Smart Toolz

Smart Card Development Kit

User Guide

CardLogix

NOTICES

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of CardLogix Corporation. CardLogix Corporation shall not be held liable for technical and editorial omissions or errors made herein; not for incidental, or consequential damages resulting from the furnishing, performance or use of this material. CardLogix Corporation reserves the right to revise this publication and to make changes from time to time in its content without obligation of CardLogix Corporation to notify any person or organization of such revision or change.

Trademarks

CardAppz®, M.O.S.T. Toolz®, and Smart Toolz® are trademarks of CardLogix Corporation. Winplex® and Printplex® are registered marks of CardLogix Corporation. All terms used in this document that are known to be trademarks or service marks have been capitalized where appropriate. CardLogix Corporation cannot attest to the accuracy of this information. Use of a term should not be regarded as affecting the validity of any trademark or service mark.

General Notice

Some of the product names used herein have been used for identification purposes only and may be trademarks of their respective companies. Microsoft Windows 7, 2000, XP, and Vista are all registered trademarks of the Microsoft Corporation.

FCC

This equipment has been tested and found to comply with the limits for a class B digital device, pursuant to part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Statement of Electromagnetic Compliance

This product has passed all electromagnetic interference and susceptibility testing required by the European Community and thus bears the "CE" mark.

- This product has passed all electromagnetic interference and susceptibility testing required by the European Community and thus bears the "CE" mark.
- This Class B Digital Apparatus meets all requirements of the Canadian Interference Causing Equipment Regulations.

CARDLOGIX CORPORATION SOFTWARE LICENSE AGREEMENT AND HARDWARE AGREEMENT FOR SMART TOOLZ DEVELOPMENT KIT

1) This Agreement is made by and between you (either an individual or an entity) and CardLogix Corp.

TITLE

2) Subject to any rights deriving from any patent (either registered and/or pending and/or that can be registered). CardLogix and/or any third party hold or shall hold in the hardware (i.e. all the tangible elements of the PRODUCT) or any part thereof, CardLogix sells you the HARDWARE.

3) THE SOFTWARE CONTAINED IN THE PRODUCT (including any revisions, improvements and/or updates) and RELATED DOCUMENTATION (necessary and/or related to the use of the PRODUCT, in hard copy and/or electronic form) IS NOT FOR SALE; title in and to the SOFTWARE & DOCUMENTATION shall remain solely with CardLogix.

LICENSE

4) CardLogix grants you a non-exclusive right to use the PRODUCT for the sole purposes of developing smart cards and smart card applications based exclusively on CardLogix cards. The SOFTWARE & DOCUMENTATION shall not be used for any other purposes.

5) You are permitted to make one (1) copies of the SOFTWARE solely for development and backup purposes.

6) You may alter, merge, modify or adapt the Winplex® SOFTWARE but only in order to "and to the extent required" to merge it with your smart card application. You may not disassemble and/or decompile and/or reverse engineer the SOFTWARE or any part thereof. You have a royalty - free right to reproduce and distribute binary executable files which include some or all of the Winplex SOFTWARE, only in binary executable form, provided that the binary executable files do not constitute an application that may compete with and/or imitate and/or substitute the SOFTWARE. The media containing the binary executable files displays your copyright notice, and the title page or title page version of the documentation which accompanies the binary executable files contains the following CardLogix copyright notice:

   "Portions Copyright 1997 - 2010, CardLogix Corp. All Rights Reserved."

   Within forty five days of your first transfer or shipment of said application to a third party, you shall send CardLogix a copy of the CD label or title page containing the CardLogix copyright notice specified above. CardLogix will acknowledge receipt of this copy.

7) In the event that you cannot satisfy the conditions of the above paragraph, contact CardLogix regarding an amendment to this Agreement.

8) You may not rent, lease, loan, or sub-license the PRODUCT or any part thereof. You may not transfer your rights under this Agreement to any party.

9) Apart of the right of use explicitly granted herein, you shall have no other rights, express or implied, in the SOFTWARE.

PROPRIETARY RIGHT

10) All intellectual property rights in the SOFTWARE & DOCUMENTATION and some of intellectual property rights in the HARDWARE are owned by CardLogix and are protected by its copyright, patent, trademark, trade name and trade secret laws and international treaty provisions. You agree that the SOFTWARE is the proprietary property of CardLogix and that it is distributed subject to restricted disclosure only and that the license does not convey to you an interest in or to the SOFTWARE, but only grants you a limited right of use in accordance with the terms of this Agreement.

11) The HARDWARE is provided "AS IS" and CardLogix shall have no liability to you for the infringement of any patents, copyrights, trade secrets or other proprietary right by the HARDWARE or any portion thereof.

LIMITED WARRANTY & LIMITATION OF LIABILITY:

12) CardLogix warrants that (a) the SOFTWARE will perform substantially in accordance with the accompanying user documentation for a period of 1 year from the date of receipt and (b) that the HARDWARE, under normal use and service, is free from defects in materials and workmanship for a period of 1 year from the date of receipt. Any implied materials and workmanship are warranted for a period of 1 year from the date of receipt. Any implied warranties on the SOFTWARE and HARDWARE are limited to 365 days and/or one (1) year, respectively. The foregoing warranty shall not apply to consumable portions of the HARDWARE which are expendable by nature. Some states/jurisdictions do not allow limitations on duration of an implied warranty.

13) CardLogix's entire liability and your exclusive remedy shall be at CardLogix's option either (a) return of the price paid or (b) repair or replacement of the SOFTWARE or defective HARDWARE that does not meet CardLogix's Limited Warranty and which is returned to CardLogix, with a copy of your receipt, within 90 days of the date of receipt. This Limited Warranty is void if failure of the SOFTWARE or defective HARDWARE has resulted from: (i) accident, abuse or misapplication and/or Modifications are made to the PRODUCT by anyone other than CardLogix; (ii) attachments, features or devices that are employed on the HARDWARE which are not supplied by CardLogix or not approved for use, in writing, by CardLogix; (iii) or other than the current version of the SOFTWARE available from CardLogix is used on the HARDWARE. The warranty and remedies set forth herein are exclusive and in lieu of all others, oral or written, express or implied. No CardLogix dealer, distributor, agent or employee is authorized to make any modification or addition to this warranty, save that nothing in it affects any rights you may have against us for death or personal injury caused by our negligence.

14) Except for and to the extent expressly provided herein, CardLogix makes no warranty or representation, either expressed or implied, with respect to the PRODUCT, including its quality, performance, merchantability or fitness for a particular purpose.

15) ***Please note that the SOFTWARE is inherently complex and may not be completely free of errors.*** To the degree permitted by applicable law, in no event shall CardLogix be liable for direct, indirect, special, incidental, cover or consequential or any other damage whatsoever (including, without limitation, damages for loss of business information or other pecuniary loss) arising out of or related to this Agreement or the performance or breach of CardLogix's liability and/or the use of or inability to use the PRODUCT, even if CardLogix has been advised of the possibility of such damages. In no case shall CardLogix liability under any provision of this Agreement exceed the amount actually paid by you for the PRODUCT. To the extent that applicable law does not allow the exclusion or limitation of implied warranties or limitation of liability for incidental or consequential damages, the above limitation or exclusion may not apply to you.

GENERAL

16) This Agreement is governed only by the laws of the State of California and only the courts in California shall have jurisdiction in any conflict or dispute arising out of this Agreement.

17) Any cause of action arising out of or related to this Agreement must be brought by you no later than one year after the cause of action has occurred.

18) This License Agreement is effective upon your opening the accompanying Package and shall continue until terminated. You may terminate this license Agreement at any time. Upon the breach by you of any term or condition of this License Agreement, and any subsequent failure to correct the breach within fourteen days of notification of the breach, CardLogix may terminate this License Agreement. Upon termination of this License Agreement by you or CardLogix, you agree to immediately return the SOFTWARE to CardLogix, to continue to maintain the SOFTWARE confidential, and to immediately destroy all copies of the SOFTWARE, whether in whole or in part, whether modified or not, whether in source object or binary executable.

COPYRIGHT

DISCLAIMER

Although CardLogix endeavors to ensure the best possible product quality, CardLogix does not warrant that the Smart Toolz®, MIFARE®, and M.O.S.T. Toolz® Software will function properly in every hardware and software environment. The Smart and M.O.S.T. Toolz® Software may not work in combination with some networks and some programs. It may not work with modified versions of the operating system or with specific patches. The Smart & M.O.S.T. Toolz® Software may not function properly with certain modems and/or certain printers. Not all supported devices will work with all operating systems.

Table of Contents

📋 Section I: Introduction

Welcome to Smart Toolz® Smart Card Development Kit

With this kit, you will find everything you need to build a smart card system based on memory and contactless cards and to test out your ideas. The items included in your kit are production items and can be purchased in volume separately through CardLogix. The design environment is setup primarily for the storage of records and information. If you are building a stored value system that will require a wide distribution of cards, please call CardLogix sales at +1 949 380 1312 to discuss the specific issues relating to printing, terminals, security and card and key issuance.

How to Use This Manual

The CardLogix Smart Toolz® User Manual is divided into sections to help you use your Smart Toolz® as quickly and efficiently as possible. The sections include simple explanations and easy-to-follow steps that will help you understand the installations and functions of your Smart Toolz® kit. The guide assumes you are generally familiar with how to use a personal computer and are able to identify your USB, COM, and serial ports.

Smart Toolz® was designed for a variety of applications and for a variety of users with different skill levels. In order to help all users identify the sections of most interest to them we have employed the following iconography. When you see the 📋 icon, it is an item for the general use of the system. When you see the ⌨ icon, the information is focused for programmers.

- 📋 **Section II: Getting Started** – Provides information about your Smart Toolz® kit. Included in the kit are the installation instructions and your Warranty Registration Card.
- 📋 **Section III: Planning - New to Smart Cards:** A general guide to smart card program development.
- 📋 **Section IV: CardAppz® Overview** – An explanation of the use and implementation of a card system with CardAppz®. This is the recommended approach if you have no programming experience.

- **Section V: Card Configuration Utility (CCU)** – Allows developers to complete their smart card projects through a program interface in a simplified design, test, and produce method.
- **Appendix A: ISO 7816 Error Codes** – Describes error codes that can be returned from each type of smart card included in this kit.
- **Appendix B : Memory Address Range** - A table listing the address ranges for the various card types.
- **Appendix C: Reader Command Matrix** - Tables that cross-reference commands and particular reader types.
- **Appendix D: Glossary** – Lists commonly used terms in the smart card industry.

Section II: Getting Started

Check the Contents of Your Smart Toolz® Kit

Carefully remove the contents of the Smart Toolz® kit. To protect your Toolz when in storage, save the box and all packaging materials.

☑ Software License Agreement (second page of this manual)

☑ USB Contact-Contactless smart card reader/writer

☑ CD-ROM containing software, PDF documentation and examples

☑ Smart Toolz ® User guide

☑ Winplex® User Guide

☑ 10 CardLogix sample cards, and technical briefs on each

☑ CardLogix Warranty Registration Card (first page of this manual)

Minimum System Requirements

To run Smart Toolz®, your computer must have a Windows 7, Vista, or XP SP3 operating system running on a Pentium® PC with at least 128 megabytes of RAM and an optical disc drive.

Recommended: A Pentium® 2.4 GHz system or higher running Windows Vista© or Windows XP© SP3 with at least 256 megabytes of RAM and an optical disc drive.

Installing Smart Toolz®

1) Make sure to read and accept the terms of the Software License Agreement.
2) Start Windows.
3) Insert the program CD into your CD-ROM drive.
4) Double click the Setup icon or, from the Windows program manager, select Run from the File menu.
5) In the command line dialog box, type “d:\setup”, where d: is the letter of your CD-ROM drive, and click OK.
6) Follow the on-screen instructions.

