org) = 1

Expiration date = 20110411 (YYYYMMDD = April 11, 2011)

11010	10000	11001	00100	10000	10110	00001	00001	00001	10000
Start	Agency Code 1341				Field	System Code 0001			
10110	10011	00010	11100	01101	10101	00100	10110	10000	10110
Field	Credential Number 987654							Cred	Field
10000	10110	00001	00001	01000	00001	10000	10000	00001	00100
Indiv Cred	Field	Person Identifier 1234567890							
10000	10000	10000	10000	11001	00100	10000	10000	10110	01000
		Org	Organizational Identifier 1341				Per/Org	Field Sep	
00001	10000	10000	00001	00100	10000	10000	11111	11001	
Expiration Date 20110411							End	LRC	

END-USER LICENSE AGREEMENT

LICENSE AGREEMENT

End-User License Agreement for RF IDEas™ SOFTWARE and HARDWARE - RF IDEas' pcProx®, Proximity Activated Readers, Software Developer's Kit, and Proximity Reader DLLs, and Protocol(s).

IMPORTANT-READ CAREFULLY: This End-User License Agreement ("EULA") is a legal agreement between you (either an individual or a single entity) and the manufacturer RF IDEas ("Manufacturer") with which you acquired the RF IDEas software and hardware product(s) identified above ("PRODUCT"). The PRODUCT includes the RF IDEas reader, computer software, the associated media, any printed materials, and any "on line" or electronic documentation. By installing, copying or otherwise using the PRODUCT, you agree to be bound by the terms of this EULA. The SOFTWARE PORTION OF THE PRODUCT includes the computer software, the associated media, any printed materials, and any "on line" or electronic documentation. By installing, copying or otherwise using the PRODUCT, you agree to be bound by the terms of this EULA. If you do not agree to the terms of this EULA, RF IDEas is unwilling to license the PRODUCT to you. In such event, you may not use or copy the SOFTWARE PORTION OF THE PRODUCT, and you should promptly contact the vendor you obtained this PRODUCT from for instructions on return of the unused product(s) for a refund.

The products described in this publication are intended for consumer applications. RF IDEas assumes no liability for the performance of product. RF IDEas products are not suitable for use in life-support applications, biological hazard applications, nuclear control applications, or radioactive areas. None of these products or components, software or hardware, are intended for applications that provide life support or any critical function necessary for the support of protection of life, property or business interests. The user assumes responsibility for the use of any of these products in any such application. RF IDEas shall not be liable for losses due to failure of any of these products, or components of these products, beyond the RF IDEas commercial warranty, limited to the original purchase price.

SOFTWARE PRODUCT LICENSE The PRODUCT is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The SOFTWARE PORTION OF THE PRODUCT is licensed, not sold.

1. GRANT OF LICENSE. This EULA grants you the following rights: *Software. You may install and use one copy of the SOFTWARE PORTION OF THE PRODUCT on the COMPUTER. *Network Services. If the SOFTWARE PORTION OF THE PRODUCT includes functionality that enables the COMPUTER to act as a network server, any number of computers or workstations may access or otherwise utilize the basic network services of that server. The basic network services are more fully described in the printed materials accompanying the SOFTWARE PORTION OF THE PRODUCT. *Storage/Network Use. You may also store or install a copy of the computer SOFTWARE PORTION OF THE PRODUCT on the COMPUTER to allow your other computers to use the SOFTWARE PORTION OF THE PRODUCT over an internal network, and distribute the SOFTWARE PORTION OF THE PRODUCT to your other computers over an internal network.

1.1 General License Grant RF IDEas grants to an individual, a personal, nonexclusive license to make and use copies of the SOFTWARE PRODUCT for the sole purposes of designing, developing, and testing your software product(s) that are designed to operate in conjunction with any RF IDEas designed proximity reader product. You may install copies of the SOFTWARE PRODUCT on an unlimited number of computers provided that you are the only individual using the SOFTWARE PRODUCT. If you are an entity, RF IDEas grants the right to designate one individual within your organization to have the sole right to use the SOFTWARE PRODUCT in the manner provided above.

1.2 Documentation. This EULA grants an individual, a personal, nonexclusive license to make and use an unlimited number of copies of any documentation, provided that such copies shall be used only for personal purposes and are not to be republished or distributed (either in hard copy or electronic form) beyond the user's premises and with the following exception: you may use documentation identified in the SOFTWARE PRODUCT as the file format specification for RF IDEas' proximity readers solely in connection with your development of software product(s) or an integrated work or product suite whose components include one or more general purpose software products.

1.3 Storage/Network Use. You may also store or install a copy of the SOFTWARE PRODUCT on a storage device, such as a network server, used only to install or run the SOFTWARE PRODUCT on computers used by a licensed end user in accordance with Section 1.1. A single license for the SOFTWARE PRODUCT may not be shared or used concurrently by other end users.

