15370 Barranca Parkway
Irvine, CA 92618

OMNIKEY 5x27 CK

Keyboard Wedge Configuration

USER GUIDE

5127-902, C.0

March 2013

Contents

List of Figures

List of Tables

Trademarks

Revision History

Date	Author	Description	Document Version
021313	Kieran Wingfield	Data selection and Manipulation added	C.0
082212	Juergen Kusper	Changed product number from 5127CK to 5x27 CK	B.1
062712	Juergen Kusper	Service Pack 1 Features added: - PACS bits parsing - DEC Output	B.0

Contacts

North America	Europe, Middle East and Africa
15370 Barranca Parkway Irvine, CA 92618 USA Phone: 800 237 7769 Fax: 949 732 2120	Phoenix Road Haverhill, Suffolk CB9 7AE England Phone: +44 1440 714 850 Fax: +44 1440 714 840

Asia Pacific
19/F 625 King's Road North Point, Island East Hong Kong Phone: 852 3160 9800 Fax: 852 3160 4809 support.hidglobal.com

1 Overview

HID Global's OMNIKEY® 5x27 CK readers open new market opportunities for system integrators seeking simple integration and development of readers using the standard CCID (Circuit Card Interface Device).

With the keyboard wedge functionality, users of OMNIKEY 5x27 CK readers can retrieve data from a card that is presented to the reader and directly input the card data into an application using keystroke emulation. This eliminates the need for customers to manually enter the card data into an application.

This guide explains how to setup the reader to use different card types in the Keyboard Wedge mode using the web browser interface.

In order to use the reader browser interface, the EEM-USB driver must be installed.

For installation instructions see the OMNIKEY 5x27 CK Quick Start Guide (5127-901).

Note: HID provides various Service Packs for the OMNIKEY 5x27 CK. Some functions have been introduced with later Service Packs only, in such cases you will find these exceptions noted in this user guide. For downloading the latest Service Pack for your OMNIKEY 5x27 CK reader, access the Developer Center: http://www.hidglobal.com/main/developers/omnikey-5127-ck/

Service Packs are available in the **Downloads** section.

Check the firmware version of the OMNIKEY 5x27 CK Reader from the **General Overview** tab in the built-in web interface (see Section 1.2 Reader Web Interface, page 5).

1.1 References

Document Number	Description
5127-901	Quick Start Guide
5127-903	Software Developer Guide
AN0407	Firmware Upgrade

1.2 Reader Web Interface

Start your browser on the Computer.

1. Enter http://192.168.63.99/ into the address bar and press Enter. The web server **Start** page opens.

1.3 Change Settings

When altering configuration parameters the description or value color changes to green (Figure 1 – System Configuration).

Note: Modifications are only applied when **Apply Changes** is executed in the web interface (see Section 2 Keyboard Wedge Mode, page 7).

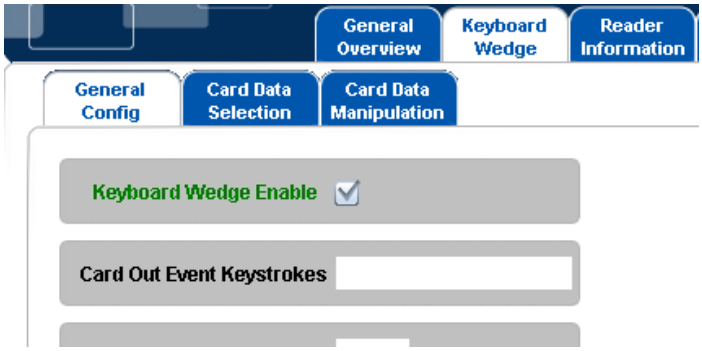

Figure 1 – System Configuration Changes

Note: Press Enter to finalize text field changes.

1.4 Save Changes

To save changes, select the **System Config** tab and click **Apply Changes**. The changed configuration parameters revert to black.

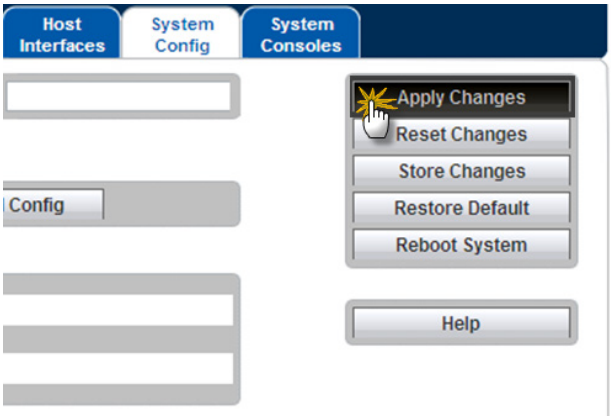

Figure 2 – Saving System Configuration Modifications

2 Keyboard Wedge Mode

The default configuration for the OMNIKEY 5x27 CK is **CCID** mode. Before using the Keyboard Wedge output, activate the Keyboard Wedge mode.

To enable the Keyboard Wedge mode, select the **Keyboard Wedge** tab and select the **Keyboard Wedge Enable** checkbox. Return to CCID mode by de-selecting the **Keyboard Wedge Enable** checkbox.

Note: When Keyboard Wedge is selected, the 5x27 CK enumerates as a Human-Interface USB device. Therefore, CCID interfaces are not be available. The web interface is available in both CCID and Keyboard Wedge modes.