The setup program will copy the required files to your system and create the necessary program groups. To re-install the Winplex® API in another location, go to the Smart Toolz® installation directory, click on the individual folder and move the DLL's to your working location.

Troubleshooting

If you are experiencing difficulty with any of the Smart Toolz® software or hardware please contact us at www.cardlogix.com/corporate/contact.asp or send an e-mail to sales@cardlogix.com describing the problem.

When contacting us regarding problems please make sure you have the following information available:

1) Smart Toolz® serial number (found on the inside lid of your box).
2) Specify your operating system and system information.
3) Specify the card P/N that is being used.
4) Specify the DB# that is relevant.
5) Describe the problem in detail.

THE Smart Toolz® WARRANTY COVERS ONLY FUNCTIONAL DEFECTS, NOT IMPLEMENTATION DIFFICULTY! (See warranty for full details.)

Returns

Please contact CardLogix regarding any and all merchandise that is not performing to the published specifications. In the event of a defect in material or workmanship during the warranty period, CardLogix, at its discretion, will repair or replace the defective product after the defective product is returned to CardLogix by the owner.

For return of product, you must be issued, by CardLogix, a Return Material Authorization (RMA) number. We cannot issue an RMA Number unless you have registered your system. This RMA Number is to be included in any returned merchandise. Please mark the RMA Number on the box and on the packing slip.

You are responsible for the shipping charges when returning items to CardLogix for quality review. If the items are found to be defective and within the warranty terms and/or period, CardLogix will pay for all shipping charges (company policy is UPS 2nd day air) returning the devices to you.

Section III: Planning – New to Smart Cards

Summary

This section is divided into two parts:

Getting Started

This first section contains information about your Smart Toolz® Software Development Kit that will help you use it more efficiently. When appropriate, we've included links to CardLogix web pages related to the topic, for more information on our products and services.

System Planning

This section helps you plan, design, and implement your system with some general guidelines. These guidelines are meant for general reference only, and do not cover every possible design step and contingency. Additional information can be found at smartcardbasics.com and smartcardalliance.org.

Planning a card system can be a daunting task. CardLogix tries to make this task easier by providing an evolutionary path for your implementation with our products. At the beginning of this journey you will need to understand the issues around your business requirements in the context of cards and the ecosystem that they will interact in.

For your starting reference, you may need the following documents:

- ISO 7816-1,2,3 specification at a minimum
- ISO 14443 for contactless cards
- CardLogix Specifications or Tech briefs for the card types that you will use
- The CardLogix Graphics Design and Security Printing Guide—This is an overview of graphics and printing to enhance card use and/or security. It is included in the Smart Toolz® documentation and is also available on the CardLogix site at http://www.cardlogix.com/support/documentation.asp

- CardLogix Specifications or Tech briefs for the readers or terminal types that you will use
- The supplied Winplex® User guide
- The supplied Smart Card Basics publication

When writing directly to the memory of a semiconductor, the specifications for each chip are very helpful. Due to confidentiality agreements, it is best to get these directly from the manufacturer.

To purchase the ISO specifications please visit www.iso.org for the latest documentation on ISO Standard 7816 and ISO 14443, which govern the design of contact and contactless smart cards.

To learn more about card and card reader choices, please visit www.cardlogix.com/products/cards/smart/scfamilies/ and www.cardlogix.com/products/readers/.

To begin, we recommend that you start with the CardAppz® software demonstration application.

IMPORTANT NOTE - ASSIGNING CARD TYPES: While using CardAppz® or the Card Configuration Utility (CCU), found in the Card Toolz program group, at the step when you select a card reader, you must specify either 'contact' or 'contactless' to correspond with the type of card you are developing with. The supplied reader and software does not automatically select card type. Also, if you are switching between applications you will have to reset the reader back for each program.

CardAppz®

CardAppz® helps build a card design, allowing you to demonstrate card capabilities within a completely configurable card database and system. CardAppz® enables you, the designer, to select templates, set security levels, and assign user types, from casual to power levels. This might be a design idiom for your project. This also allows you to check that the reader and cards are working as expected.

Card Configuration Utility (CCU)

This software gives designers the power to configure card parameters, load data into cards, and communicate with cards through the supplied reader. It is designed for the professional programmer. The CCU helps you visualize how data is displayed and configured in the card.

Winplex® API

The Winplex® library includes provided sample code for reading and writing to a card and commands for managing many different reader types. With Winplex®, you can hook the card and reader into your software application quickly. The supplied sample executable and code enables code visualization for a variety of commands.

System Planning

Smart card system design requires advance planning to be successful and to avoid problems. It is highly recommended that you graphically diagram the flow of information for your new system. The first question to consider is 'Will the card and system transact information, or value, or both?' If it stores keys or value (i.e.; gift certificates or sports tickets), greater design detail is required than in data-only systems. When you combine information types on a single card, other issues arise. The key to success is not to overrun the system with features that can confuse users and cause problems in management. We recommend that you phase-in each feature set as each one is working. To properly implement a functional smart card system you should be able to answer the following questions.

NOTE: *These are only general guidelines, provided as a basis for your individual planning. Many other steps may be involved and are not mentioned here. For more extensive planning information regarding identity management and national IDs we recommend that you may want to review the GSA Smart Card Handbook. For financial card issuance we recommend the Global Platform Publications. If you have any questions, feel free to call your CardLogix sales representative and our team will get you pointed in the right direction.*

Card Choices - Considerations

The choice of card types is but one of many decisions regarding a card system. It is a choice that should not be based on card cost or performance alone. The total cost of ownership over the cards life weighed against performance is a better metric. The choice of contact vs. contactless cards should be based on:

- Transaction Speed and user environments
- Reader cost and use (contact readers are typically good for 250,000 insertions)
- (Contactless readers are typically 2x the cost)
- Data movement (contactless cards cannot effectively carry large data payloads)
- On contactless cards, data storage is typically 2x the use for dynamic data due to backup requirements

The next set of variables for consideration is file based cards vs. direct memory access. Direct memory access is usually the lowest cost per bit in the short term, but can have a cost or upgrade implications over the life of a program. By abstracting the card data into files that can be represented on many different card structures; a system will be less dependent on a single device or supplier and can be future proofed to a reasonable extent.

Some types of physical security features are more difficult to execute on a contactless card construction. Features such as colored interlayers and clear card bodies make construction costly and difficult. Also specific foils and holomag stripes can interfere with the antenna. Generally all other security features are available for both styles of cards.

The First Four

1) Do you require a completely original design? Or is there an existing application that you can use? (For the latter, please visit CardLogix Smart Partners at www.cardlogix.com/smartpartners/apply.asp.)
2) Is there a clear business case? Does it include financial and consumer behavior factors?

3) What will the card be required to do? What are the card's essential features? With multiple functionalities, prioritize, starting with the most important one and phase in additional features incrementally.

4) Will the smart card handle data, value, or both? Adding a value function increases system design security and complexity.

Basic Setup

1) Will the system be single- or multi-application?

2) Are there industry standards to conform to ISO, EAL, or ETTSI for specific encryption or chip requirements?

3) What information do you want to store in the cards?

4) How much memory is required for the applications?

5) If the system is multi-application, how will you separate different types of data?

6) Will data be obtained from a database or loaded every time?

7) Will this data concurrently reside on a database?

8) How many smart cards will be needed?

9) Have card or infrastructure vendors been identified? What are the lead times?

10) What are the required reader types, handsets, terminals, and middleware?

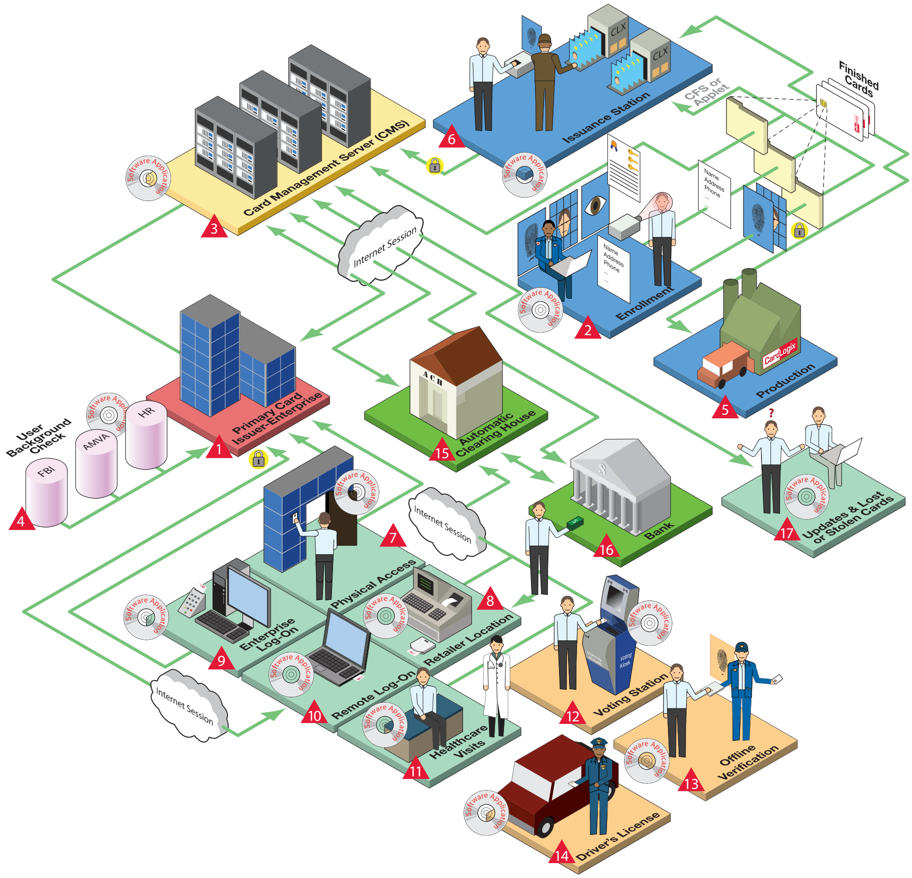
11) Is a Card Management System (CMS) necessary?

12) Do I buy an off-the-shelf software and hardware for issuance? or do I have CardLogix do this task?

13) Who will do the artwork?

14) What is needed on the card? (e.g. signature panels, magnetic stripes, embossing, etc.)

Figure 2-1. Multiple-application card usage system diagram.



Physical Design/Graphics

1) How many types of artwork will be included in the issuance?
2) What about security printing options? (e.g. guilloches, microprinting, holograms, Card Validator® images, etc.)
3) Will I print variable data at issuance?
4) Should the smart card be laser engraved to prevent low level tampering
5) Will there be an over-laminate applied to the smart card?
6) Do I need to consider level 2 (covert) and level 3 (forensic) type security attributes on the card?
7) Will the card be distributed with a carrier (Tyvek sleeve, card wallet or blister pack)?

Security

1) What are the security requirements?
2) Does all of the data need to be secure? Or only some?
3) Who will have access to this information?
4) Who will be allowed to change this information?
5) In what manner will you secure this data? (e.g. encryption, host passwords, card passwords, PINs, etc.)
6) Should the keys/PINs be customer or system activated?
7) How shall I diversify the system and card keys?
8) How will you identify the card issuance and versions?
9) Will the system utilize PKI and Digital Certificates? If so, how will they be managed?

Value Applications

1) Is value in your cards re-loadable or one-time use?
2) How will you distribute the cards?
3) How will cards be activated and loaded with value?
4) Will there be a refund policy?
5) What is the minimum and maximum value to store on each card?

