RFID EAS

pcProx® CHUID

Configuration Utility
User Manual

Rev B

Thank You!

Congratulations on the purchase of your pcProx® FIPS 201 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

The Basics 1

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

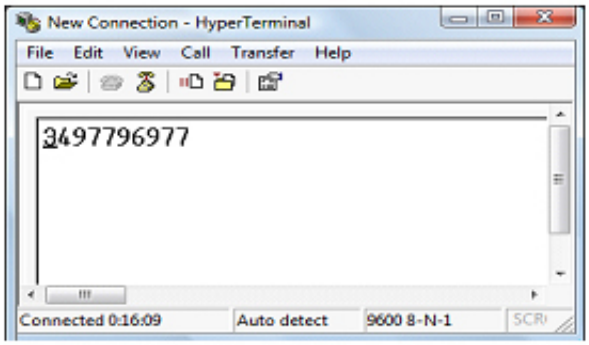
Output Formats

Keyboard Emulation

Card Data

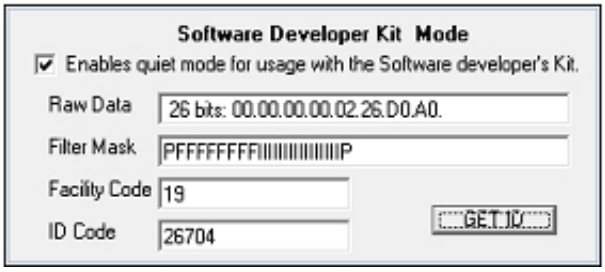
ASCII Output

Raw Card Data



SDK/API Mode

Raw Card Data



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).

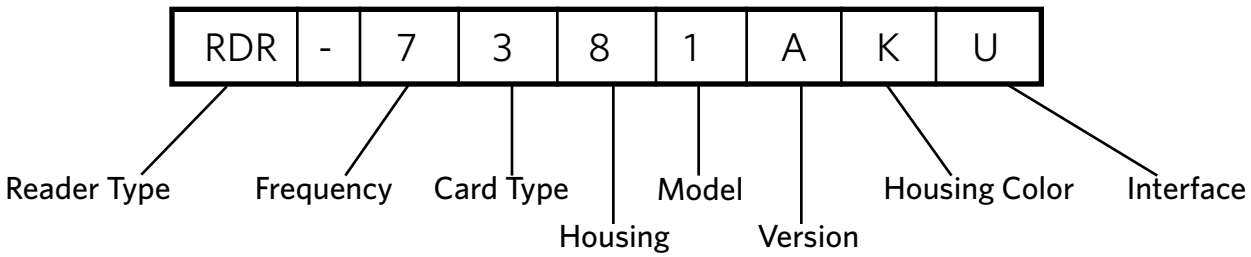
Hardware

2

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).

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 or USB 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.

Software

3

pcProx Plus CHUID Configuration Software Installation

Save the pcProx Plus CHUID 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 **pcProxCHUID-Config.exe** to download the installation utility. This file is the pcProx Plus CHUID installation program. Use this icon on the desktop to open the installation program to configure the device once the software is installed.

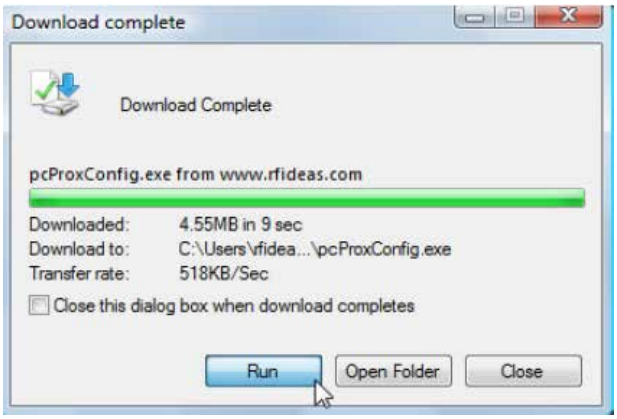
The File Download – Security Warning window displays.

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

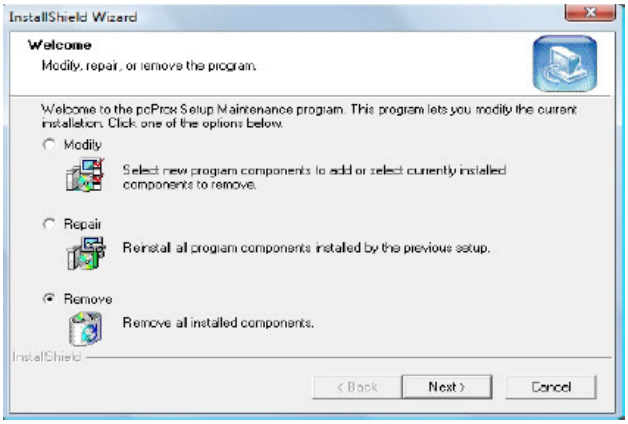
The configuration software downloads to the appropriate location.



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

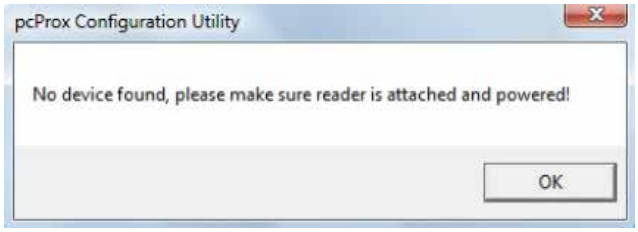
Follow the prompts to install the configuration software.