1.4 Sample Code. RF IDEas grants you the right to use and modify the source code version of those portions of the SOFTWARE PRODUCT identified as "Samples in the SOFTWARE PRODUCT" ("Sample Code") for the sole purposes to design, develop, and test your software product(s), and to reproduce and distribute the Sample Code, along with any modifications thereof, only in object code form.

2. DESCRIPTION OF OTHER RIGHTS AND LIMITATIONS.

*Limitations on Reverse Engineering, Decompilation and Disassembly. You may not reverse engineer, decompile, or disassemble the PRODUCT, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation

*You may not reproduce or otherwise emulate, in whole or in part, any form the protocol(s) defined within this PRODUCT for use without a RF IDEas PRODUCT Redistributable Code. If you are authorized and choose to redistribute Sample Code ("Redistributables") as described in Section 1.4, you agree to: (a) distribute the Redistributables in object code only in conjunction with and as a part of a software application product developed by you using the PRODUCT accompanying this EULA that adds significant and primary functionality to the SOFTWARE PRODUCT ("Licensed Product"); (b) not use RF IDEas' name, logo, or trademarks to market the Licensed Product; (c) include a valid copyright notice on the Licensed Product; (d) indemnify, hold harmless, and defend RF IDEas from and against any claims or lawsuits, including attorney's fees, that arise or result from the use or distribution of the Licensed Product; (e) otherwise comply with the terms of this EULA; and (g) agree that RF IDEas reserves all rights not expressly granted. You also agree not to permit further distribution of the Redistributables by your end users except: (1) you may permit further redistribution of the Redistributables by your distributors to your end-user customers if your distributors only distribute the Redistributables in conjunction with, and as part of, the Licensed Product and you and your distributors comply with all other terms of this EULA; and (2) in the manner described in Section 1.4.

*Separation of Components. The PRODUCT is licensed as a single product. Its component parts may not be separated for use on more than one computer.

*Single COMPUTER. The PRODUCT is licensed with the COMPUTER as a single integrated product. The PRODUCT may only be used with the COMPUTER.

*Rental. You may not rent or lease the PRODUCT without permission from RF IDEas

*Software Transfer. You may permanently transfer all of your rights under this EULA only as part of a sale or transfer of the COMPUTER, provided you retain no copies, you transfer all of the PRODUCT (including all component parts, the media and printed materials, any upgrades, this EULA and, if applicable, the Certificate(s) of Authenticity), AND the recipient agrees to the terms of this EULA. If the PRODUCT is an upgrade, any transfer must include all prior versions of the PRODUCT.

*Separation of Components. The PRODUCT is licensed as a single product. Its component parts may not be separated for use on more than one computer.

*Termination. Without prejudice to any other rights, RF IDEas may terminate this EULA if you fail to comply with the terms and conditions of this EULA. In such event, you must destroy all copies of the SOFTWARE PORTION OF THE PRODUCT and all of its component parts.

3. UPGRADES. If the SOFTWARE PORTION OF THE PRODUCT is an upgrade from another product, whether from RF IDEas or another supplier, you may use or transfer the PRODUCT only in conjunction with that upgraded product, unless you destroy the upgraded product. If the SOFTWARE PORTION OF THE PRODUCT is an upgrade of a RF IDEas product, you now may use that upgraded product only in accordance with this EULA. If the SOFTWARE PORTION OF THE PRODUCT is an upgrade of a component of a package of software programs which you licensed as a single product, the SOFTWARE PORTION OF THE PRODUCT may be used and transferred only as part of that single product package and may not be separated for use on more than one computer.

4. OEM COPYRIGHT. All title and copyrights in and to the PRODUCT (including but not limited to images, photographs, animations, video, audio, music, text and "applets," incorporated into the PRODUCT), the accompanying printed materials, and any copies of the SOFTWARE PORTION OF THE PRODUCT, are owned by RF IDEas or its suppliers. The PRODUCT and SOFTWARE PORTION OF THE PRODUCT is protected by copyright laws and international treaty provisions. You may not copy the printed materials accompanying the PRODUCT.

5. DUAL-MEDIA SOFTWARE. You may receive the SOFTWARE PORTION OF THE PRODUCT in more than one medium. Regardless of the type or size of medium you receive, you may use only one medium that is appropriate for your single computer. You may not use or install the other medium on another computer. You may not loan, rent, lease, or otherwise transfer the other medium to another user, except as part of the permanent transfer (as provided above) of the SOFTWARE PORTION OF THE PRODUCT.

6. OEM PRODUCT SUPPORT. Product support for the product is not provided by RF IDEas or its subsidiaries. For product support, please refer to the OEM supplies support number provided in the documentation. Should you have any questions concerning the EULA, or if you desire to contact OEM for any other reason, please refer to the address provided in the documentation provided.