Deployment Recommendations

1) Establish clear and achievable program objectives
2) Analyze the application and IT environment
3) Make sure the organization has a stake in the project's success and that management buys into the program
4) Set a budget
5) Name a project manager
6) Assemble a project team and create a team vision
7) Graphically create a dataflow diagram
8) Assess the card and reader options
9) Write a detailed specification for the cards and system
10) Set a realistic schedule with inchstones and milestones
11) Establish security parameters for people and the system
12) Build your on-card and host file structures
13) Phase in each system element and test as you deploy
14) Re-assess your system for security leaks
15) Deploy the first phase of cards and test the system
16) Train the key employees responsible for each area
17) Set up a system user manual
18) Check the reporting structures
19) Create contingency plans, should problems arise
20) Deploy and announce your system
21) Advertise and market your system

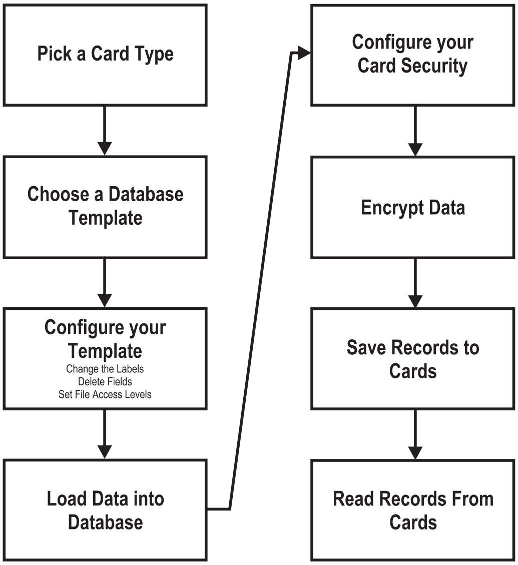
Section IV: CardAppz® Overview

Software Overview

The CardAppz® Application Software allows you to test out and demo different data and card database structures without programming. This enables you to visualize a design and verify card feasibility quickly and inexpensively. Once this is done, you can advance your design to production using CardAppz® and the Card Configuration Utility, plus the Winplex® API, the rest of your Smart Toolz® kit and any additional system elements – such as readers, to construct a solid backbone for card issuance.

The software is structured to take you through a series of logical steps that are easy to find, select and implement. When used in conjunction with the "Planning Your System" tips found in Smart Card Basics, you answer key questions that cover both the technical and business issues concerning smart card deployment. Each CardAppz® process step is set within a simple and intuitive framework, designed to look and feel like other programs you work with everyday.

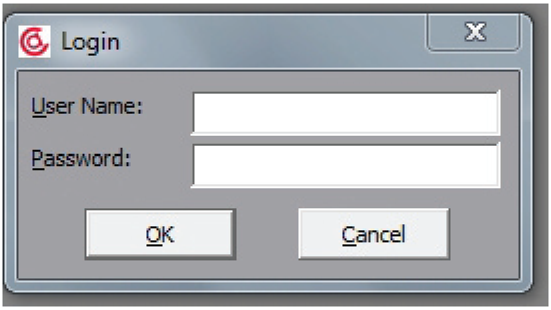
Figure 4-1: CardAppz® Implementation Workflow



Getting Around

Start CardAppz® by clicking on the icon in the Smart Toolz® program group. You are first presented with the following dialog box:

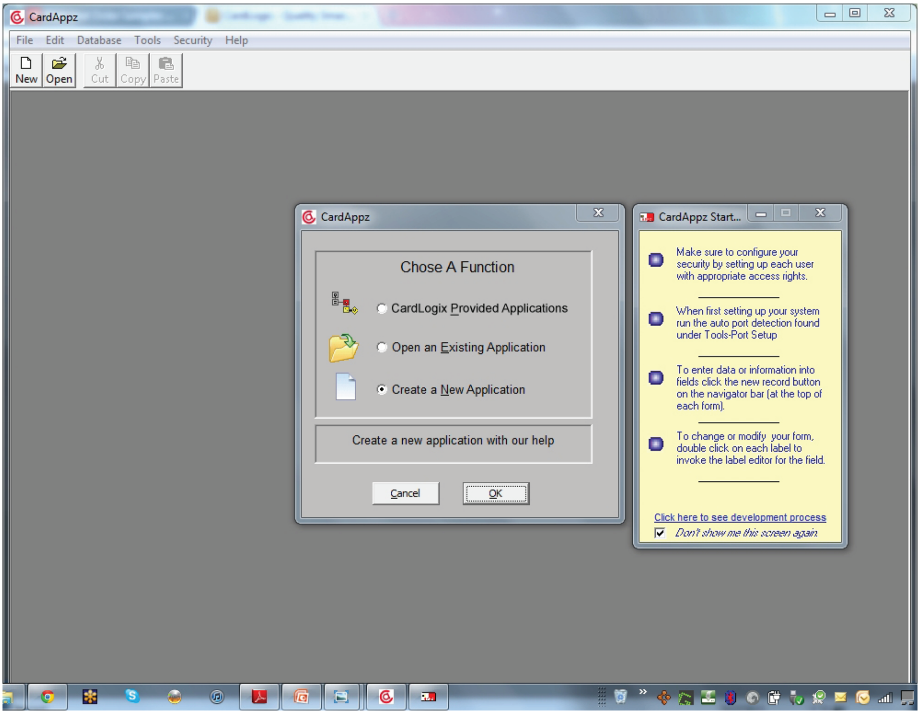
Figure 4-2: CardAppz® Login Screen



The initial user name and password is found on the inside cover of your CD-ROM case. These settings should be changed as soon as possible by the software owner to prevent tampering. Refer to the section titled "Protecting your Information" on page 4-9 for more information.

After the initial login, you are presented with the initial SmartApp dialog box and Start-up Hints dialog box. You can close either of these screens by clicking the X in the upper right-hand corner.

Figure 4-3: SmartApp & Startup Hints dialog boxes



Chose A Function

○ CardLogix Provided Applications

○ Open an Existing Application

● Create a New Application

Create a new application with our help

Make sure to configure your security by setting up each user with appropriate access rights.

When first setting up your system run the auto port detection found under Tools-Port Setup

To enter data or information into fields click the new record button on the navigator bar (at the top of each form).

To change or modify your form, double click on each label to invoke the label editor for the field.

Click here to see development process

☑ *Don't show me this screen again.*

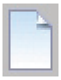
The first option is pre-formatted databases. These are finished applications that you may use as is, or can be customized for your own individual requirements. If you need help with these sample applications, please contact CardLogix directly.

The second option is opening an existing application. This selection gets you to your designs as CardAppz® automatically creates a sub-directory for your work. The path can be viewed on the title bar of your file. You may also invoke this option at any time by clicking the Open button.

Figure 4-4: CardAppz® Open button

The third option is to create a new application. You may also invoke this option at any time by clicking the New button. Unlike some programs, CardAppz® requires you to first setup your file name before anything can be done. This is due to the integrated nature of the program.

Figure 4-5: CardAppz® New button



Fill in the name of your planned card database in the dialog box. You can rename your file later if you wish. You will notice that your file name will appear on the title bar of the form.

Opening Your Work

1) Click the Open button or go to the File menu and pull down the Open command.
2) In the Look In dialog box, click the drive that contains the document.
3) In the box beneath Look In that lists folders and files, double-click the name of the folder that contains the document. Continue double-clicking sub folders until you open the sub folder that contains the document you are looking for.
4) In the list of files, double click the document name.
5) Click Open or double click on the file name.

Choosing Your Card Types

After the selection of a file name you must select a card type. The most important thing to remember when selecting a card type is user memory; unless you are building a stored value system (See "Planning Your System" for this information).

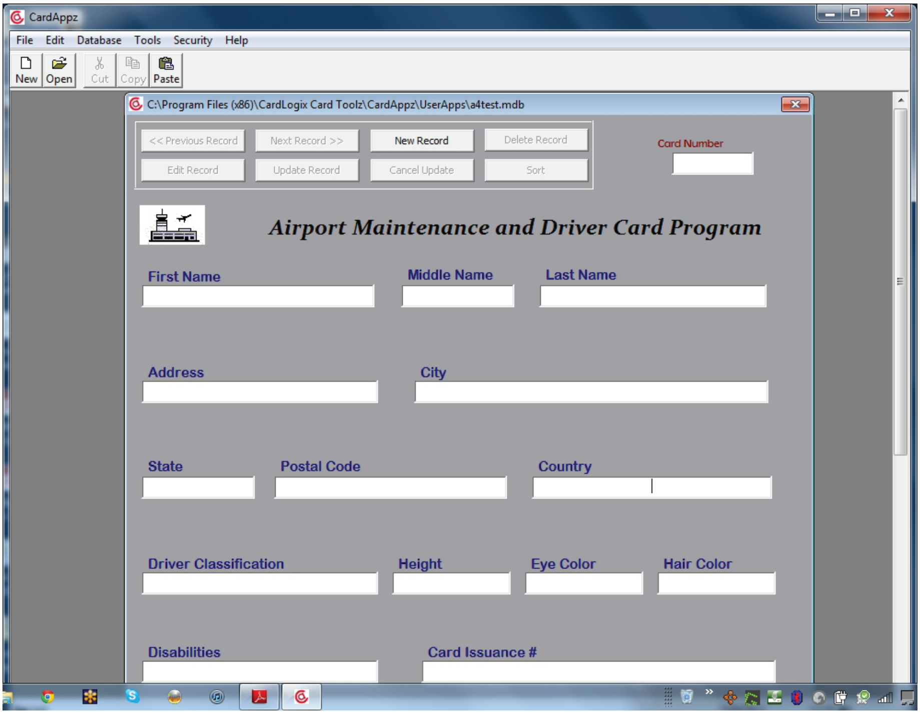
The information to be stored must be able to fit into the appropriate card size. This specific issue will be key to your overall system cost; so plan carefully. If you do not know the storage requirements of your system, use the formula we have provided below. It is the best method of calculating, in a rough order of magnitude, the bits needed for each card. For reference 8 bits equals 1 byte.

1) First go to Microsoft© Word (or any other word processing program that has a word count function), and open a new document
2) Type or copy the maximum record information, line by line, representing each field.
3) Go to the Edit menu and pull down the command for select all Ctrl+A, or highlight all of the data.
4) Go to the Tools menu and pull down the command for word count.
5) Take the total number of characters and multiply by 8.

This will give you an approximation, in bits, of what you will want to store. All cards over 1k bytes (8,000 bits) will be compressed automatically. Assume a minimum of 10% gain in storage. If you encrypt a card you will increase the file size by at least 10-15%. Encryption is only allowed on cards that are 1k bytes (8,000 bits) or larger.

CardAppz® gives you a card-by-card menu (see Figure 4-7) that pops up after you have started a new design. If you need to compare features, any device can be selected by double clicking on the Select button for that part number.

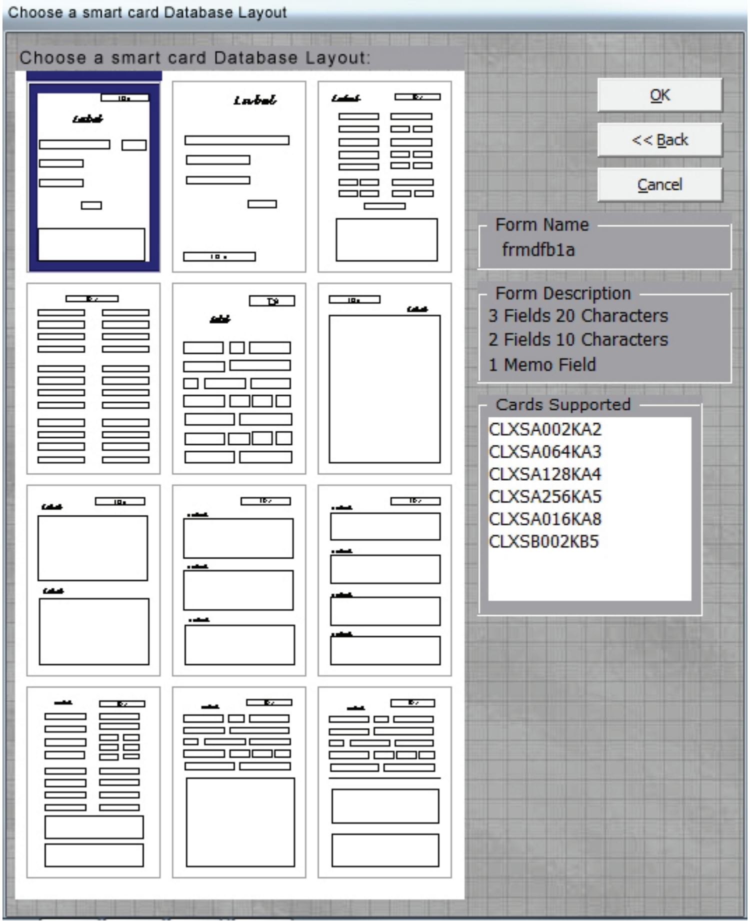
Figure 4-6: Card type selection dialog box



Building Your Database

CardAppz® assumes you have a basic knowledge of database mechanics and you understand the differences between a field, file and record. After selecting your card type, you need to choose your database structure. These pre-formatted data structures are generic and should suit many applications.