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:



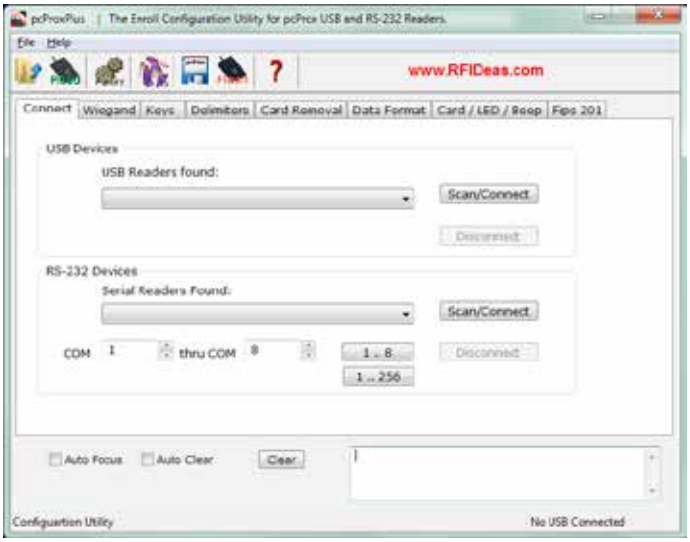
There are two utilities to configure the device. The following section describes the pcProx configuration utility just installed.

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 Plus CHUID Configuration Utility

The pcProx Plus CHUID Configuration utility is used to configure FIPS 201 CHUID or pcProx devices. The utility allows for more delimiters to be added with the card data and offers the 'Bit Wizard' function to determine which bits are card ID and which are FAC.



Button Menu Bar

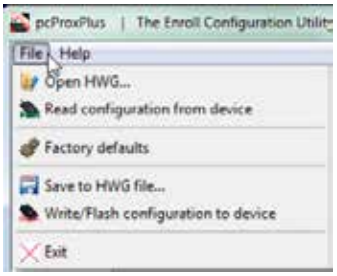
The Button Bar displays the following commands:

Commands	Click to...
File Open	Loads a specific configuration into the selected device.
Read	Loads the device's flash memory setting onto the screen to display the current device configuration.
Factory	Resets the device configuration to the factory default settings. This automatically flashes the device and then reads back the default configuration.
Bit Wizard	Opens the Bit Wizard Advanced Feature window.
Save	Saves the configuration as a HWG+ file.
Flash	Saves the configuration to the device s flash memory. The configuration is automatically saved to flash memory. No validation message displays.
About	Displays the application name, version and copyright information

If no device is found, the following message displays:

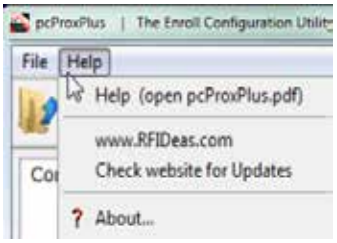
File Menu

The file menu lists the same commands as the Button Bar, but also includes the **Exit** command.



Help Menu

The Help menu lists the following commands:



Commands	Description
Help	Click to open the pcProx Plus User Manual
www.RFIDeas.com	Click to open the RFIDeas.com website
Check website for Updates	Click to go to the RFIDeas.com website to check for an update to the configuration utility.
About	Click to display the software name and library versions.

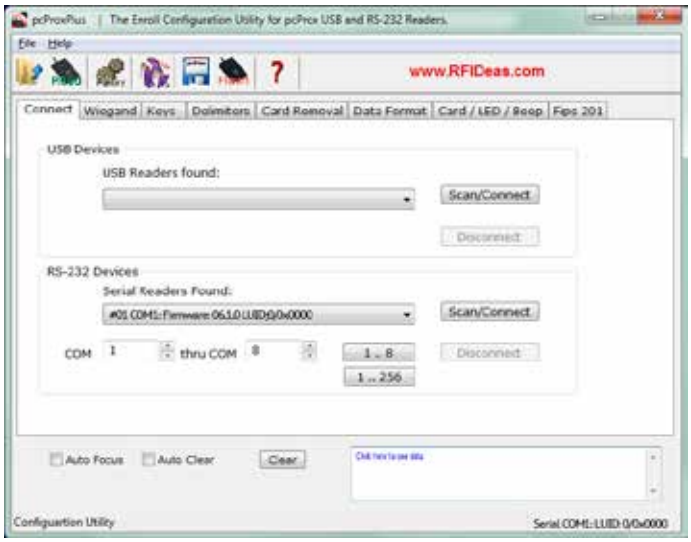
Connect Tab

The Connect section is used to connect the configuration application to the internal storage in the device. Once selected, you can read, modify, and write to the device.

USB Devices - Scan/Connect

This section scan the USB ports for active devices. The drop-down list is populated with the device(s) found. The list holds up to 16 devices and shows the Device firmware version and serial number.

When selecting devices from the drop down list the LED on the selected device will flash to identify it. The selected device is always displayed in the lower right corner of the status bar. When a new device is selected it will automatically read its configuration and update the values on the screen. When the selected device is unplugged, a popup dialog will inform you the active device is no longer active.



Note: The device connected that displays in the lower right hand corner matches the device information on the Connect tab. If, for example, the device connected is a pcProx FIPS 201 reader, the Delimiters, Card Removal, and Data Format tabs are not necessary. These functions are all configured on the Fields tab.

RS-232 Devices - Scan/Connect

This section scans for RS-232, physical COM port devices, virtual COM port devices, including USB, CDC and PCMCIA devices. It stops after the first USB device is found. The list holds only one serial device.

Note: The message, 'Click here to see data' only displays for serial readers. USB keystrokes can also be viewed in Notepad or any active window/application. If a USB device and a serial device(s) are connected, click **Disconnect** to configure a different device.

The configuration first looks for USB devices and keeps track of the first 16 devices in no specific order if 17 devices are connected. Every time a new device is connected, all devices may be read in a

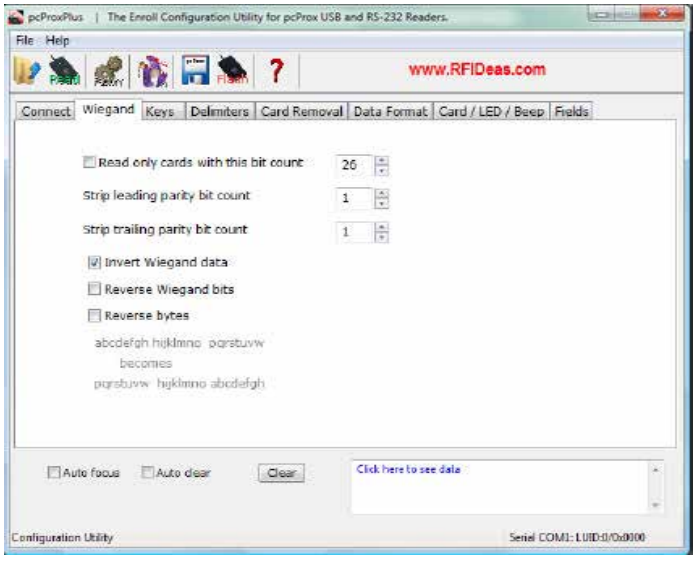
different order. For example, it may read ‘A B C’ the first time and ‘C B A’ the second time.