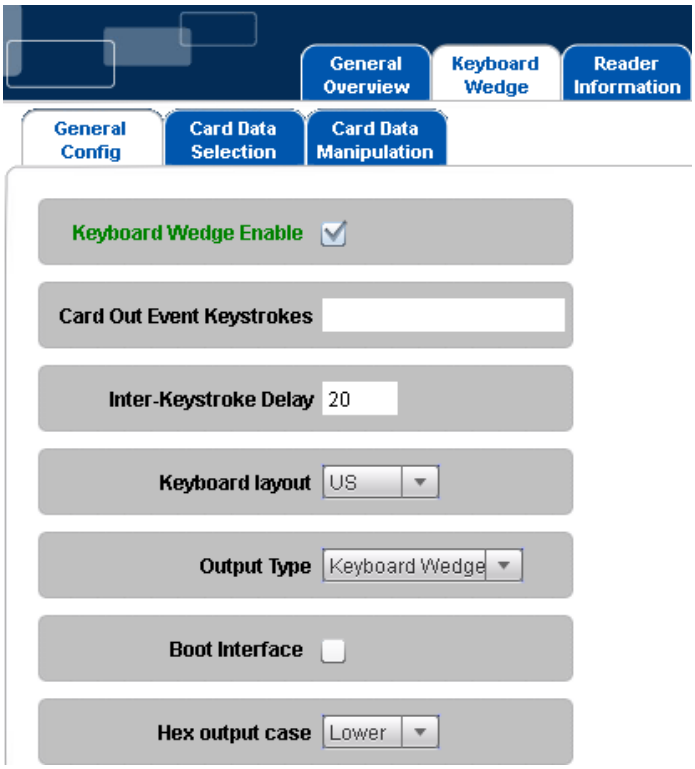

Figure 3 – Enabling Keyboard Wedge

Due to card data selection and manipulation, the keyboard wedge configuration options are split into separate tabs with firmware 03000000 or higher:

General Config: Keyboard wedge settings that apply to all card types

Card Data Selection: Identifies per card type what data fields shall be selected and sent to the host system via the keyboard wedge interface

Card Data Manipulation: Sets the data manipulation operations as well as output modes for each selected data field. This allows modifying the bitstream read from the card with various options (see later chapters for supported operations) before it is sent to the host system.

2.1 General configuration

2.1.1 Card Out Event Keystrokes

OK5x27 CK readers perform keyboard wedge actions on two events:

Card In: A card is detected by the reader

Card Out: A card is removed from the reader

Card Out defines a set of keystrokes that are sent when a card is removed from the reader. Due to the card removal from the reader, those keystrokes are generic (card-independent) and apply to all card types supported by the reader. If left open, no action is performed by the OK5x27 CK reader when a card is removed from the field.

2.1.2 Keyboard Layout

This selection compensates differences in regional keyboard layouts (for example, different interpretation of Y key on a US and DE keyboard). This setting must be adjusted to the actual setting of the host system in which the 5x27 CK is connected.

Example: A **Y** in the keyboard wedge layout **US** generates a **Z** on a host-PC using the German keyboard layout. Only when the keyboard wedge is configured to **DE** will the **Y** be interpreted correctly as a **Y** on the host-PC.

2.1.3 Output Type (firmware 03000000 or higher)

Keyboard wedge mode includes two output types, Keyboard Wedge and Custom Report.

The Keyboard Wedge output is the standard. The device enumerates as a keyboard and outputs the keyboard wedge data as a series of keystrokes.

In Custom Report output the device enumerates as a custom HID USB device. In this output the USB reports have improved to process the USB reports. The following describes custom report formats.

- The packet size is 40 bytes.
- 1st byte is the length of data in the packet.
- 2nd byte is the version of the report.
- The following bytes contain the keyboard wedge data.
- If the data length, version, byte length combine to less than the USB packet size (40 bytes), additional zeroes are added for the remaining length.

2.1.4 Boot Interface (firmware 03000000 or higher)

The Boot Interface option allows the device to advertise support for the keyboard boot interface in its HID device descriptor when it enumerates as a keyboard device. If enabled, the device is operational on host systems that only have minimal USB device handling, without support for full USB descriptor parsing.

2.1.5 Hex Output case (firmware 03000000 or higher)

The Hex Output case option specifies whether hexadecimal output is lower or upper case. The setting applies to all card types.

2.2 Card Data Selection Options

The Card Data Selections tab allows setting the keyboard wedge actions once a card is detected by the reader. Card-in events are customizable depending on the detected card type.

2.2.1 Card Types

The reader supports the following card types.

- iCLASS Seos (firmware 03000000 or higher)
- HID Prox (includes Indala Prox and EM4102 Prox - firmware 03000000 or higher)
- iCLASS (includes iCLASS SE)
- MIFARE Classic 1k/4k
- MIFARE UltraLight /C
- MIFARE DESFire
- MIFARE DESFire EV1 (MAC secured, DES/3DES, 3K3DES and AES encrypted - firmware 02000000 or higher)
- MIFARE Plus
- FeliCa (CSN only - firmware 03000000 or higher)

All supported cards are available for configuration in the **Card Type** dropdown menu on the **Card Data Selection** tab. Default configuration is that all card types are active and preset data fields are sent upon card detection.

Deselect cards through the web server by selecting the **Enable** button on each card page. The reader ignores deselected cards.

2.2.2 Card In Event Keystrokes

The Card In event defines a generic keystroke header that is sent upfront of any card data. This header is sent upon detection of the selected card type even when no card data is selected in the configuration.