Figure 4-7: Database structure selection dialog box


Choose a smart card Database Layout
Choose a smart card Database Layout:
OK
<< Back
Cancel

Form Name
frmdfb1a

Form Description
3 Fields 20 Characters
2 Fields 10 Characters
1 Memo Field

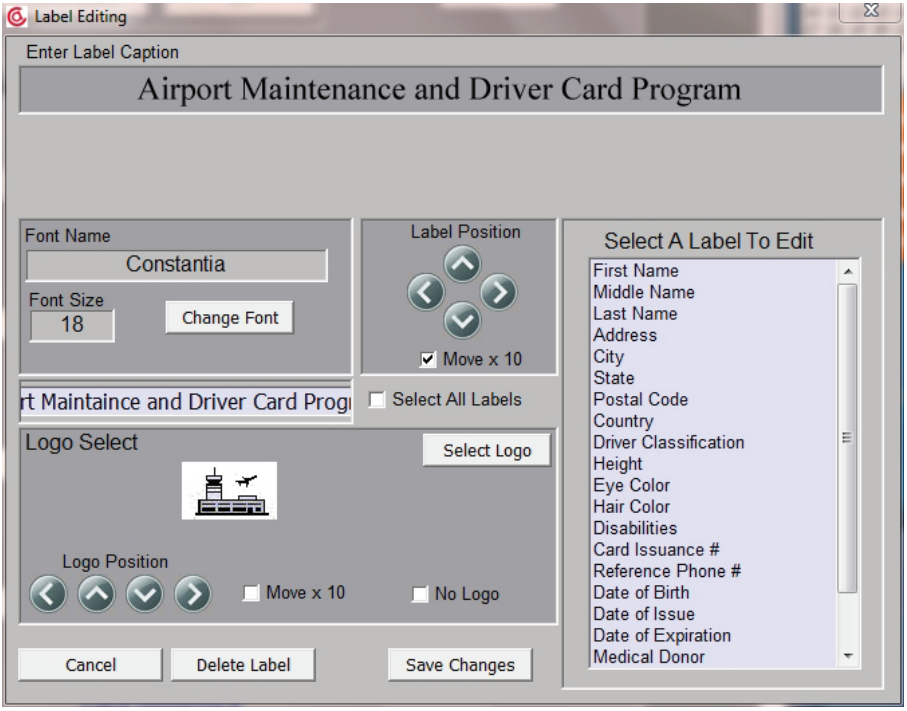
Cards Supported
CLXSA002KA2
CLXSA064KA3
CLXSA128KA4
CLXSA256KA5
CLXSA016KA8
CLXSB002KB5

Scroll through the selection thumbnails and take note of the file structure information on the bottom right of the dialog box. Each template has a card ID field already supplied for version control. You will also see a list of each card type as you click on each selection. If your chosen template does not match your selected card type you must click on the back button and choose another card type.

Labeling Your Fields

With CardAppz®, labeling your database is simple. Each file and field has an associated label that will describe its intended use for the user and your reports. To change the caption from label (1, 2, 3…) to your specific name, place your cursor over the label and double click on the item you want to change, as shown below.

Figure 4-8: Field labeling dialog boxes



Enter Label Caption: Airport Maintenance and Driver Card Program

Font Name: Constantia

Font Size: 18

☑ Move x 10

rt Maintaince and Driver Card Progı

□ Select All Labels

□ Move x 10

□ No Logo

Select A Label To Edit

- First Name
- Middle Name
- Last Name
- Address
- City
- State
- Postal Code
- Country
- Driver Classification
- Height
- Eye Color
- Hair Color
- Disabilities
- Card Issuance #
- Reference Phone #
- Date of Birth
- Date of Issue
- Date of Expiration
- Medical Donor

In the label editing dialog box that appears, click on the item button and enter your new name in the dialog box. Click done if you are finished editing. To change the look and style of each of these labels you can invoke the label editor individually or as a group. To change all the fonts and label sizes for the file, click the Select All button on the label editing dialog box or go to the `Format` menu item and pull down to the command `Select All Labels`. Make your edits on the label editing dialog box and click done when finished.

There are four primary character formats that you can apply with the Label Editor pop-up box:

1) Font type
2) Font style
3) Font size
4) Font color

Note: *The font's appearance on screen and in print may differ, depending on your computer and printer. If you select a font that your system can't display or isn't installed, Windows will substitute a similar font.*

Hint: *To reposition the label more quickly in any direction, enable the button of the direction in which you wish to move your label, then hold down the Enter key or check the 10 x box.*

Deleting A Field And Corresponding Label

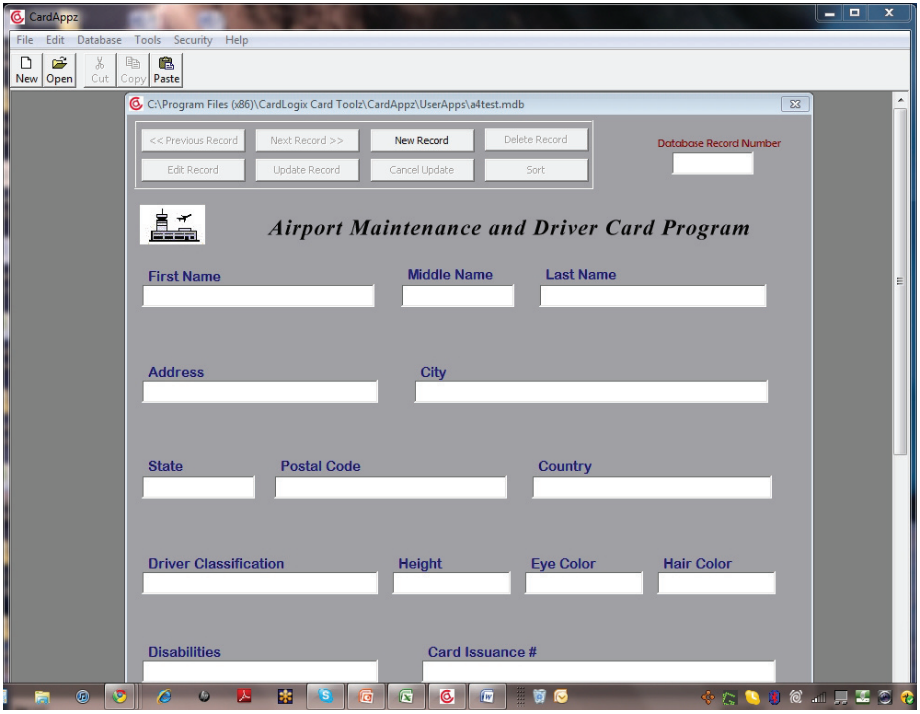
To delete a field, place your cursor over the label associated with that field and double click on the label or caption and the Label Editing dialog box will appear. Select the delete button. You can delete a field at any time as long as it has not been locked.

Warning: *In the process of establishing your database, you must be careful to delete only those fields that you are sure will not be needed after card issuance. Once cards are issued, deletion of a field will lose all data stored and reinstallation will not be possible. This will in turn render your issued cards unreadable, generating an error message when use is attempted.*

Entering Your Data

To enter new data, click on the "New Record" button at the top of the page. The fields are now ready to accept data.

Figure 4-9: Data entry dialog box



Getting data into the host system can be accomplished in four ways.

- The normal method is through standard manual data entry.
- You can also cut and paste data from other CardAppz® files or from other applications. For example, to edit large amounts of text you can open your favorite word processor, edit your text there, and then copy [Ctrl+C] and paste [Ctrl+V] it into the CardAppz® database text field.
- Data can also be read into the system from individual smart cards that were created with the same CardAppz® software, provided that the passwords and keys match.
- The last method for entering data is to import data from other databases - see **`Tools > Import`**.

Note: For reasons of continuity, you cannot assign card numbers. Rather, these are assigned automatically, after you have entered your data, and have pressed the Update Record button.

Exporting Your Data

To export a file, select Export from the Tools menu. This will automatically open up a sub-menu giving you the choice of saving to Microsoft Access Database Format or a Delimited ASCII text file.

Properties

Can't quite recall the properties of your cards or files? Bring up the file in question, and click on File tab. From the drop down menu, select Properties. This will give file information and card information. File information will tell you the database type, when the file was created, when the file was modified, when the file was last accessed, and the host security attributes. Card information will tell you the type of card you selected, how many cards were issued, if there are passwords set, and if the cards are encrypted.

Saving Your Work

CardAppz® automatically saves your database when you create it. After editing the database template, choose Save from the File menu. This saves the overall layout and design of your database, not the records within the database itself.

As you enter data into your database, each data record within the database is saved when you click on the "Enter Record" button.

You can save a copy of the active database file with a different name or in a different location by going to the File menu and pulling down the `Save As` command.

Note: *You cannot save a document in another file format. For example, you cannot save a CardAppz® document in a file format other than the .clx extension that the program was supplied with. To export data see* `Tools > Export`.

To save changes to a read-only file, use the `Save  As` command to save it with a new name.

Protecting Your Information

CardAppz® lets you, as the card issuer, set all of the parameters for card and data security. As an overview, there are two types of data system security, host-based and card-based. The safest systems employ both methods.

Note: *The protection of data is a subject of much debate. CardLogix provides a range of options for you to implement security but the ultimate choice and responsibility is yours as a card system designer and issuer. Most systems that are compromised are done so from the inside. Every reported breach of a smart card stored value system that we know of has been due to card emulation or theft through the mismanagement of keys/ passwords by disgruntled/unscrupulous employees.*

Host-Based Security - File Access

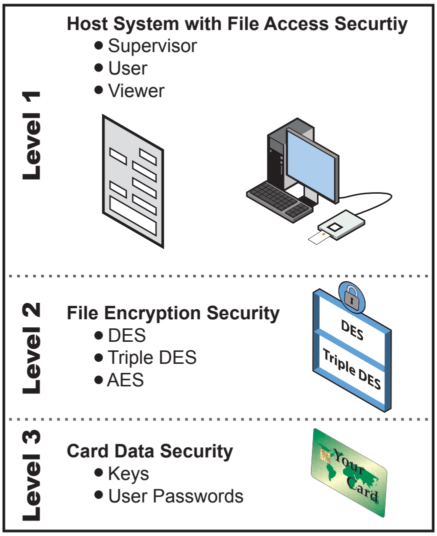
The person designing with CardAppz® has the power to set the access levels to the program data and files from the host. The CardAppz® default mode uses no passwords except for the initial login. The first point of control to your CardAppz® system is the login password setup by a supervisor. All passwords are hashed and spread-out through the program so that simple points of attack are made more difficult. The initial user name and password should be changed immediately at the setup and editing of your first file.

The System has three file access levels. They are:

- Supervisor - All rights and privileges to change databases, card ID#s, file locking, encryption keys and labeling. This password must be set before the other two levels can be initialized.
- User - This level is for the data-entry personnel to input and change data on both cards and the host.
- Viewer - This is a read-only mode.