Function	Description
Scan/Connect	The utility searches for any USB or serial devices connected and stops when it finds the first one. A message states the number of devices connected. Click the device list to connect to a different device. Click Disconnect in the USB Devices section to connect to an RS-232 device.
Logical Unit ID (LUID)	If more than one device is connected, a LUID can be assigned to each device. This can be entered in hex or decimal.
Disconnect	Click to disconnect the device displayed.
COM thru COM	Select the appropriate COM ports that are connected to device(s).
1..8	Click to check only ports 1 to 8.
1..256	Click to check all ports 1 to 256.
Auto Focus	Check to automatically place the cursor in the test area to the right to see card data as it is read.
Auto Clear	Check to automatically clear the previous card data when the next card is presented to the device.
Clear	Click to clear the card data that displays in the test area to the right.

Wiegand Tab

Use this tab to configure the bits the device reads from the ID token.

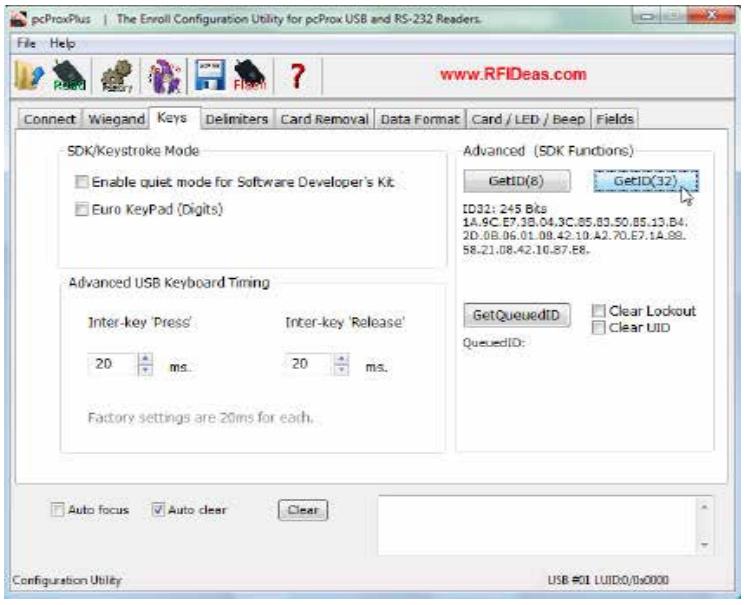
No data is sent from the device unless the bit count is matched. The total bits received from the card must match this bit count, parity bits included. If you select this checkbox and set the field for 26, the reader will only respond to 26 bit cards.



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.
Strip leading parity bit count	Set the device to strip leading parity bits from 0 to 15.
Strip trailing parity bit count	Set the device to strip trailing parity bits from 0 to 15.
Invert Wiegand data	Check if using a legacy application that requires the Wiegand data to be inverted.
Reverse Wiegand data	Check to reverse the order of Wiegand bits. This is primarily used for Card Key proximity 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.

Keys Tab

Use this tab to configure the device's keyboard parameters and Software Developer's Kit (SDK) advanced features.

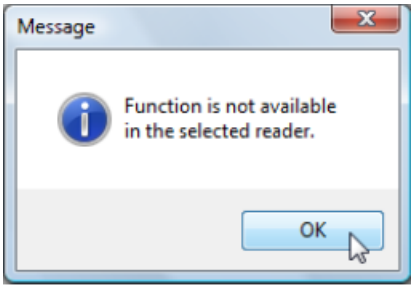


Function	Description
Enable quiet mode for Software Developer's Kit	If using the Software Developer's Kit (SDK), utilize this check box to stop the device from sending any card data as keystrokes or ASCII characters. With an SDK, users can change the mode of operation for the pcProx reader. For a USB reader this means from 'keyboard' to SDK. For an RS-232 reader this means from direct ASCII output asynchronously to the SDK's synchronous mode of polling. In the SDK mode, all keystroke or character send data is inhibited. The card's data may be read using functions included in RF IDeas' SDK. **Note:** The SDK is sold separately and is useful to software developer's to tightly integrate their software with the card reader. Please visit www.RFIDeas.com to learn more.
Euro Pad (Digits)	This feature is for compatibility with the AZERTY keyboard found in Europe. (i.e., keyboard numeric keypad keys).

Advanced USB Keyboard Timing

Function	Description
Inter-key 'Press' time	Sets USB inter-key Press' time delay in msecs. This is how long the key is held down. Default 20ms
Inter-key 'Release' time	Inter-key 'Release' time Sets USB inter-key 'Release' time delay in msecs. This is the delay between keystrokes. Default 20ms

If the device does not support the GetQueuedID function, the following message displays:

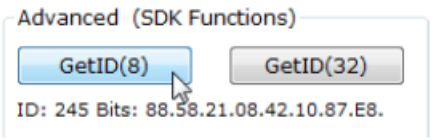


Advanced SDK Functions

Function	Description
GET ID(8)	Click while scanning a card over the device. The ID displays under the button. This returns 64 bits maximum.

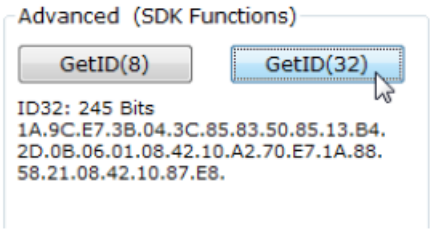
Function	Description
GET ID(32)	Click while scanning a card over the device. The ID displays under the button. This returns 255 bits maximum.
GetQueuedID	Click to display the last card data read. This returns 255 bits maximum.
Clear Lockout	Check to clear the time remaining to allow the device to read the next card immediately.
Clear UID	If clearUID is set, the card and the over run counters will be cleared for the next read. If clearHold is set, the reader will be ready to read another card immediately.