2.2.3 Card Data Types

5x27 CK supports preset and custom data fields in Keyboard Wedge mode.

2.2.3.1 Preset Data Fields

Preset data fields represent the cards pre-configured data objects and for the 5x27 CK those are the PACS-Bits and CSN. Memory area, key configuration is preset in the 5x27 CK; therefore, no configuration is required to access those data fields.

PACS-Bits. The Physical Access Control System (PACS)-Bits are the data which grant access to a facility.

Note: PACS-Bits are only available for cards containing HID physical access credentials. Included are iCLASS Seos, HID Prox, iCLASS SE/ iCLASS SR, MIFARE (iCLASS SE) and DESFire EV1 (iCLASS SE).

iCLASS SE/ iCLASS SR credentials require firmware 02000000. iCLASS Seos credentials require firmware 03000000.

CSN. The Card Serial Number (CSN) is a data string which identifies your card uniquely.

Note: CSN is not available for HID Prox cards, since those cards do not provide an additional unique serial number besides PACS-bits.

2.2.3.2 Custom Data Fields

Custom data fields allow access to custom data stored anywhere in the card user memory. Therefore, the address + length of a custom data field, as well as the access key have to be configured prior to use. Memory structure, naming conventions and security measures are specific to card type, the web interfaces presents the required configuration input for the selected card type.

Note: For retrieving custom data, the corresponding access keys need to be available in the OMNIKEY 5x27 CK. Enter key references using decimal in the keyboard wedge configuration interface.

See the Software Developer Guide, Chapter 9 for key loading details. (www.hidglobal.com/main/developers/omnikey-5127-ck/)

Note: Offset and data length are defined as BYTE. In the example below OFFSET = 1, shifts the read zone by one byte and limits it to one byte:

Data on card (4 bytes total)
HEX 12345678
BIN 0001 0010 0011 0100 0101 0110 0111 1000

Output with OFFSET = 1, LENGTH = 1
HEX 34
BIN 0011 0100

For DESFire and DESFire EV1 cards with linear / cyclical record, set LENGTH to one, since it refers those cards to one record.

2.2.3.3 PACS Custom Data Fields (firmware 02000000 or higher)

HID credential physical access information is a unique bitstream that contains several data sections like Facility Code or Card Number. The pre-set data PACS function bits provide the full PACS bits stream. See Section 2.2.3.1 Preset Data Fields, page 9.

In case you are extracting only part of the full PACS bitstream, 5x27 CK readers provide the function PACS custom:

When activated, define and send separately up to three (3) data sections within the PACS bitstream over the Keyboard Wedge interface.

This option is available for card types provided with HID PACS bits (HID Prox, HID iCLASS, MIFARE Classic, MIFARE DESFire EV1) and requires Service Pack 1 or higher.

Definition of PACS data sections is done the same way as custom data fields (pre-/ post-strokes, Offset, Length). Since PACS data is typically not organized in full bytes, offset and length input represent bits (and not bytes as with custom data fields).

Furthermore, for each PACS sections, define the output type individually.

EXAMPLE: The configuration below defines two PACS sections:

- FACCODE starting at bit 0 with a length of 4 bits
- CARDNR starting at bit 7 with a length of 10 bits

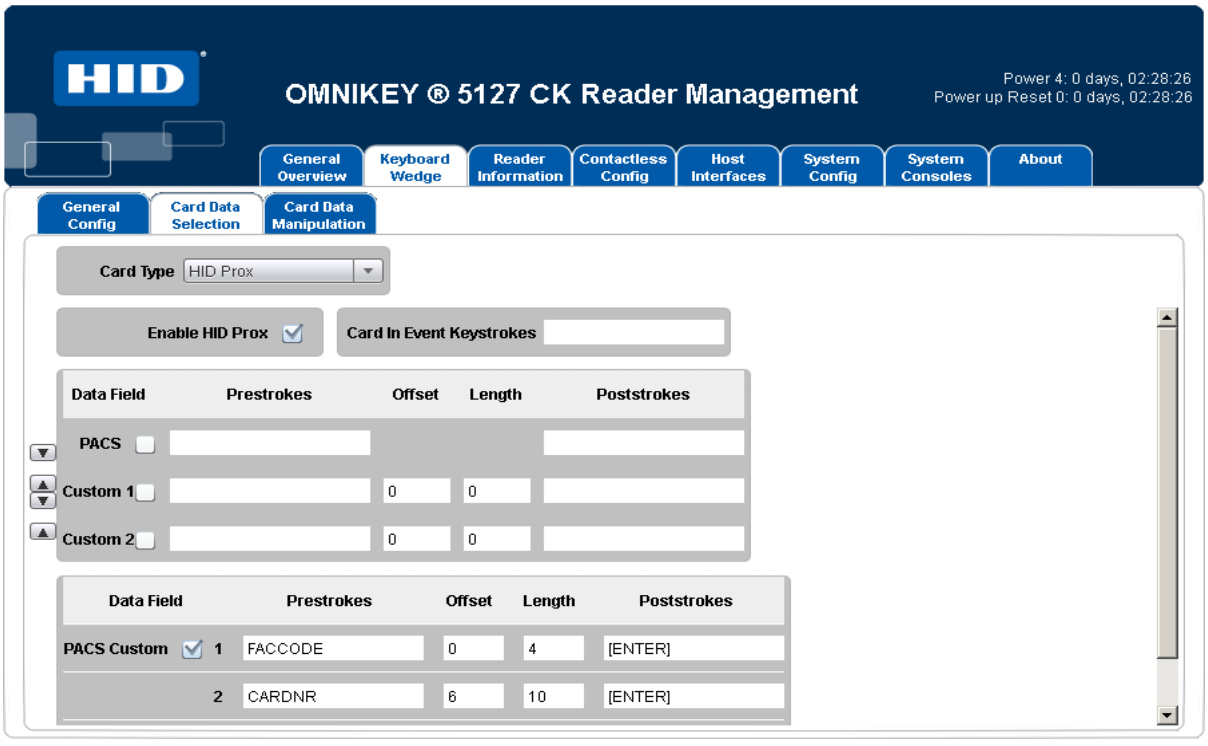

Figure 4 – Prox Card Custom PACS Card Data Selection Example

Assuming credential PACS bits are 1000000010000000000001001111, the keyboard wedge output is the following.