These differing levels of program access are to demonstrate how a finished Smart Card Data Application may be structured. With proper programming and issuance all passwords should be replaced with cards as program access permissions are set.

Figure 4-10: CardAppz® security levels diagram



Setting Your Security Levels

Protecting your CardAppz® database application is easy. Here is how you set your **Supervisor**, **User**, and **Viewer** passwords.

*Note: Setting a Supervisor name and password replaces the default settings created by CardLogix at the point of manufacture. To ensure that the user name "Supervisor" and the password "Password" cannot be used again, they must be deleted by selecting Delete User instead of simply deleting the settings when prompted.**Remember:** All passwords are case-sensitive.*

Do not use a viewer password if you will be replicating a database. Replicated databases cannot be synchronized if database passwords are defined.

Protecting your CardAppz® database application is easy. Here is how you set your Supervisor, User, and Viewer passwords.

A) Save and close the database file. If the database is shared on a network, ask all other users to close the database.

B) On the **`File`** menu, click **`Open`**.

C) At the file directory dialog box, choose your file and double click to open the database.

D) On the **`Security`** menu, pull down the command to **`User Manager`**, and then select the appropriate option: **`Create User`**, **`Delete User`**, or **`Edit User`**

E) Select **`Permission Level`**: **`Supervisor`**, **`User`**, or **`Viewer`**

F) Enter User Name

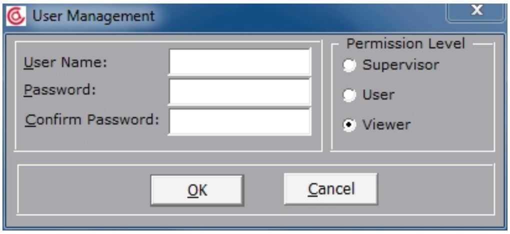
G) In the Password box, type your password. Remember passwords are case-sensitive.

H) In the Verify box, confirm your password by typing it again, and then click OK.

The password is now set. The next time you or any other user opens the database, a dialog box will be displayed that requests a password. Remember that a password is case- sensitive; you must type it exactly as defined.

Repeat this process for the user and viewer passwords as well.

Figure 4-11: CardAppz® user management dialog box



Permission Level

○ Supervisor

○ User

● Viewer

Warning: *If you lose or forget your password, it can't be recovered and you won't be able to open your database. You are done for. Even our programmers can't help you! There is no back door! Keep a list of your passwords and their databases in a safe place.*

Remember: *All passwords are case-sensitive.*

Do not use a file password if you will be replicating a database. Replicated databases cannot be synchronized if database passwords are defined.

If you need more extensive security than provided by a file password, you need to define user-level security. You can't set a file password if user-level security has been defined for your database and you don't have supervisory permission for the database. Also, a database password is defined in addition to user-level security. If user-level security has been defined, any restrictions based on user-level security permissions remain in effect.

Encryption

The CardAppz® program supports three types of host-based encryption that can aid in card and card system security. Used independently, these methods will not necessarily prevent people from looking at the data in a card or file, but reading the files should prove to be a complete waste of time, due to the scrambled nature of encrypted data. All CardLogix cards with capacities above 32,767 bits or 4k bytes are automatically compressed using a loss-less algorithm incorporated in the CardAppz® system. This compression function alone applies a basic type of encryption.

The Data Encryption Standard (DES) is utilized for its proven security services. Developed in the early 1970s by IBM, DES is a Symmetric Key Algorithm (sometimes referred to as secret key). It is now considered the de-facto standard among many private industries and government. DES has been widely accepted as the standard for secure transmissions that conform to the requirements of the American Banking Association. When used with smart cards, DES data can be secured at a very low cost.

As an enhancement to DES we have incorporated Triple DES as an option for encrypting your data. This methodology requires two passwords per routine and can be used if you need enhanced security.

As a third option we let you apply the AES algorithm to your data. This proven algorithm has never been cracked and offers an encryption method that most hackers will not even guess at.

A) To encrypt a file you must first establish the file user levels.

B) Next, go to Security on the menu bar and pull down to the encryption function you want to perform.

C) When the dialog box opens, fill in your passwords/Keys, write them down, and store them in a safe place.

D) Your passwords/keys can be any combination of numbers or text. The system will then exponentiate this input to a large digital number that becomes your electronic key for this encryption.

Warning: *There are no back doors to your data; so consider these actions carefully before possibly locking yourself out of your own system due to misplaced passwords.*

Setting Card-Based Security

Various versions of the CardLogix memory card families support on-card passwords and digital keys. These functions may only prevent the alteration of the data without authorization. Please refer to the most current version of the CardLogix Smart Card Selection & Design Guide for the specific details. To establish higher security on certain cards, the memory must be fused at the point of manufacture. Please contact us regarding these issues directly. If your application will require higher security and needs to utilize dynamic password sessions and advanced authentication or encryption methods, please refer to the CardLogix M.O.S.T.® and Credentsys Cards to meet these requirements.

Note: *If you are in need of a high security card system, contact CardLogix directly for information on our family of microprocessor cards.*

To set up the passwords for on-card security cards, follow these simple steps.

A) Go to Security on the menu bar and pull down to the Set Card Password command.

B) When the dialog box opens, fill in your password and write it down in a safe place.

C) Your card password can be any combination of numbers or text up to 8 characters long.

D) After entering your password, make sure your card is inserted in the reader and you save the password to the card.

Key Management

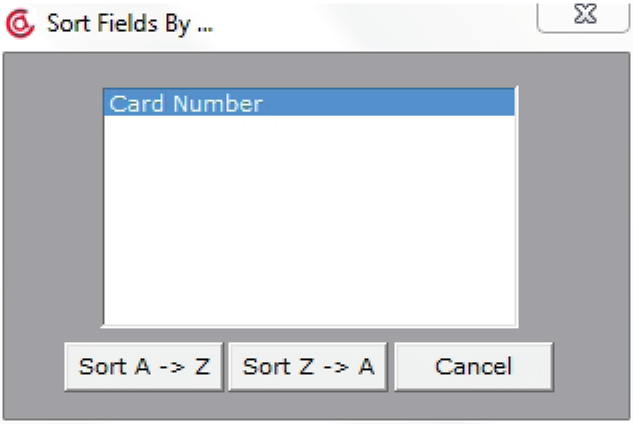
Keys are passwords that are automatically presented by the reader. When setting up your passwords consider the issues of auto passing the key for convenience of the system user. Each group of applications may have one or more keys that separate the applications from each other. The following cards have a factory default key set for a customized system change. Contact CardLogix for more information.

- CLXSA000KB1
- CLXSB002KB5
- CLXSB000KB6
- CLXSU190KF6T=0ED
- CLXSU004KK4T=0LC

Sorting Your Data

By sorting your data in a specific structure you can often find information more quickly. This is true whether you're looking at multiple records in a file, or viewing a list of values. In CardAppz® you can sort data in ascending order (0 to 9, A to Z) or descending order (9 to 0, Z to A) by going to the Navigator Bar and selecting sort, then select how you wish your fields to be sorted. The names you have given each field (in place of Label 1, 2, 3...) will appear in the "Sort Fields By..." window.

Figure 4-12: CardAppz® field sorting dialog box



When you are done with your proof of concept your are ready to take the next step. This step usually starts with the included Card Configuration Utility and/or M.O.S.T. Toolz® used in conjunction with Winplex® to complete your system.

Section V: Card Configuration Utility

Design It. Test It. Produce It.

The Card Configuration Utility designed by CardLogix Corporation allows developers to complete their smart card projects in a simplified *Design*, *Test*, and *Produce* method. Developers can design a card structure through an easy to use interface that provides access tools designed specifically for the features of the card that they are using. This allows advanced features of a card to be implemented without spending endless hours in research and documentation. At anytime, a project can be written directly to a physical smart card for testing prior to production. That way developers can test their ideas and catch any mistakes. Once satisfied with the design of their smart card structure, the utility can save a project as a project file to be sent to CardLogix for a production run.

I^{2}C Memory Cards (A2 through A9)

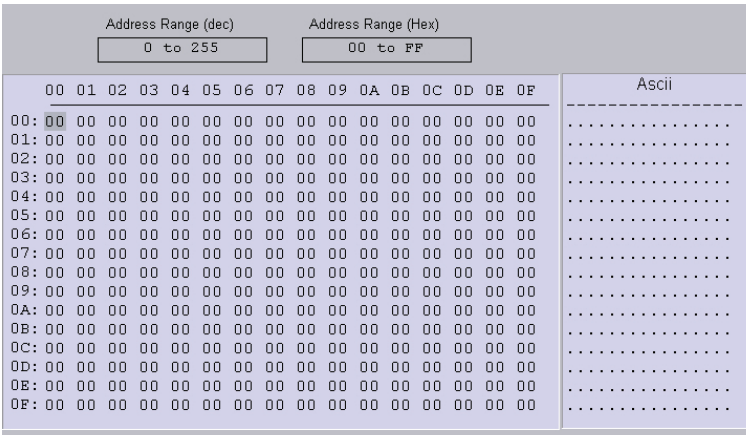
Card Layout

I^{2}C cards have a synchronous protocol that requires each reader to have support built-in, before use. Readers by SCM, MagTek and Omnikey do support these functions, but will vary in the support for the larger card densities such as 512k and 1 megabit devices.

It is recommended that these details be investigated before your full system is deployed.

Design Note

If you are using the I^{2}C devices for an embedded design, make sure that your reader is designed around the full electrical specification—including the lower voltage devices. As these devices mature and go through revisions with each semiconductor manufacturer, the electrical characteristics will vary.

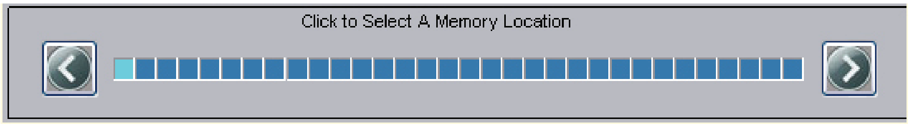


```
        Address Range (dec)          Address Range (Hex)
             0 to 255                     00 to FF

     00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F          Ascii
                                                         ----------------
00:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
01:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
02:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
03:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
04:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
05:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
06:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
07:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
08:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
09:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0A:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0B:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0C:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0D:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0E:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
0F:  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00     ................
```

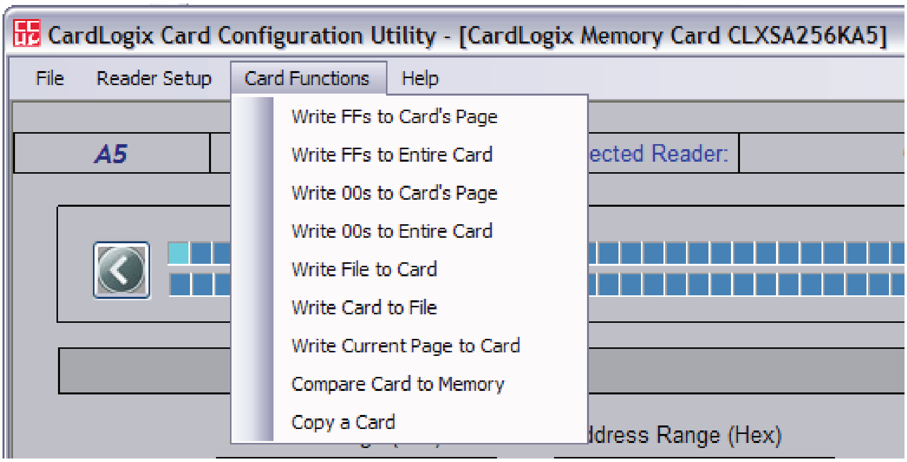
- I^{2}C Memory Cards have a continuous memory system, starting with Byte 00
- The Card's data is shown to the user in 256 byte pages
- You can edit bytes by selecting a byte and entering the hex values (using the 0-9, A-F keys)
- The Tab and Arrow Keys can be used to select other bytes in the current page
- An ASCII representation of the byte values is displayed on the right side of the screen

Navigating the Card



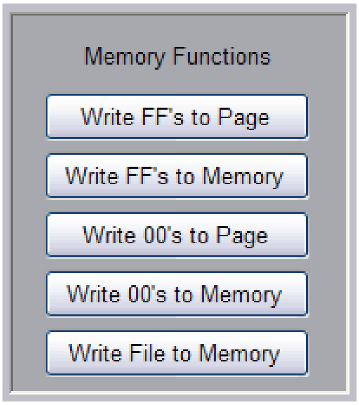
To change to other pages of the card, click on the arrow indicators show above. The blue boxes represent how many pages are available on your particular I^{2}C card. In the above example, for an A3 card there are 32 pages. The lighter colored blue box indicates we are currently on the first pages of the card. The boxes are clickable and will take you directly to that corresponding page.