GETID(8) Data Display



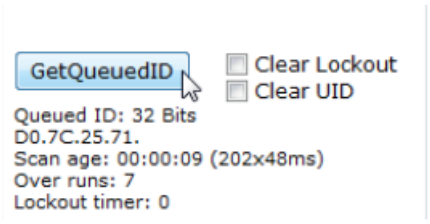
The Most Significant Byte is first – 88.
The Least Significant Byte is last – E8.

GETID(32) Data Display



The MSB is first – 1A.
The LSB is last – E8.

GetQueuedID Data Display



HH:MM:SS displays – 00:00:09.

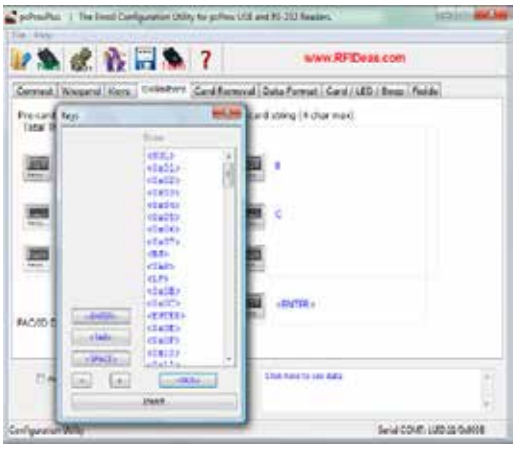
Delimiters Tab

This section allows you to configure the reader to send extra data, in addition to what is read from the card. For the USB model, this data is keystrokes, while for the RS-232 readers, this is ASCII characters. Due to the reader's internal limitations, there are a maximum of 3 keystrokes/characters which can be shared for either leading or trailing keystrokes purposes.

For example, you can define 2 pre-keystrokes, 1 keystroke between the facility code and ID, and 1 trailing keystroke plus an appending keystroke.

Click the appropriate keyboard icon to configure the pre and post delimiters and then select **Insert**.

> **Note**: Only 3 pre and post delimiters can be configured. If 3 pre-delimiters are set, no post delimiters can be set.
> Use backspace to delete unwanted keystrokes.
> The Scan Code output for the key selected displays above the list of keys.



Pre-Card String

The first section defines up to 3 pre-keystrokes or characters to be sent prior to any data retrieved from the card. The keystroke/characters are selected from the pull-down boxes.

> **Note**: There are 3 special characters (hex 01, 02, 03) that can be specified. These add an extra measure of difficulty in reproducing card data directly from the keyboard. It may be useful when using the proximity reader as a logon device.

FAC/ ID Delimiter

This field holds the character that will be placed between the FAC and ID codes.

> **Note**: There are 3 special characters (hex 01, 02, 03) that can be specified. These add an extra measure of difficulty in reproducing card data directly from the keyboard. It may be useful when using the proximity reader as a logon device.

Post-Card String

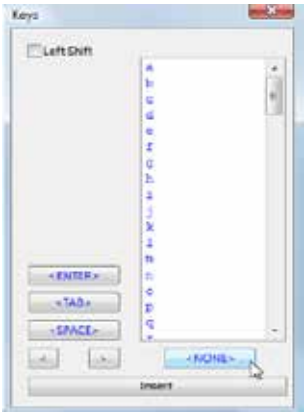
Select from 0 to 4 characters to send after the card data is sent. These first three characters are shared between the pre and post string.

Card Removal Tab

Use this tab to send up to 2 card removal ASCII characters when the card is removed from the reader.

Click the keyboard icon and select the appropriate ASCII character(s). Click **Insert**.

Click **<NONE>** to remove any characters.



Data Format Tab

Use the Data Format tab to set the length of the ID or FAC data, 64 bit math, and HEX numbers sent from the device.

Note: When **Fixed Length ID / FAC** Fields, **Send ID**, and **Send FAC** are checked the other fields are enabled.

Fixed Length ID / FAC Fields

Function	Description
Fixed Length ID / FAC Fields	Click to set the ID / FAC codes to a fixed length
ID Digits	Enter the number of zeros to add to the front the ID data to create a specific length.
FAC Digits	Enter the number of zeros to add to the front of the FAC data to create a specific length.

Send ID

Function	Description
Send ID	Check to send the ID portion of the card data from the device.
Send ID as hex number	Check to send the ID portion as a hexadecimal number.
Bit Count Portion of ID only	Enter the bit count of ID portion.

Send FAC

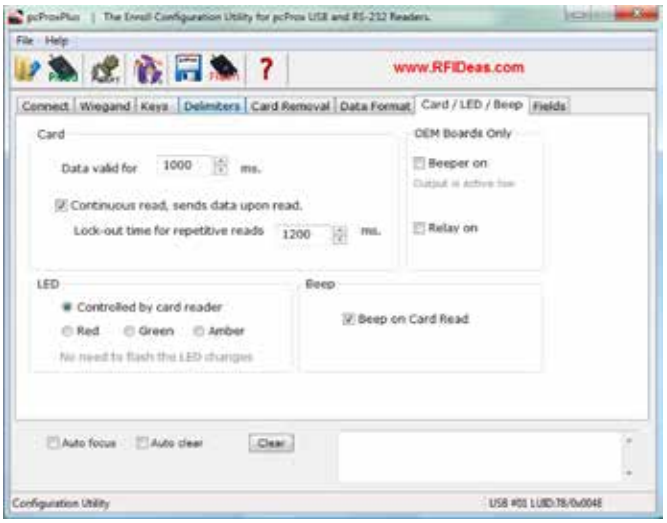
Function	Description
Send FAC	Check to send the Facility/Site (FAC) code
Send FAC as hexadecimal number	Check to send this code in hexadecimal. This is set for KANTECH 10 proximity cards.

64 Bit Math

Function	Description
64 Bit Math	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. Check this to display the true representation of the number or if the card is over 32 bits.