FOR THE LIMITED WARRANTIES AND SPECIAL PROVISIONS PERTAINING TO YOUR PARTICULAR JURISDICTION, PLEASE REFER TO YOUR WARRANTY BOOKLET INCLUDED WITH THIS PACKAGE OR PROVIDED WITH THE SOFTWARE PRODUCT PRINTED MATERIALS.

Limited Warranty: RF IDEas warrants to the original buyer of this product, that the hardware and related disk(s) are free of defects in material and workmanship for a period of one year from date of purchase from RF IDEas or from an authorized RF IDEas dealer. Should the RF IDEas products fail to be in good working order at any time during the one-year period, RF IDEas will, at its option, repair or replace the product at no additional charge, provided that the product has not been abused, misused, repaired or modified. This warranty shall be limited to repair or replacement and in no event shall RF IDEas be liable for any loss of profit or any commercial or other damages, including but not limited to special, incidental, consequential or other similar claims.

No dealer, distributor, company, or person has been authorized to change or add to the terms of this agreement, and RF IDEas will not be bound by any representation to the contrary. RF IDEas SPECIFICALLY DISCLAIMS ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS OF PURPOSE. Since some states do not allow such exclusion of limitation of incidental or consequential damages for consumer products, check the statute of the state in which your business resides. This warranty gives you the specific legal rights in addition to any rights that you have under the laws of the state in which your business resides or operates.

Returns: RF IDEas products which require Limited Warranty service during the warranty period shall be delivered to the nearest authorized dealer or sent directly to RF IDEas at the address below with proof of purchase and a Return Materials Authorization (RMA) Number provided by RF IDEas Technical Support Dept. Replacement parts or complete boards become the property of RF IDEas. If the returned board or unit is sent by mail, the purchaser agrees to pre-pay the shipping charges and insure the board or unit or assume the risk of loss or damage which may occur in transit. The purchaser is expected to employ a container equivalent to the original packaging.

Copyright: Copyright by RF IDEas 2011. All rights reserved. Reproduction or distribution of this document in whole or in part or in any form is prohibited without express written permission from RF IDEas.

Trademarks: All RF IDEas products are trademarks of RF IDEas. All other product names or names are trademarks or registered trademarks of their respective holders.

Applicable Patents: RF IDEas pcProx Plus card readers supporting HID formats retain US Patent No. 5,952,935 and U.S. Patent No. 7,439,862.

Disclaimer: This Reference Guide is printed in the U.S.A. Any resemblance mentioned in the Reference Guide to persons living or dead, or to actual corporations or products is purely coincidental. RF IDEas believes that the information contained in this manual is correct. However, RF IDEas does not assume any responsibility for the accuracy of the content of this User Manual, nor for any patent infringements or other rights of third parties. RF IDEas reserves the right to make any modifications in either product or the manual without giving prior written notification.

FCC Compliance Statement

FCC ID: M9MPCPROXHUSB100 (HID USB model)	FCC ID: M9MBUPCPROXH100 (HID RS-232 model)
FCC ID: M9MPCPROXM101 (Indala model)	FCC ID: M9MBUPCPROXA100 (AWID)
FCC ID: M9MRDR6X8X (Kantech, Indala, Casi-Rusco)	FCC ID: M9MPCPROXP100 (Pyramid)
FCC ID: M9MPCPROXC101 (Casi-Rusco model)	FCC ID: M9MRDR7P71 (FIPS 201 13.56MHz)
FCC ID: M9MRFID18561100 (MIFARE/iCLASS models)	FCC ID: M9MRDR7L81 (Legic 13.56MHz)
FCC ID: M9MRDR7081 (iCLASS Module based)	FCC ID: M9MRDR7580 (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR7581 (iCLASS MIFARE and Other 13.56MHz)	FCC ID: M9MRDR7081AKF (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR7081AKE (iCLASS MIFARE and Other 13.56MHz)	FCC ID: M9MRDR75DX (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR8XX8U (Plus combo model)	FCC ID: M9MRDR758X (iCLASS MIFARE and Other 13.56 MHz)
FCC ID: M9MRDR8058X (Multi-protocol Combo model)	FCC ID: M9M8058XCCL (Multi-protocol and Contact model)
FCC ID: M9M758XCCL (MIFARE and Contact model)	FCC ID: M9M7580CCL (MIFARE and Contact model)
FCC ID: M9MRDR80081 (Plus SIO Combo Model)	FCC ID: M9MRDR70EX (13.56MHz Express Model)

"Pursuant to FCC 15.21 of the FCC rules, changes not expressly approved by RF IDEas might cause harmful interference and void the FCC authorization to operate this product.

Note: This device complies with Part 15 of the FCC Rules and Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation. This product complies with FCC OET Bulletin 65 radiation exposure limits set forth for an uncontrolled environment.