Card Function Menu



For I^2C cards there is a special Card Functions menu. The commands in this menu, all work directly with a connected reader and I^2C card. It will not affect your current project or memory. This can be convenient for erasing a card on the fly, or for making a copy of a card.

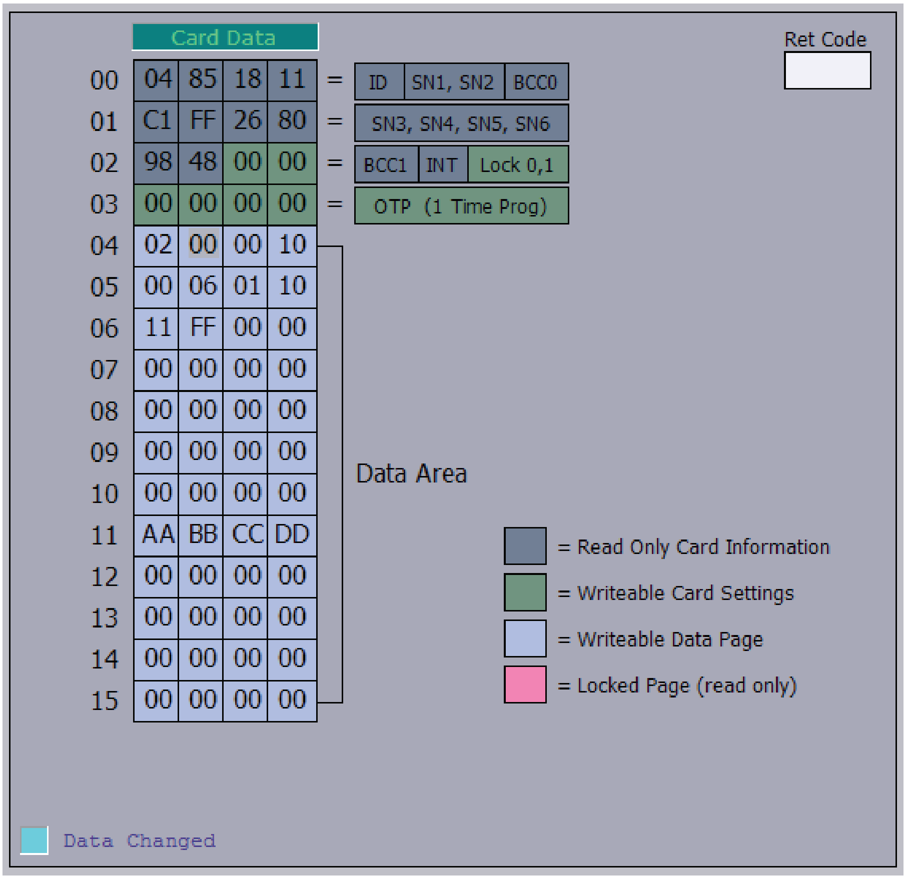
Special Memory Functions



These buttons provide special memory functions to the user. For example, you can quickly erase the working buffer with 00's or FF's. You can also import a text or binary file through the Write File to Memory button.

MIFARE® Ultralight (CLXRN512UN1) Cards

Card Header Bytes



Card Data

Ret Code

00	04	85	18	11	=	ID | SN1, SN2 | BCC0
01	C1	FF	26	80	=	SN3, SN4, SN5, SN6
02	98	48	00	00	=	BCC1 | INT | Lock 0,1
03	00	00	00	00	=	OTP (1 Time Prog)
04	02	00	00	10		Data Area
05	00	06	01	10		
06	11	FF	00	00		
07	00	00	00	00		
08	00	00	00	00		
09	00	00	00	00		
10	00	00	00	00		
11	AA	BB	CC	DD		
12	00	00	00	00		
13	00	00	00	00		
14	00	00	00	00		
15	00	00	00	00		

🟦 = Read Only Card Information

🟩 = Writeable Card Settings

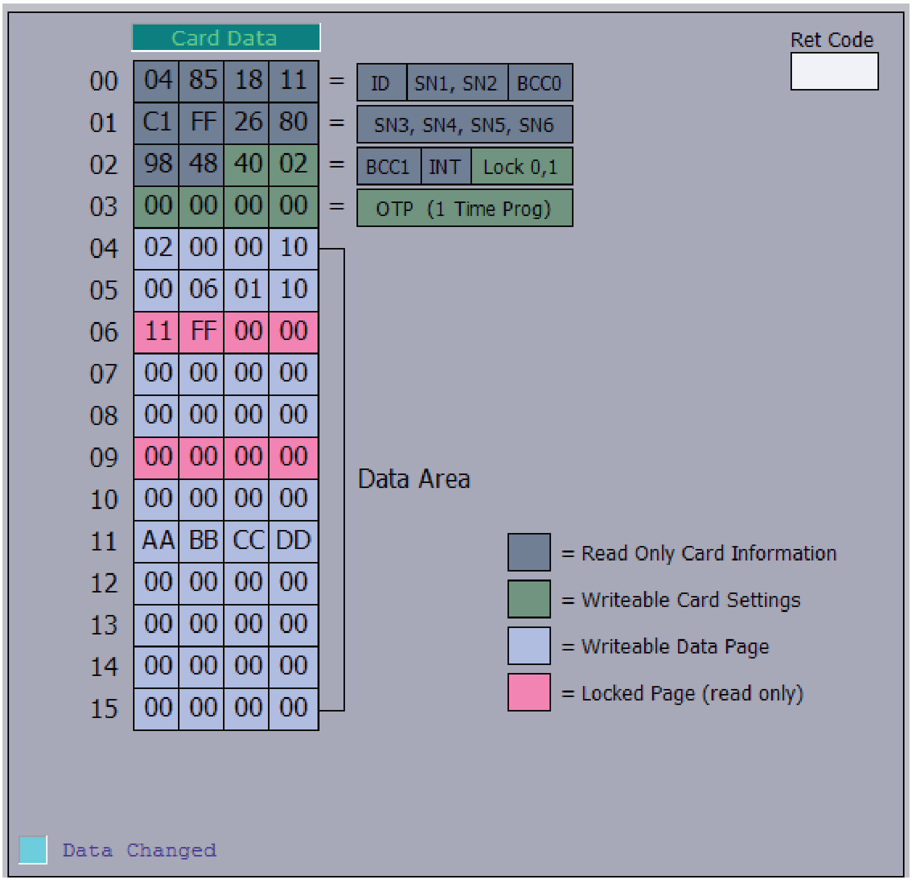
🟪 = Writeable Data Page

🟥 = Locked Page (read only)

□ Data Changed

The first 10 bytes of the MIFARE® Ultralight (CLXRN512UN1) Card are read-only. They are set by NXP in advance at wafer finalization, based on a large order commitment to indicate manufacturer, card ID, serial numbers, and other information. Editing these values will have no effect on the card since they cannot be changed.

Lock Bytes



Ret Code

	Card Data					
00	04	85	18	11	=	ID | SN1, SN2 | BCC0
01	C1	FF	26	80	=	SN3, SN4, SN5, SN6
02	98	48	40	02	=	BCC1 | INT | Lock 0,1
03	00	00	00	00	=	OTP (1 Time Prog)
04	02	00	00	10		Data Area
05	00	06	01	10		
06	11	FF	00	00		
07	00	00	00	00		
08	00	00	00	00		
09	00	00	00	00		
10	00	00	00	00		
11	AA	BB	CC	DD		
12	00	00	00	00		
13	00	00	00	00		
14	00	00	00	00		
15	00	00	00	00		

■ = Read Only Card Information

🟩 = Writeable Card Settings

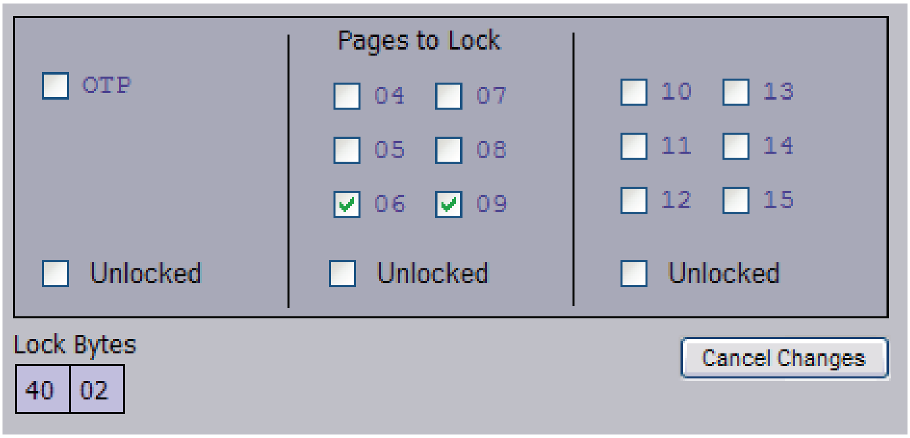
🟦 = Writeable Data Page

🟥 = Locked Page (read only)

🟦 Data Changed

Bytes 11 and 12 of the card are considered the lock bytes. The bits in these bytes lock pages (4 byte sections) of the card, beginning at the OTP area. Once a bit is set to 1 in a lock byte, the bit cannot be changed back to 0, so caution is advised when saving to a card.

Lock Bytes Tool



Pages to Lock

□ OTP

□ 04 □ 07

□ 05 □ 08

☑ 06 ☑ 09

□ 10 □ 13

□ 11 □ 14

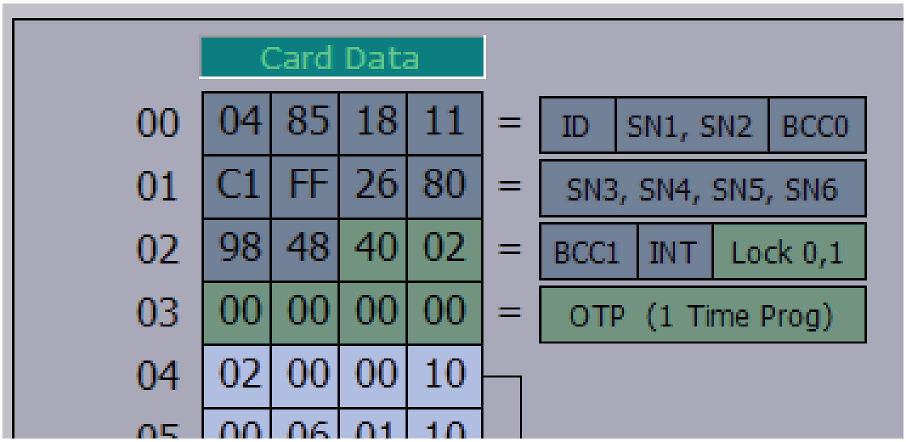
□ 12 □ 15

□ Unlocked □ Unlocked □ Unlocked

Lock Bytes: 40 02

We recommend using the utility shown, in order that specific pages or blocks of pages can be locked individually without the need to calculate byte values.