Card / LED / Beep Tab

Use the Card / LED / Beep tab to configure the pcProx, OEM board, LED and Beep.



Card

Function	Description
Card data 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 1,000.
Continuous read, sends 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.

OEM Boards Only

Function	Description
Beeper On	Check this to turn the device beeper on.
Energize Relay	Check this to activate the OEM board.

LED

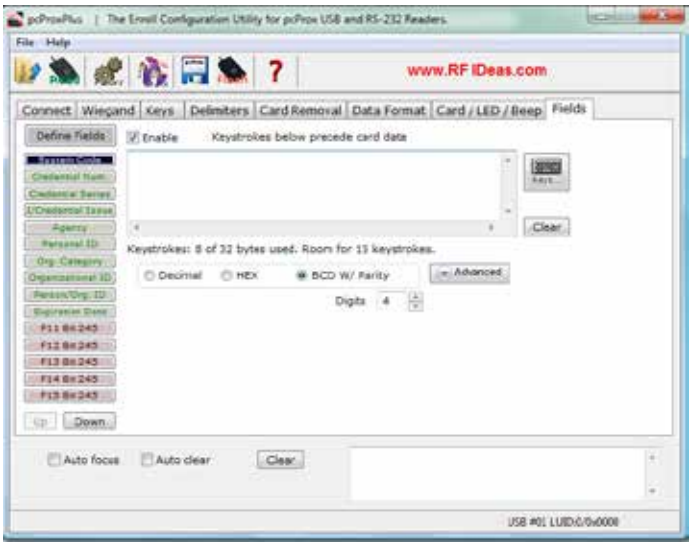
Function	Description
LED Controlled by card	Check this to turn the device beeper on.
Red	Select this to set the LED color to red.
Green	Select this to set the LED color to green.
Amber	Select this to set the color to amber.

Beep

Function	Description
Beep on Card Read	Check this to set the device to beep when a card is read

Fields Tab

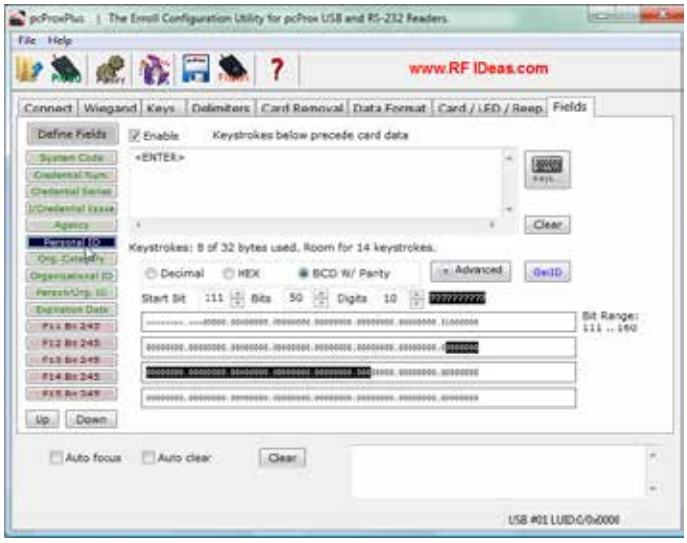
This tab allows manipulation of all fields on the Federal Information Processing Standard (FIPS) 201, iCLASS, MIFARE or proximity cards. Use the red buttons to configure additional fields. The fields can be moved to change the order displayed in the binary bit pattern display.



Function	Description
Define Fields	Click to display the reader hardware type. The correct type must be selected to allow for all card bits to be manipulated.
Enable	Check to enable the highlighted field. This allows the delimiters to be output and the corresponding card field to be processed and output. All green fields are enabled. All red fields are disabled.
Keys	Click to select keystrokes that are stored in the device's flash memory that precede card data output. Each field may have from 0 – 15 keystrokes.
Clear	Click to clear keystrokes preceding the card data.
Decimal	Click to display the card field as an unsigned base 10 number.
Hex	Click to display the card field as a base 16 number in uppercase HEX 0 – 9 and A – F.
BCD w/ Parity	Displays the card data in binary coded decimal, where each 5 bits represent 1, 2, 4, 8, and parity. FASCN data is always odd parity.

Advanced Button

This displays the bit ranges of the card.



Click each field button to display the location of the card binary data. In the example below, the Personal ID starts at bit 111, is 50 bits long, and is 10 digits. The Bit Range is 111 .. 160 and the card bit pattern is highlighted. This output format is displayed in binary coded decimal with parity (BCD w/ Parity). This is the 245 bit configuration. If any additional keystrokes were entered to precede the card data, click Clear to remove them.

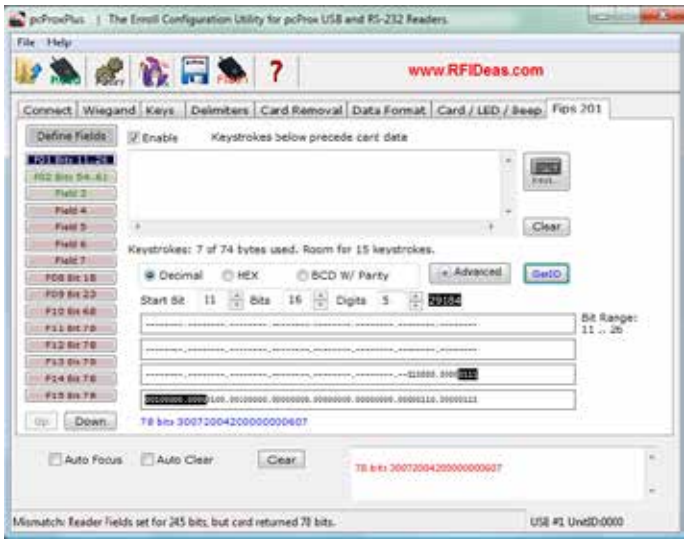
Function	Description
Advanced	Click to display the binary bit pattern.
GetID	Click to display the binary bit pattern captured from the card.
Start Bit	Enter a number to define the left most significant starting bit for the field.
Bits	Enter the number of bits to add to the Start Bit to define the range of bits in the field.
Digits	This is the number of digits that will display in a selected field.
Up	Click to move the highlighted field up one position.
Down	Click to move the highlighted field down one position.