FACCODE section in BIN Output	1000
FACCODE section in DEC Output	8 (left-padding of 0's for full-byte conversion)
CARDNR section in BIN Output	0010000000
CARDNR section in DEC Output	128

2.3 Card Data Manipulation Options

2.3.1 PACS Leading Byte (firmware 03000000 or higher)

HID credential physical access information is not necessarily byte-organized (for example, HID 26 bit PACS formats). Output modes like ASCII, BCD, DEC and HEX do require full-bytes for data conversion from binary input.

This option sets the mode how non full-byte PACS data is converted to full byte stream before format conversion is performed:

- If un-selected (=default), leading zeros are added to any non-full-byte bit stream.
- If selected, trailing zeros are added to any non-full-byte bit stream. Furthermore, a leading PACS byte is added, indicating the number of bits the output is shifted right.

Example: Data on card (HID iCLASS H10302 format, 26 bits PACS bits)

0000 0101 0000 0010 0101 0010 01

Representation of PACS bits by 5x27 CK BIN output

In BIN mode the 5x27 CK performs no modifications to the PACS bits. The keyboard wedge output will therefore be:

00000101000000100101001001

Representation of PACS bits by 5x27 CK HEX, DEC, ASCII and BCD output with PACS Leading Byte not selected:

In HEX, ASCII, DEC and BCD mode the PACS bit stream will be modified to full bytes.

The data is shifted right by the number of unused bits.

For the example the 5x27 CK builds the following bit stream:

000000	000001010000001001010010 01
Unused bits	PACS bits

This modified bit stream is converted according to the selected output:

HEX **00140949**

ASCII **…I**

DEC **1313097**

BCD **0001 0011 0001 0011 0000 1001 0111**

Representation of PACS bits by 5x27 CK HEX, DEC, ASCII and BCD output with PACS Leading Byte selected:

The 5x27 CK adds trailing filler bits as well as a leading byte that states the number of unused bits in the last byte.

00000110	000001010000001001010010 01	000000
Header byte indicating the # of filler bits in last byte (= 6 bits)	PACS bits	Filler bits

This modified bit stream is converted according to the selected output:

HEX **0605025240**

DEC **25853841984**

2.3.2 Filtering (firmware 03000000 or higher)

Filter a byte (entered as decimal code) from raw data.

Direction: Leading = filter bytes from the start of raw data, Trailing = filter bytes from end of raw data.

2.3.3 Reverse (firmware 02000000, upgraded with firmware 03000000)

This option allows reversing the standard read order of the card data, it applies to custom data fields, PACS and CSN bits. The order is changed on byte-level according to the following example.

Card Data (HEX)	01 02 03 04
Reverse Byte Order output (HEX)	04 03 02 01

Note: In firmware 02000000, it applies only to custom data fields. PACS and CSN bits will not be affected by this command. From version 03000000 and higher, it applies to all data fields.

2.3.4 Padding (firmware 03000000 or higher)

Padding bytes are added to the output string.

Byte: ASCII character value (in decimal) to add to output string. It is output depending on the Format as specified above. So 48 would be output as 30 in hex or 0 in decimal. Binary is a special case, where only 0, 1, 48 or 49) are allowed – other values will be displayed as 1.

Direction: Leading = add padding to start of string, Trailing = add padding to end of string.

Length: Number of output characters to pad out to. This is format-independent, so entering 10 gives you 10 hex digits, 10 decimal digits, 10 ASCII characters, 10 binary bits, etc.

2.3.5 Format

The following output types or formats are supported.

BIN — **Binary Representation**

The defined read area bit stream is sent to the Host system as 0 and 1 key strokes the same way as how they are stored on the card (there are no leading or trailing bits/keystrokes added).

EXAMPLE:

Data on card (4 bytes): 0001 0010 0011 0100 0101 0110 0111 1000

BIN Keyboard Wedge Output: 0001 0010 0011 0100 0101 0110 0111 1000

HEX — **Hexadecimal Representation**

The defined area bit stream is sent as 0-F keystrokes to the Host system according to the HEX representation of the bit stream.

EXAMPLE:

Data on card (4 bytes): 0001 0010 0011 0100 0101 0110 0111 1000

HEX Keyboard Wedge Output: 12345678

ASCII — **American Standard Code for Information Interchange Representation**

The defined area bit stream is sent as ASCII keystrokes to the Host system according to the ASCII representation of the bit stream. Non-printable characters (for example, ACK) are substituted by a period (.).

EXAMPLE:

Data on cards (4 bytes): 0001 0010 0011 0100 0101 0110 0111 1000

ASCII Keyboard Wedge Output: .4Vx

BCD — **Binary Coded Decimal Representation**

The defined area bit stream is sent as 0 and 1 keystrokes to the Host system according to BCD representation of the bitstream.