The reader may not recognize value cards in the presence of high RF fields. If the current reading is erratic, the user shall take the following step: Move the equipment from any known transmitters nearby. For more information contact Tech Support at 866.439.4884.

- A**
- API Mode 5-6
 - ASCII 4-5, 16, 20, 21, 23-26
 - Auto-Connect Device 16
- B**
- Beep 21
 - Bit Count of ID Portion Only 18
- C**
- COM 12, 16, 22
 - Confi Changed 16
 - Connect Search RS-232 Only 16
 - Credential Form Factors 7
 - CSN 7, 25
- D**
- Data 7, 19
 - Data is Valid For 17
 - Device Connect 15
 - Device Found 16
 - Device Manager 12
 - Disable Appending Keystroke 20
- E**
- Emulate ProxPro 26
 - Enable 64 Bit Math 26
 - Enable FAC/ID Character 20
 - Enable Output as Hex 25
 - Enable Quiet Mode for Usage w/ SDK 24
 - Enter LUID 0-32767 23
 - Ethernet 16
 - Euro Key Pad 26
- F**
- Facility (FAC) & ID Codes 18, 19, 24
 - FAC Fixed to this Length 19
 - FAC Hex 18
 - Filter Mask 24
 - Frequencies 7
 - Force Data to Length 19
- G**
- GET ID 24
 - Get/Set Internal Timing 17
- H**
- HWG 29, 30
- I**
- ID Fixed to this Length 19
 - Ignore H/W Data Inversion Override 25
 - Interface 10
 - Inter-Key Press Time 17
 - Inter-Key Release Time 17
 - Invert Wiegand Data 25
- K**
- Keyboard Emulation 5
 - Keystrokes 20
- L**
- LED 12, 21
 - List of Unique USB Reader Addresses 23
 - Lock-Out Time for Repetitive Reads 17
 - LUID 23
- O**
- OEM 12
- P**
- Part Number 9
 - pcProx 4, 6, 22, 27, 30, 31
- R**
- Raw Data 24
 - Read pcProx 125 kHz or 13.56 MHz
 - Read Only Cards W/ this Bit Count 25
 - Reset to Defaults 16
 - Reverse Bytes 25
 - Reverse Wiegand Bits 25
 - RS-232 11-12, 16, 20, 22, 23, 25, 26
- S**
- SDK Controls LED 21
 - Search From....To 16
 - Send Data Upon Read 17
 - Send FAC Code 18
 - Send FAC Only is Checked 19
 - Send ID Code 18
 - Send ID Only is Checked 19
 - Software Developers Kit (SDK) 4, 5, 6, 18, 21, 23, 24
 - Special1, Special2, Special3 20
 - Strip Parity Bit Count - Leading Parity 18
 - Strip Trailing Parity Bit Count 18
- T**
- Test RS-232 Model 16
 - This Char Sent Between FAC & ID 20
 - This Keystroke Appended to Data 20
 - Trailing Keystrokes 20
- U**
- UID 7
 - USB 11, 16, 20, 22, 24
- W**
- Wiegand 11, 25
 - Write to pcProx 125 kHz or 13.56 MHz

Other Products & Accessories



Software Developer's Kit

Allows independent developer's to use their application to read proximity access badge Read ID data of more than 1 billion cards in the field



PVC Label Proximity Card

Credit card size with paper release liner, 500 cards per box



Proximity Cards, Labels, Key Fobs

Complete selection of various manufacturers proximity cards, labels and key fobs. Marked with data code and ID number, available in several Wiegand formats



pcProx Read/Write Contactless

Reads and writes directly to the smart cards



pcProx Writer and Playback

Desktop read-only for iCLASS and NXP and smart cards



pcProx Playback Starter Kit

Plays back card sector data in ASCII or keystrokes



pcProx Sonar

Presence detector configured as a keyboard



PS/2 to USB Power Tap

Powers a USB RF IDEas device from a PS/2 port



Mounting Brackets

Further adjust the standard mounting of the device angle

RF IDEas Inc.
© 2013 RF IDEas. All rights reserved.

Specifications subject to change without notice.

Windows, Macintosh, Solaris, Sun Ray and Linux are trademarks of their respective companies.
All other trademarks, service marks and product or service names are property of their respective owners.

Mention of third-party products is for informational purposes only and constitutes neither an endorsement nor a recommendation. RF IDEas assumes no responsibility with regard to the performance or use of these products.
All understandings, agreements, or warranties, if any, take place directly between the vendors and the prospective users.

Please feel free to call, e-mail or visit our web site for a full list of applications, products, configuration options, supported cards and form factor specifications. Our web site includes application videos, support materials, case studies and detailed information about our product line.

Every effort has been made to ensure that the information in this manual is accurate. RF IDEas is not responsible for printing or clerical errors.