Note: The message that displays the number of bytes used and how much room for keystrokes above the **Advanced** button is determined by the device's flash memory. In this example the configuration is: "Keystrokes: 8 of 32 bytes used. Room for 14 keystrokes." Every field is 15 keystrokes maximum. All fields share 96 bytes.

The Bit Range that displays to the left of the binary bit pattern is the Start Bit field total + the Bits field total – 1.

Get ID

Click **GetID** and scan the card to display the output format of the FIPS 201, iCLASS, MIFARE and proximity card and the interpretation display of the card data. Click GetID to define the fields to set up the device.

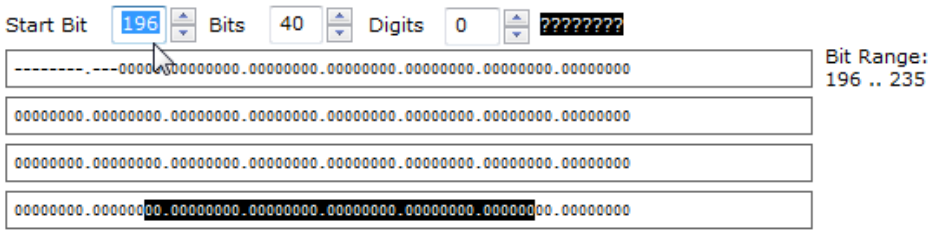


In this example, The Agency data starts at bit 11, is 16 bits long, and is 5 digits. The location of the agency data is highlighted in the binary bit pattern. The Bit Range is 6 .. 25.

The actual card data displays in blue below the binary bit pattern layout. The interpretation of the card data displays in red in the text field. The card data in blue will always be the same. The card data in red changes based on configuration settings flashed to the device.

Note: Click Clear to delete the red card data in the text field. A confirmation message displays.

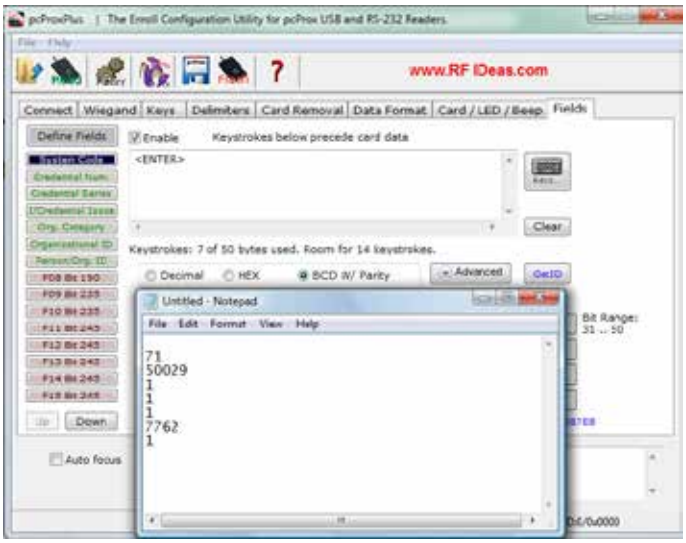
The Start Bit changes the actual location of the selected field on the binary bit pattern.



Note: The '???' that display to the right of the Digits field indicate the BCD parity is incorrect. Verify the correct field is selected.

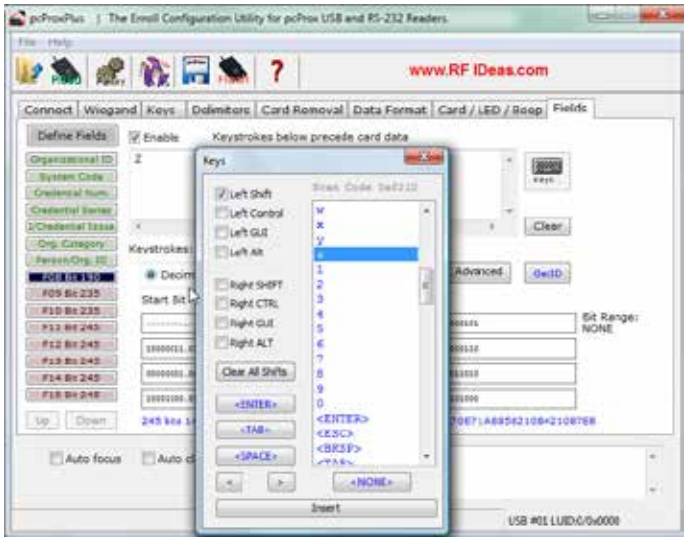
Change Fields Configuration

Click on the appropriate field button and uncheck Enable to remove field data from being displayed. In the example below, the Agency, Personal ID, and Expiration Date fields have been removed. Additional function keys display to configure more fields.



Assign Preceding Keystrokes

If Enable is checked for a field, specific keystrokes can be assigned to precede card data output.

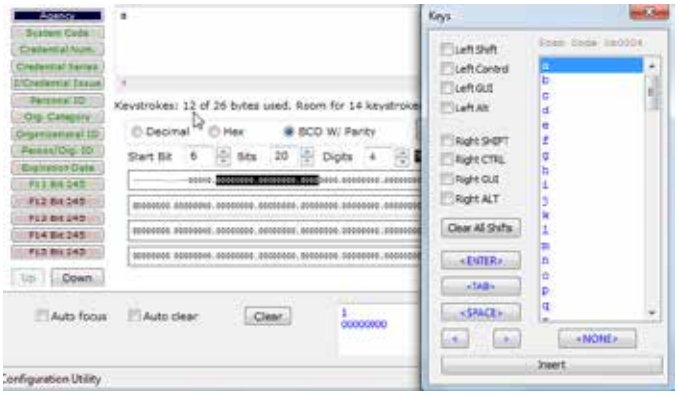


Note: The Scan Code output for the key selected displays above the list of keys.