BCD converts each byte of the bitstream to a decimal number.

Since the resulting decimal number can be in the range of 0-255, the 5x27 CK reserves three digits for each number, a leading 0 is added.

Each decimal number digit is converted into 4 bits.

EXAMPLE:

Data on card (4 bytes): 0001 0010 0011 0100 0101 0110 0111 1000

(Decimal number: 305419896)

BCD Keyboard Wedge Output: 0011 0000 0101 0100 0001 1001 1000 1001 0110

DEC — **Decimal Coded Representation (available with firmware 02000000 or higher)**

The defined area bit stream is sent as 0-9 keystrokes to the Host system according to decimal representation of the bitstream.

Maximum bitstream length to be converted is 8 bytes.

EXAMPLE:

Data on card (4 bytes): 0001 0010 0011 0100 0101 0110 0111 1000

DEC Keyboard Wedge Output: 305419896

2.4 Building an Output String

2.4.1 Card In Event

The 5x27 CK lets you customize your output string for a Card In Event; the following objects are available for configuration on the **Card Data Selection** tab.

Card in Event Keystrokes	Option to enter header information to an output string.
Data Fields	Select either the cards preset or custom data field.
Pre-strokes	Keystrokes sent before the data field.
Post-stroke	Keystrokes sent after the data field.

You can have multiple data fields in one output string (for example, PACS bits followed by a custom data field). In this case, ensure the desired data fields are activated and fully configured.

Change the order of the output string data fields by using the up/down arrow buttons (left of the data field names).

Separate data fields from each other by using pre- and post-strokes.

2.4.2 Card Out Event

The 5x27 CK lets you define an output string to be sent when a card is taken from the reader.

Note: This output string is sent for each card type and does not support card data.

2.4.3 Supported Characters & Commands

In most cases, keyboard stroke data (Pre and Post, or both) are strings of standard ASCII characters. In addition, use **control** characters, such as the **Enter** key. Enclose the control character (key) in brackets [ ], for example, [ENTER].

IMPORTANT:

For confirming post- or pre-keystrokes in firmware versions below 02000000, press 

, for the reader to perform validity check on the keystrokes.

For firmware versions 02000000 or above, pressing 

is not required, the reader performs a validity check automatically once the focus is taken from the data field (for example, by pressing the **Tab** key or clicking another data field).

For valid keystrokes, the font color turns from black to green. The text color remains green until you click **Apply Changes** and the settings are then loaded to the reader (see Section 1.4 Save Changes, page 6).

In case the validity check fails, the font color turns red.

Possible failures include the following.

- Wrong syntax in control commands
- Exceeding max length per data field - seven (7) characters

The following table lists all supported control characters.

Control characters have to be capital letters.

Combine keystrokes with ASCII characters to allow shortcuts on the computer. For example, **`[ALT]F[CTRL]N[ENTER]`** creates a new text file when the Notepad application is active on the computer.

Table 1 – Supported Control Characters

Control Character / Key	Abbreviation
End	END
Enter	ENTER
Esc	ESC
Cursor down	DOWN
Cursor up	UP
Cursor left	LEFT
Cursor right	RIGHT
Space	SPACE
Tab	TAB
F1	F1
…	…
F12	F12
Shift	SHIFT
Ctrl	CTRL
Alt	ALT
Delete	DEL
Windows	GUI

3 Card Configuration Examples

3.1 Example 1 – Reading iCLASS Card PACS Data

1. Enable **Keyboard Wedge** mode, see Figure 3 – Enabling Keyboard Wedge.
2. Select the **Keyboard Wedge** tab and select the **Card Data Selection** tab.
3. From the **Card Type** drop-down menu, select **HID iCLASS**.
4. Click the **Enable HID iCLASS** check box.
5. Click the **PACS** checkbox.
6. In the **PACS** Pre-strokes text field, enter **Start**.
7. Press Enter