One Time Programming

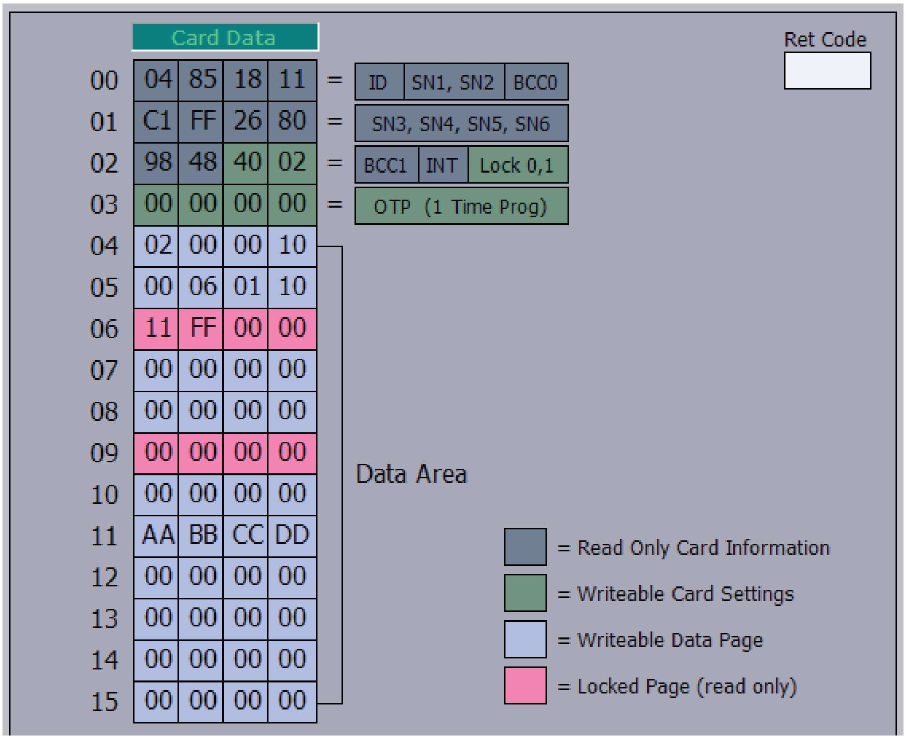


Card Data

00	04	85	18	11	=	ID SN1, SN2 BCC0
01	C1	FF	26	80	=	SN3, SN4, SN5, SN6
02	98	48	40	02	=	BCC1 INT Lock 0,1
03	00	00	00	00	=	OTP (1 Time Prog)
04	02	00	00	10		
05	00	06	01	10		

Bytes 13 through 16 of block 3 are bytes that can only change state one time. Any 0 bit can be changed to a 1 bit but 1 bits can never be set back to 0 bits again. This feature can be used for a 32-tick one-time counter or as a "One Trip Pass" for events or ticketing. Optionally this block can be used for data storage where the values will not change later, like a serial number. When used for this purpose, block 3 needs to be locked after programming.

Data Area



Ret Code

	Card Data					
00	04	85	18	11	=	ID, SN1, SN2, BCC0
01	C1	FF	26	80	=	SN3, SN4, SN5, SN6
02	98	48	40	02	=	BCC1, INT, Lock 0,1
03	00	00	00	00	=	OTP (1 Time Prog)
04	02	00	00	10		
05	00	06	01	10		
06	11	FF	00	00		
07	00	00	00	00		
08	00	00	00	00		
09	00	00	00	00		
10	00	00	00	00		
11	AA	BB	CC	DD		
12	00	00	00	00		
13	00	00	00	00		
14	00	00	00	00		
15	00	00	00	00		

Data Area

■ = Read Only Card Information

🟩 = Writeable Card Settings

🟦 = Writeable Data Page

🟥 = Locked Page (read only)

This is the primary data area of the card. It is where your application data will be stored. The 48 bytes stored here can be modified by entering hex values (0-9, A-F) on the keyboard. You can navigate through the bytes using the tab or arrow keys.

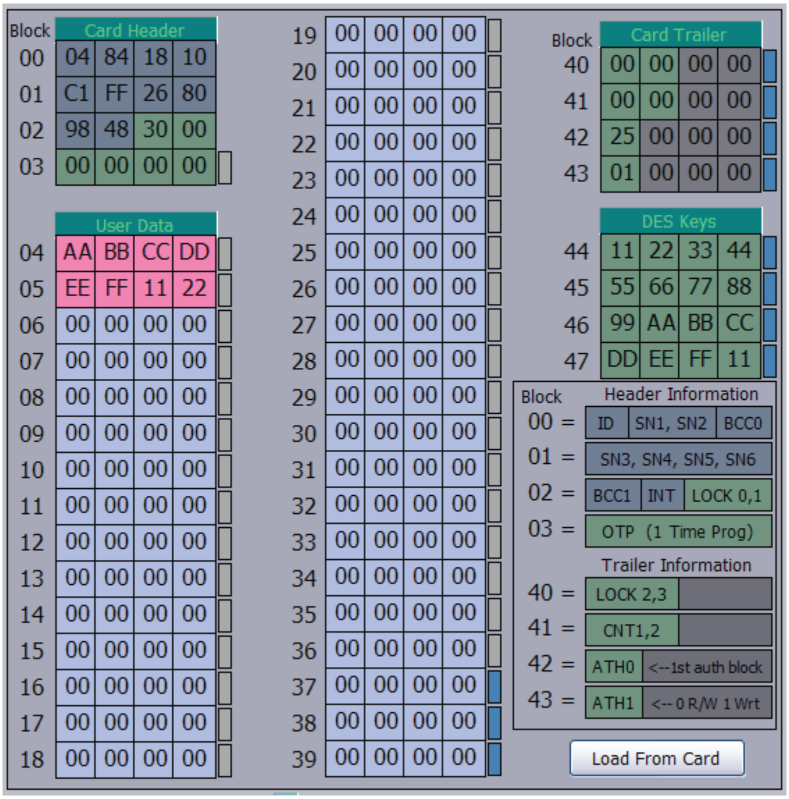
MIFARE® Ultralight C

Overview of the MIFARE® Ultralight C

The MIFARE® Ultralight C is a low cost, non-contact smart card with encrypted security. The Ultralight C is useful for applications that do not require a great deal of memory, but do require security of TripleDES encryption. This card contains 144 bytes of user data, a TripleDES encryption key, a 7 byte unique serial number, and lock bytes for marking desired memory areas as read only.

Typical applications where the MIFARE® Ultralight C would be useful include: Transportation, stored value/loyalty progams, gift cards, and secure area identification.

Card Structure



Block	Card Header			
00	04	84	18	10
01	C1	FF	26	80
02	98	48	30	00
03	00	00	00	00

	User Data			
04	AA	BB	CC	DD
05	EE	FF	11	22
06	00	00	00	00
07	00	00	00	00
08	00	00	00	00
09	00	00	00	00
10	00	00	00	00
11	00	00	00	00
12	00	00	00	00
13	00	00	00	00
14	00	00	00	00
15	00	00	00	00
16	00	00	00	00
17	00	00	00	00
18	00	00	00	00

19	00	00	00	00
20	00	00	00	00
21	00	00	00	00
22	00	00	00	00
23	00	00	00	00
24	00	00	00	00
25	00	00	00	00
26	00	00	00	00
27	00	00	00	00
28	00	00	00	00
29	00	00	00	00
30	00	00	00	00
31	00	00	00	00
32	00	00	00	00
33	00	00	00	00
34	00	00	00	00
35	00	00	00	00
36	00	00	00	00
37	00	00	00	00
38	00	00	00	00
39	00	00	00	00

Block	Card Trailer			
40	00	00	00	00
41	00	00	00	00
42	25	00	00	00
43	01	00	00	00

	DES Keys			
44	11	22	33	44
45	55	66	77	88
46	99	AA	BB	CC
47	DD	EE	FF	11

Block	Header Information		
00 =	ID	SN1, SN2	BCC0
01 =	SN3, SN4, SN5, SN6		
02 =	BCC1	INT	LOCK 0,1
03 =	OTP (1 Time Prog)		

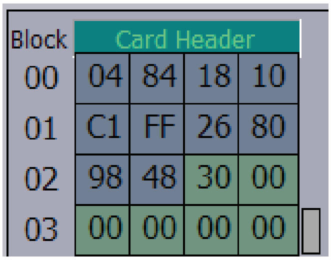
	Trailer Information	
40 =	LOCK 2,3	
41 =	CNT1,2	
42 =	ATH0	<--1st auth block
43 =	ATH1	<-- 0 R/W 1 Wrt

The card structure of the MIFARE® Ultralight C consists of 48 blocks, as follows:

Blocks	Definition
00 through 03	Card Header
04 through 39	User Data
40 through 43	Card Trailer
44 through 47	DES Keys

Card Header

The Card Header consists of a 7 byte **Unique Serial ID, Block Check Character Bytes, and Lock Bytes** that lock portions of the card as read only memory. Bytes 13 through 16 of block 3 are bytes that can only change state one time. Any 0 bit can be changed to a 1 bit but 1 bits can never be set back to 0 bits again. This feature can be used for a 32-tick one-time counter or as a "One Trip Pass" for events or ticketing. Optionally this block can be used for data storage where the values will not change later, like a serial number. When used for this purpose, block 3 needs to be locked after programming.



Block	Card Header			
00	04	84	18	10
01	C1	FF	26	80
02	98	48	30	00
03	00	00	00	00

Block	Byte 0	Byte 1	Byte 2	Byte 3
00	Manufacturer ID	Unique Serial #1	Unique Serial #2	Block Check Character Byte 0
01	Unique Serial #3	Unique Serial #4	Unique Serial #5	Unique Serial #6
02	Block Check Character Byte 1	Internal Use	Lock Byte 0	Lock Byte 1
03	One Time Programming 1	One Time Programming 2	One Time Programming 3	One Time Programming 4

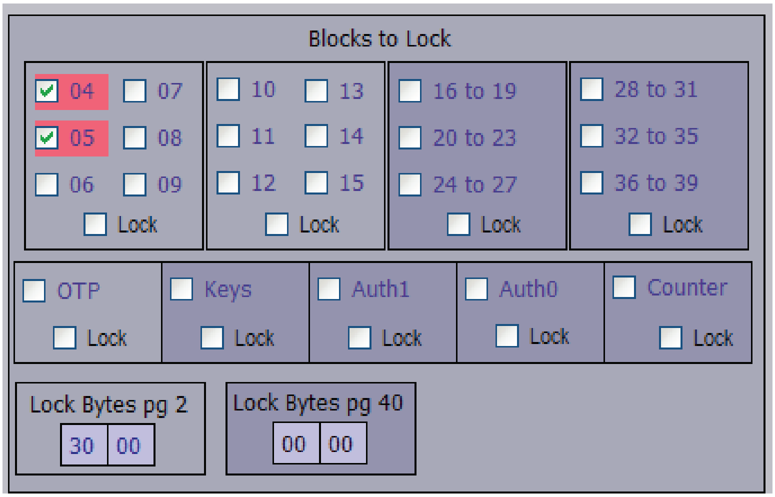
Unique Serial ID

Each card contains a unique 7 byte Serial ID, consisting of one byte for the Manufacturer ID, and 6 bytes for the remainder of the Serial Identification.

BCC0 / BCC1 Block Check Character Bytes

The BCC0 and BCC1 blocks contain checksum values for the header information.