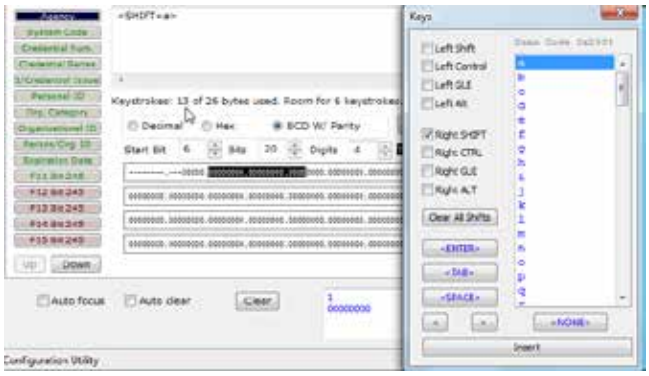
Click **Clear** to remove all preceding keystrokes as appropriate.

Note: The 'Left Shift' is the only key in the list on the right column in the Keys window that equals 1 byte of memory. Selecting 'Left Control' to 'Right ALT' and a character from the list equals 2 bytes of memory.

Each single keystroke entered to precede card data equals 1 byte of memory.



If any special character in the left column except for 'Left Shift' is selected with a keystroke, this equals 2 bytes of memory.



If all the keystrokes have been assigned to the fields, the following message displays:

Depending on the active document/window, additional functionality can be assigned to a field. For example, if the card data is read in OpenOffice, the Note feature can also be assigned.

Select the appropriate field. Click the keyboard icon. Check **Left Control**. Check **Left Alt**. Click **n**. Click **Insert**. Click **Flash** to write this configuration to flash memory. Verify the active window is OpenOffice. Scan the card. The Notes function opens when the card is read.

The value assigned to the function key in the active document/window determines the output.

Note: This configuration utility creates a ComSpecPort.txt file and saves to the default directory. This file can be opened and deleted at will.

FIPS 201 Card Configuration

In order to configure a FIPS 201 card:

- Click **Advanced** to display the card data in the binary bit pattern to determine bit length and format
- Click **GetID** and present the card to the reader
- Define the fields to match the specific output
- Configure any additional fields as appropriate
- Flash the configuration to memory

The **Advanced** button displays:

- Start bit location
- Number of bits for a specific field
- Number of digits for the field
- Location of the field within the 245 bit range

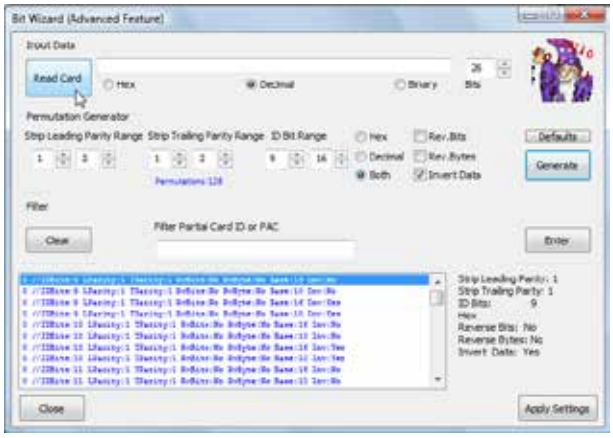
Bit Wizard (Advanced Feature) 4

Advanced Configurations with the Bit Wizard

Use the Bit Wizard (Advanced Feature) to select from a list of data permutations to assign to a card.

Input Data

Click Read Card and scan the card over the device.



The card data displays in Hex and can read 32 bits in the example above. Based on the permutation generator options selected, click **Generate** to display the hexadecimal and decimal permutations for the card data.

Below is the same card data displayed in decimal format.

Below is the same card data displayed in binary format.

Note: These fields can be edited.

Permutation Generator

Change the numbers in the following fields as appropriate. These results display when Generate is clicked.

- Strip Leading Parity Range
- Strip Trailing Parity Range
- ID Bit Range

Note: The minimum can not be higher than the maximum range number or the "Range Error on Controls" message displays below the **Strip Trailing Parity Range** fields.

After both hexadecimal and decimal permutations have been created as in the example above where **Both** is selected, click either **Hex** or **Decimal** so only half the number of permutations display. Then click **Generate**.

After only the **Inverted Data** permutations display as in the example above, click the **Rev.Bits** or **Rev. Bytes** to add these to the permutations display. Then click **Generate**.

These are the minimum left field and maximum right field combinations for these fields.

Note: Click Defaults to return to the default factory settings.

Enter the appropriate filters in the **Filter Partial Card ID or FAC** field and click **Enter**.

The permutations that match the filter display in the test area.

The data details to the right display the layout related to the highlighted permutation in the test area.

Click Clear to remove the filter characters. To display the previous list of permutations, clear the filter field and click Generate.

Select the appropriate permutation to write to flash memory.

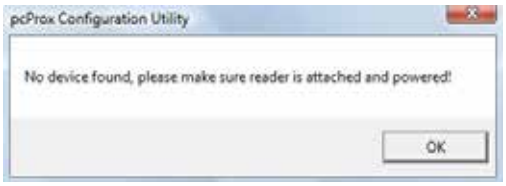
Click **Apply Settings** to copy the configuration to the dialog controls. The Bit Wizard closes. The following message displays:

Click **OK**. Click **FLASH** to write the configuration to flash memory.

Tips and Troubleshooting 5

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.

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.

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: M9MPCPROXM101 (Indala model)
FCC ID: M9MRDR6X8X (Kantech, Indala, Casi-Rusco)
FCC ID: M9MPCPROXC101 (Casi-Rusco model)
FCC ID: M9MRFID1856I100 (MIFARE/iCLASS models)
FCC ID: M9MRDR7081 (iCLASS Module based)
FCC ID: M9MRDR7581 (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR7081AKE (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR8XX8U (Plus combo model)
FCC ID: M9MRDR8058X (Multi-protocol Combo model)
FCC ID: M9M758XCCL (MIFARE and Contact model)
FCC ID: M9MRDR80081 (Plus SIO Combo Model)
FCC ID: M9MBUPCPROXH100 (HID RS-232 model)
FCC ID: M9MBUPCPROXA100 (AWID)
FCC ID: M9MPCPROXP100 (Pyramid)
FCC ID: M9MRDR7P71 (FIPS 201 13.56MHz)
FCC ID: M9MRDR7L81 (Legic 13.56MHz)
FCC ID: M9MRDR7580 (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR7081AKF (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR75DX (iCLASS MIFARE and Other 13.56MHz)
FCC ID: M9MRDR758X (iCLASS MIFARE and Other 13.56 MHz)
FCC ID: M9M8058XCCL (Multi-protocol and Contact model)
FCC ID: M9M7580CCL (MIFARE and Contact 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.