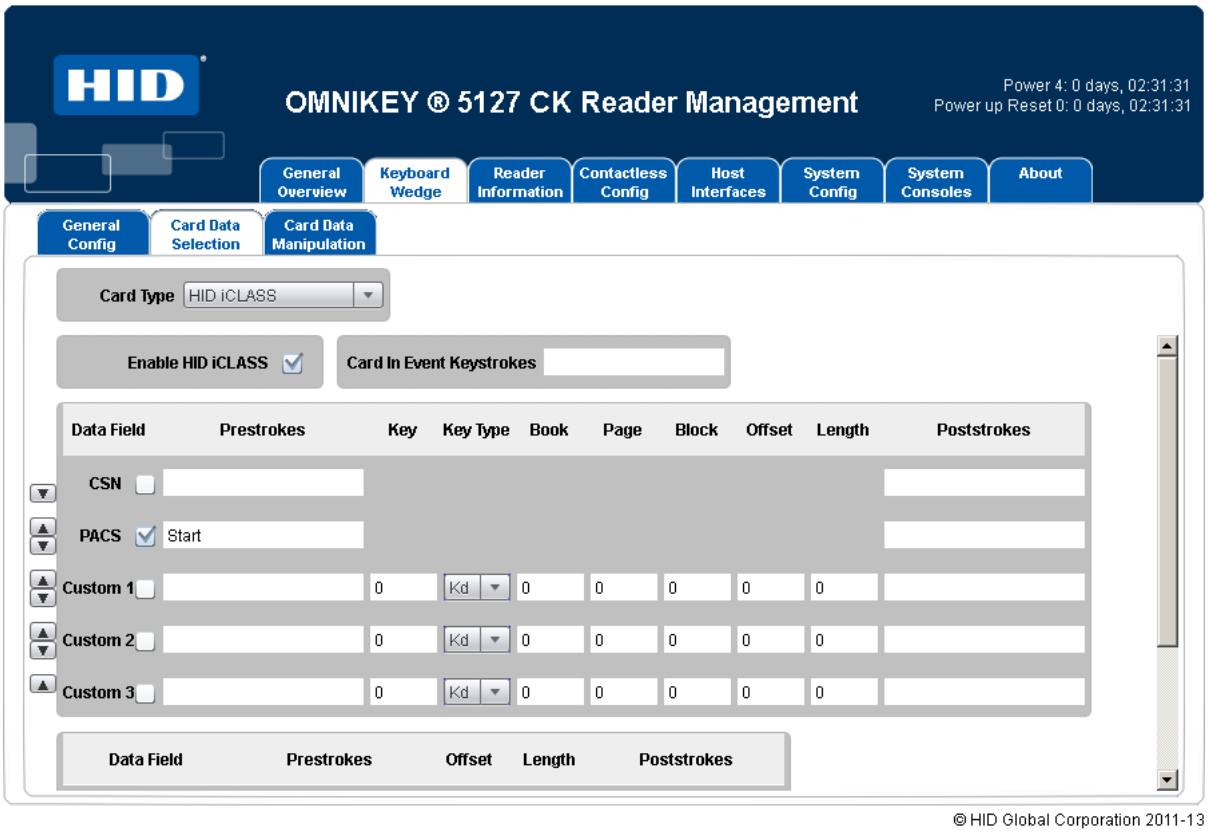

Figure 5 – iCLASS Card PACS Data Example

8. Open a text editor and place the iCLASS Sample card into the RFID field over the antenna of the reader.

9. The Keyboard Wedge enters into the editor the word **Start** followed by the PACS data in hexadecimal format. Example:

   Startf6e1b500f9ff12e0

3.2 Example 2 – Reading MIFARE Card CSN

1. Go to the **Keyboard Wedge** tab and select the **Card Data Selection** tab.
2. From the **Card Type** drop-down menu, select **MIFARE Classic**.
3. Click the **Enable MIFARE Classic** check box.
4. Click the **CSN** checkbox.
5. Enter **Start** into the Pre-strokes text field, press 

.
6. Enter **End** into the Post-strokes text field, press 

.

Figure 6 – MIFARE Card CSN Example

7. Open a text editor and place the MIFARE 1k Sample card into the RFID field over the antenna of the reader.
8. The Keyboard wedge enters into the editor the word **Start** followed by the CSN data in hexadecimal format and the word **End**.

Example:

Start70c0ad38End

3.3 Example 3 – HID iCLASS PACS Data Filtering

1. Go to the **Keyboard Wedge** tab and select the **Card Data Selection** tab.
2. From the **Card Type** drop-down menu, select **HID iCLASS**
3. Click the **Enable HID iCLASS** checkbox.
4. Click the **PACS** checkbox.
5. Enter **<pacs>** into the Pre-strokes text field, press Enter.
6. Enter **</pacs>** into the Post-strokes text field, press Enter.

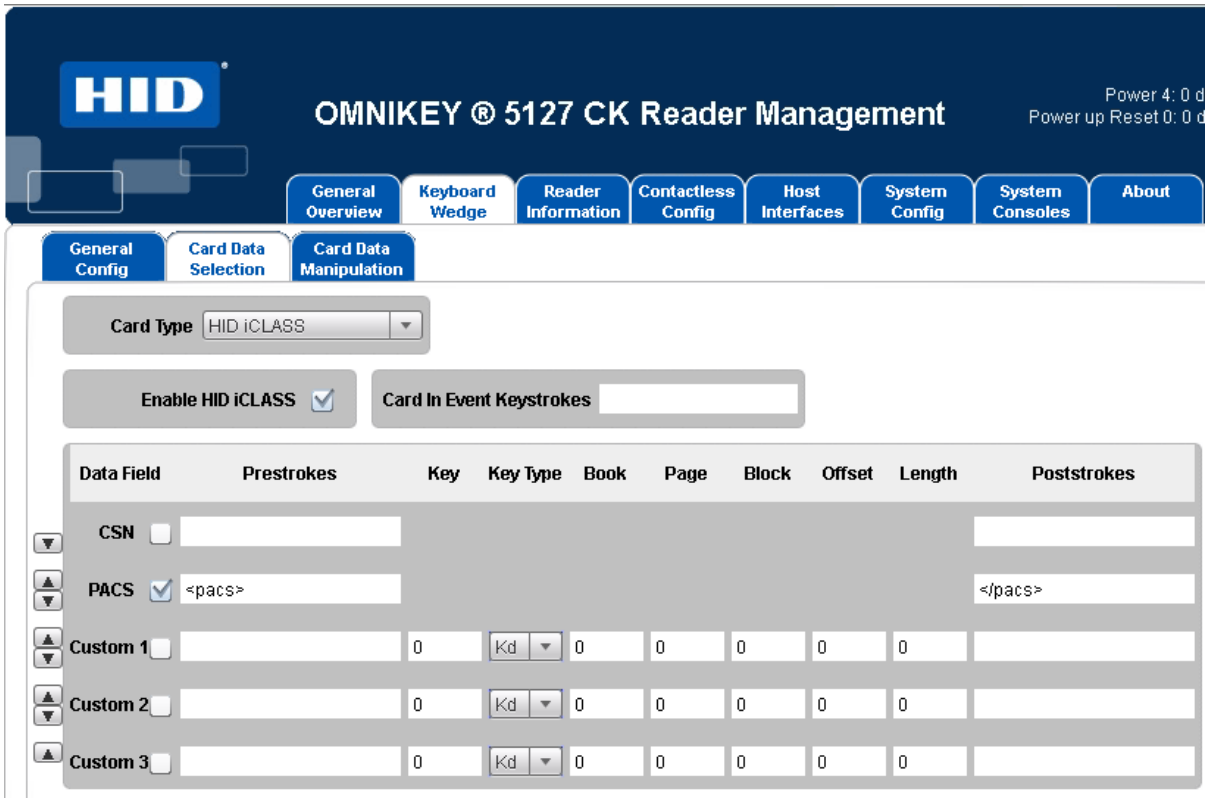

Figure 7 – HID iCLASS PACS Filtering Card Data Selection Example

7. Now select the **Card Data Manipulation** tab.
8. Click the check box in the PACS row of **Filtering** box.
9. Make sure **HEX** is selected in PACS row of **Format** box.

10. Enter 246 (decimal for f6) in the **Byte** field on the PACS row of the **Filtering** box.

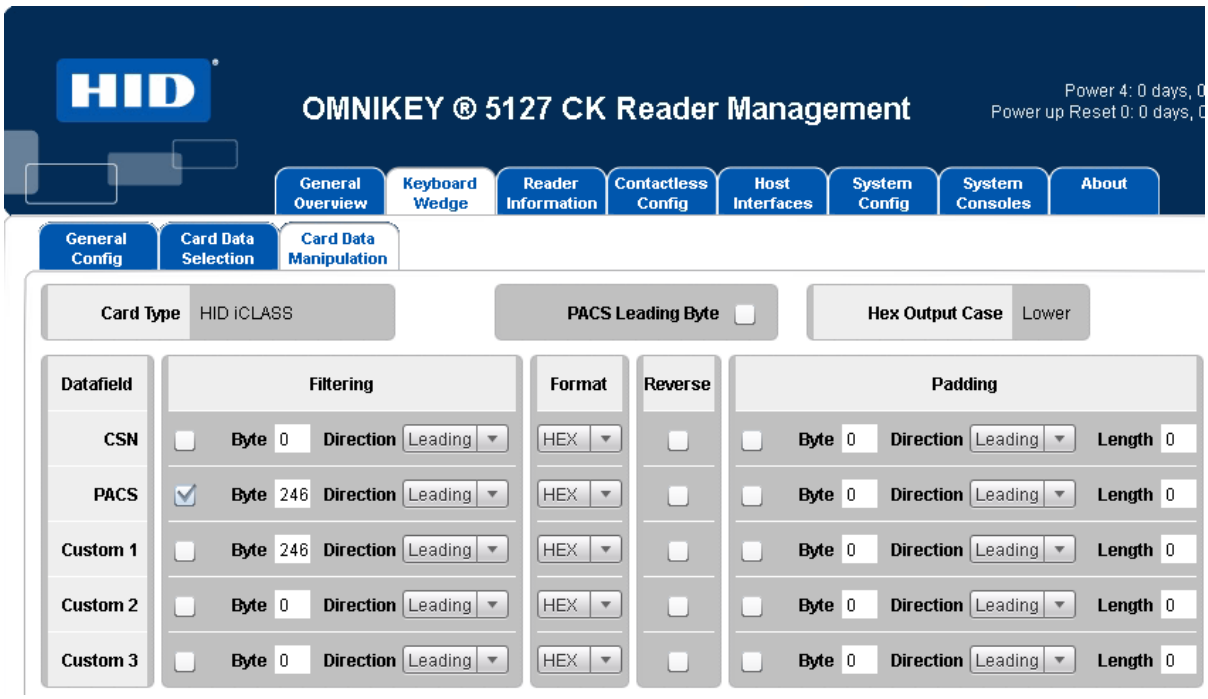

Figure 8 – HID iCLASS PACS Filtering Card Data Manipulation Example

11. Open a text editor and place the iCLASS Sample card into the RFID field over the antenna of the reader.
12. The Keyboard Wedge enters into the editor the text <pacs> followed by the filtered PACS data in hexadecimal format followed by the text </pacs>.

Example:

<pacs>e1b500f9ff12e0**</pacs>**

Note the byte f6 (= 246) has been filtered out.

3.4 Example 4 – Prox Card PACS Data Padding

1. Go to the Keyboard Wedge tab and select the Card Data Selection tab.
2. From the Card Type drop-down menu, select HID Prox.
3. Click the Enable HID Prox check box.
4. Click the PACS checkbox.
5. Enter <pacs> into the Pre-strokes text field, press Enter
6. Enter </pacs> into the Post-strokes text field, press Enter