Index

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
EXXXXXXXXXXXX.....
.....XXXXXXXXXXXXO

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.

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 delimeters 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.

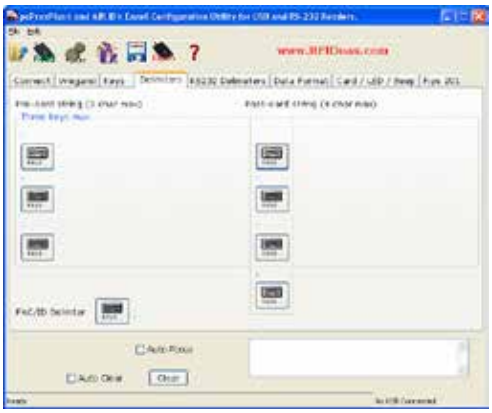
Make sure the cursor is in the Password field. Scan the ID card over the device to enter the password. If the password consists of the card data and a user supplied PIN, add this PIN. Click OK or press Enter.

Security of the Card Data

If there is a concern about individuals being able to read the contents of the proximity token, and thereby reproduce the password, there are some additional measures that can be taken.

During the configuration process you may add additional keystroke characters to the card information that are difficult to re-produce. These additional characters are labeled special1, special2, & special3.

As an example, here is a screen shot that configures the reader 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 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.

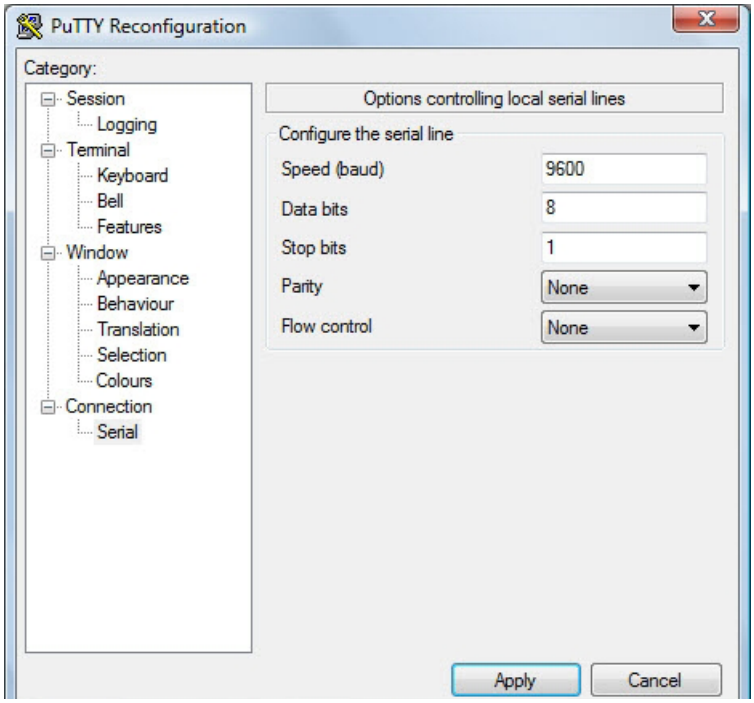
After the USB CDC device is enumerated on the Linux machine a device of either /dev/ttyACM0 or /dev/ttyACM1 is found in the /dev/directory. Minicom users may have to create a symbolic link from /dev/ttyACM0 to /dev/modem using the command ln -s /dev//tty/ACM0 /dev/modem or ln -s /dev/ttyACM1 /dev/modem.

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

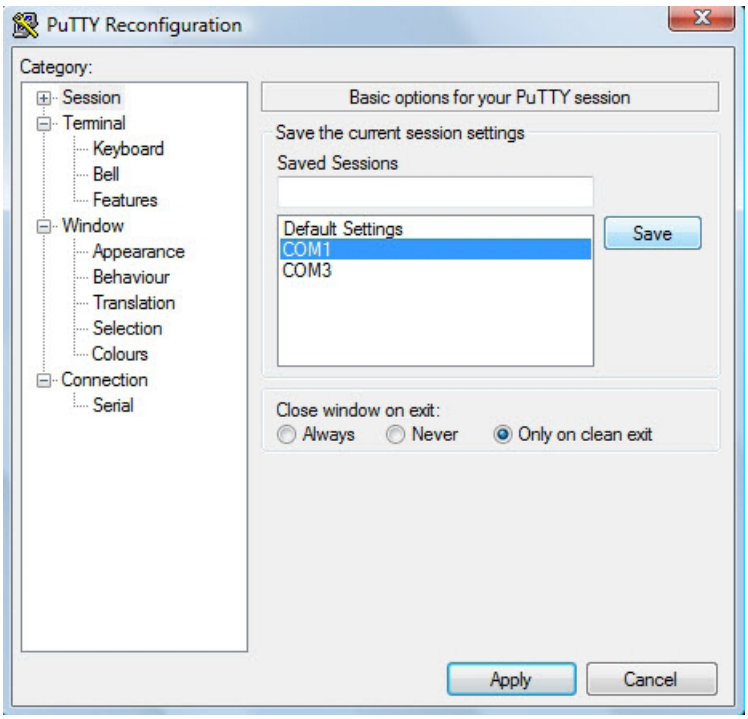
Open **putty.exe**. Click **Connection ⇨ Serial** and set the **Speed (baud)** to 9600, **Data bits** to 8, **Stop bits** to 1, and **Parity** to None. **Flow control** is not needed as there is no software or hardware handshaking.

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 o f 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 (Funtion)	rfid:disp.id.hex
rfid:cfg.reset (Function)	rfid:help (Funtion)
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 **quid** 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

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

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	

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

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.