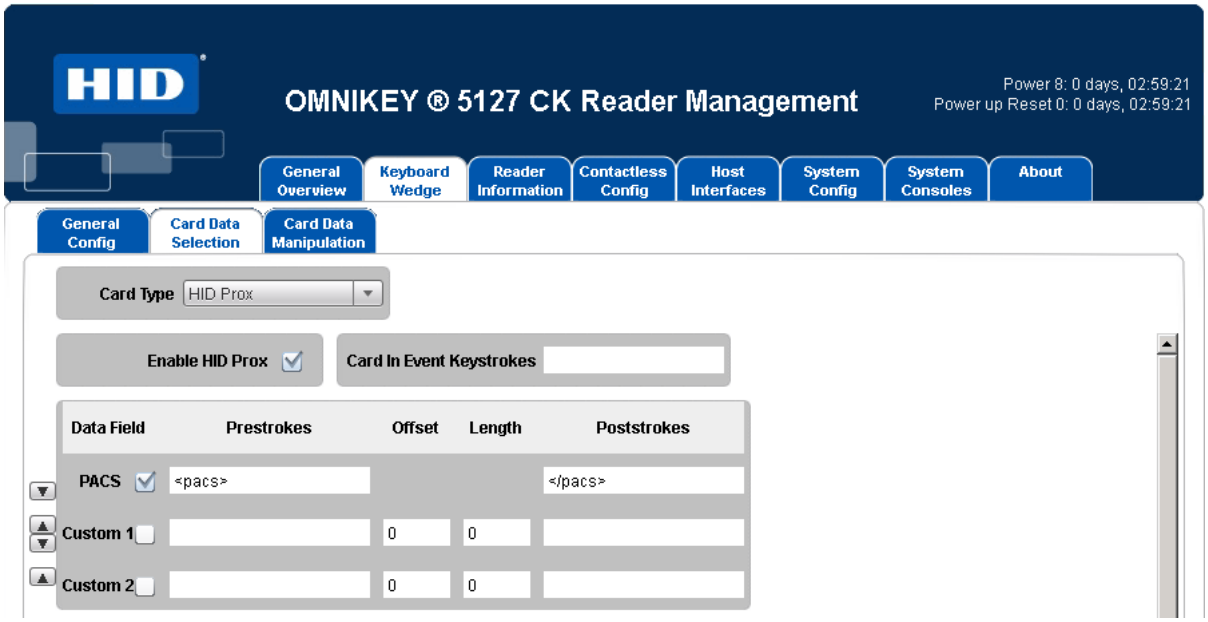

Figure 9 – Prox Card PACS Padding Card Data Selection Example

7. Select the Card Data Manipulation tab.
8. Select HEX in the PACS row of the Format box.
9. Click the check box in the PACS row of the Padding box.
10. Enter 255 in the Byte field in the PACS row of the Padding box.
11. Enter Leading in the Direction field in the PACS row of the Padding box.
12. Enter 20 in the Length field in the PACS row of the Padding box.

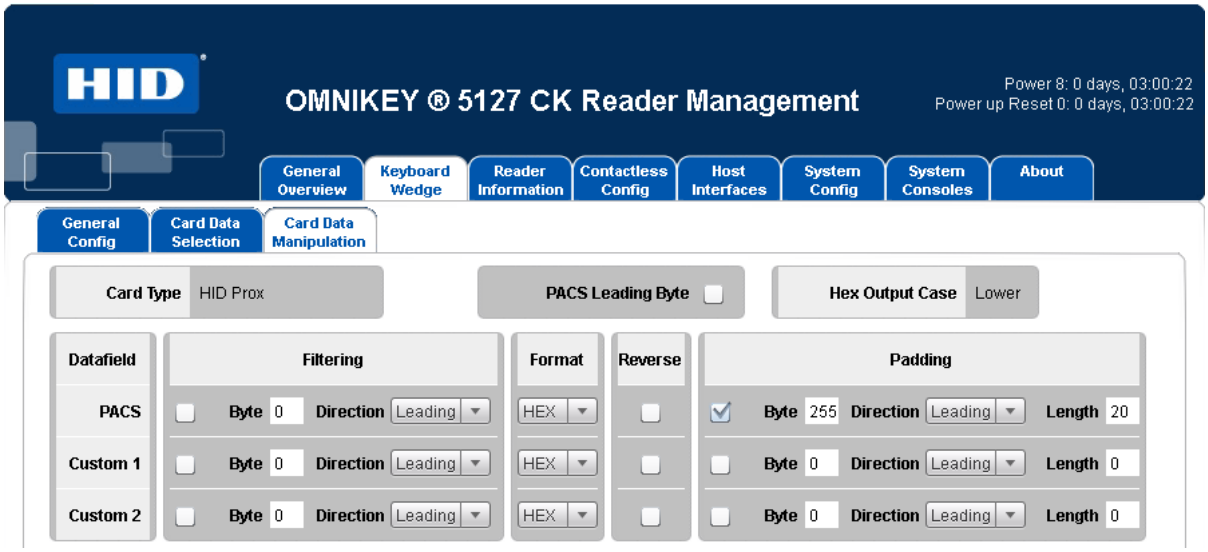

Figure 10 – Prox Card PACS Padding Card Data Manipulation Example

13. Open a text editor a place an HID Prox card into the RFID field over the antenna of the reader.
14. Assuming the data on the card is 10000000100000000001001111, the output in the editor will be:

```
<pacs>ffffffffffff0202004f</pacs>
```

4 Glossary

Abbreviation	Description
CSN	Card Serial Number
CCID	Integrated Circuit(s) Cards Interface Device
EEM	Ethernet Emulation Mode
USB	Universal Serial Bus
PACS	Physical Access Control System

HID Global Headquarters:

North America: +1 949 732 2000

Toll Free: 1 800 237 7769
Europe, Middle East, Africa: +49 6123 791 0

Asia Pacific: +852 3160 9800

Latin America: +52 477 779 1492

support.hidglobal.com