Lock Bytes



Blocks to Lock

☑ 04 □ 07
☑ 05 □ 08
□ 06 □ 09
□ Lock

□ 10 □ 13
□ 11 □ 14
□ 12 □ 15
□ Lock

□ 16 to 19
□ 20 to 23
□ 24 to 27
□ Lock

□ 28 to 31
□ 32 to 35
□ 36 to 39
□ Lock

□ OTP
□ Lock

□ Keys
□ Lock

□ Auth1
□ Lock

□ Auth0
□ Lock

□ Counter
□ Lock

Lock Bytes pg 2: 30 00

Lock Bytes pg 40: 00 00

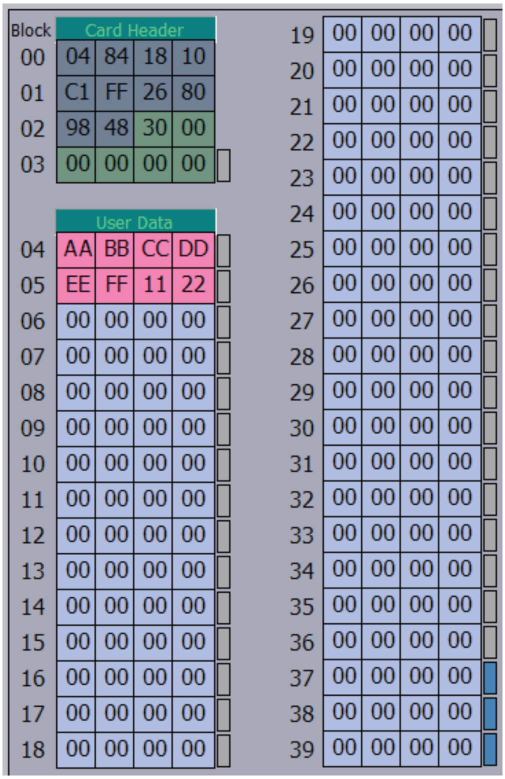
Bytes 2 and 3 of Block 02 and Bytes 0 and 1 of Block 40 of the MIFARE® Ultralight C card are called **Lock Bytes**. The values in those bytes correspond with areas that developers want to mark as read-only on the card. We recommend using the utility as shown above to lock the various sections of the MIFARE® Ultralight C. Once satisfied with your lock configurations you can lock each of the three sections permanently in their current states by clicking the corresponding Lock Check boxes.

One Time Programming

Block 03 Contains 4 OTP (One Time Programming) Bytes. These special handling bytes allow developers to add their own unique bytes that cannot be changed in the field*

*When a bit is set to 1, in the OTP area, it cannot be set back to 0. Because of the unique functionality of this area, it can also be used as a one time counter.

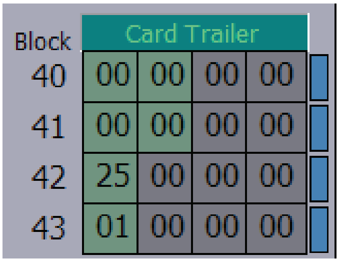
User Data



Block	Card Header			
00	04	84	18	10
01	C1	FF	26	80
02	98	48	30	00
03	00	00	00	00
	User Data			
04	AA	BB	CC	DD
05	EE	FF	11	22
06	00	00	00	00
07	00	00	00	00
08	00	00	00	00
09	00	00	00	00
10	00	00	00	00
11	00	00	00	00
12	00	00	00	00
13	00	00	00	00
14	00	00	00	00
15	00	00	00	00
16	00	00	00	00
17	00	00	00	00
18	00	00	00	00
19	00	00	00	00
20	00	00	00	00
21	00	00	00	00
22	00	00	00	00
23	00	00	00	00
24	00	00	00	00
25	00	00	00	00
26	00	00	00	00
27	00	00	00	00
28	00	00	00	00
29	00	00	00	00
30	00	00	00	00
31	00	00	00	00
32	00	00	00	00
33	00	00	00	00
34	00	00	00	00
35	00	00	00	00
36	00	00	00	00
37	00	00	00	00
38	00	00	00	00
39	00	00	00	00

Blocks 04 through 39 of the MIFARE® Ultralight C card are considered to be the User Data area. This area consists of 36 Blocks of Data, or 144 Bytes. This area is where a developer would put the primary application data.

Card Trailer



Block	Card Trailer			
40	00	00	00	00
41	00	00	00	00
42	25	00	00	00
43	01	00	00	00

Block	Byte 0	Byte 1	Byte 2	Byte 3
40	Lock Bytes 2	Lock Bytes 3	Reserved	Reserved
41	16 Bit Counter 1	16 Bit Counter 2	Reserved	Reserved
42	AUTH 1 : Authenticate Block Start	Reserved	Reserved	Reserved
43	AUTH 2 : Authentication Type	Reserved	Reserved	Reserved

Lock Bytes 2 / 3

Bytes 0 and 1 of Block 40 consist of the Lock Bytes 2 and 3. These bytes determine which sections of the card are locked in the second half of the card. Be sure to look at the Lock Tool for easy manipulation of the lock byte values.

16 Bit Counter

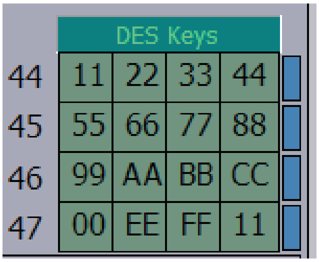
Bytes 0 and 1 of Block 41 make up a 16 bit user counter. Once a bit has been set to 1 it cannot be changed back to 0. This area can be used for applications that keep track of one-time use, such as a trip counter.

Authentication Bytes

The first byte of block 42 is called AUTH1. The value placed there determines the start of the authentication area. The number entered there will be the number of the first block that needs to be authenticated with the DES keys. Authentication is required for every block that follows that block number. The transport setting of hex 30 (decimal 48) is one higher than the last block on the card (block 47), indicating that no blocks require authentication. In the example shown, block 42 has a value of hex 25 (decimal 37) indicating that all blocks from 37 and up require DES authentication.

AUTH2, located at byte 43 determines the type of authentication required. If the value at block 43 is 00 then authentication is for both Read and Write. If the value at block 43 is 01 then the authentication required is only for Write operations and read operations are not restricted. These two are the only authentication configurations possible.

DES Keys



	DES Keys			
44	11	22	33	44
45	55	66	77	88
46	99	AA	BB	CC
47	00	EE	FF	11

The MIFARE® Ultralight C uses 3DES as its authentication encryption method. Blocks 44 through 47 make up the 16 byte secret key. For security purposes, be sure that the second half of the bytes are not identical to the first 8 bytes or else the encryption method will be weakened to Single DES. The 16 byte key used here will be for authentication with the card where required.

Finalizing Your Card

After you have set your card configuration values, write your configuration to a MIFARE® Ultralight C card using the File menu by selecting `File>Save to Card`. Using your application program, thoroughly test all card functions.

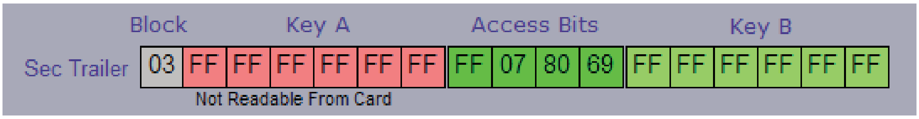
Once you are satisfied that your card structure is working correctly with your application the next step is to send your card project file (CPF) to CardLogix to have 'check cards' made. Save your configuration settings to your project file using the file menu `File>Save Project File` or `File>Save Project File As`...

Send your project file to CardLogix. Using your project file, check cards will be made and returned to you, along with a sign-off sheet. After you sign off on the check card we can advance your design to full production. You should also contact your CardLogix sales person for information on the personalization features of your card.

If your cards contain information that needs to be secure, such information may need to be transferred using an encryption method, PGP keys or other secure means of data transfer. Please contact your CardLogix representative for further information.

MIFARE® 1K (CLXRN008KN3) Cards

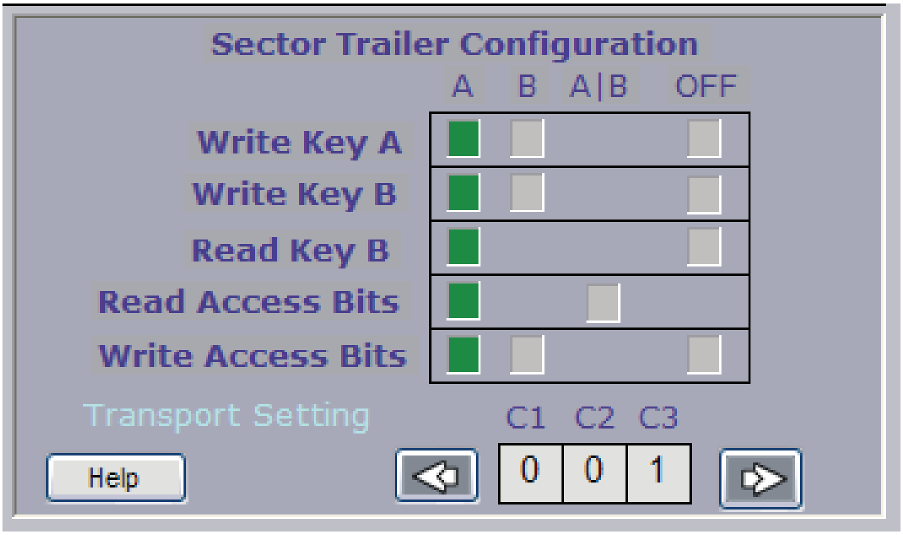
Sector Trailer



	Block	Key A						Access Bits				Key B					
Sec Trailer	03	FF	FF	FF	FF	FF	FF	FF	07	80	69	FF	FF	FF	FF	FF	FF
		Not Readable From Card															

Block 03 of each Sector on a MIFARE® 1K card contains a Sector Trailer. A Sector Trailer consists of an access key--called Key A, Access Bits that determine access privileges to blocks in the current sector, and Access Bits in the Sector Trailer itself. There is a second key--called Key B--that can be used to access specific areas of the card, or it can be configured as readable with Key A.

Sector Trailer Configuration



Sector Trailer Configuration

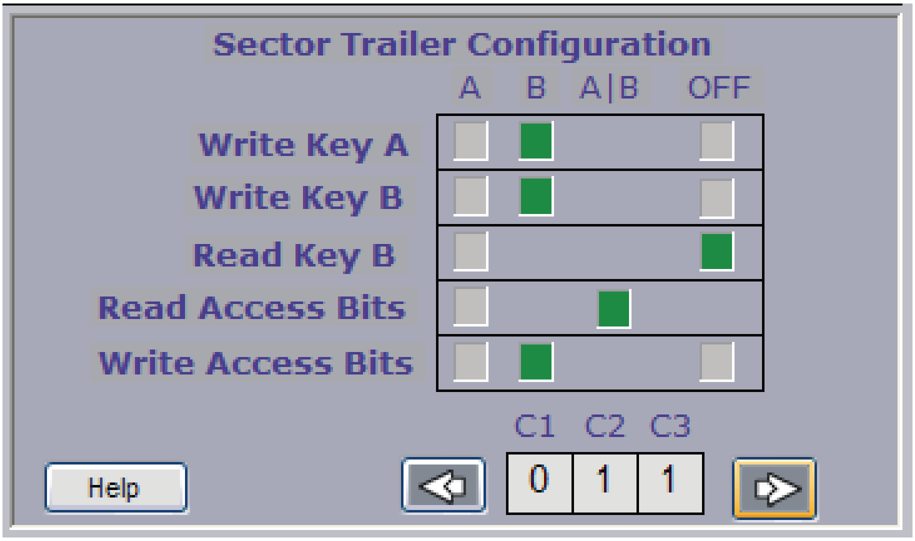
	A	B	A|B	OFF
Write Key A	🟩	□		□
Write Key B	🟩	□		□
Read Key B	🟩			□
Read Access Bits	🟩		□	
Write Access Bits	🟩	□		□

Transport Setting

C1	C2	C3
0	0	1

How you access the Sector Trailer For Read or Write Access is determined by the Sector Trailer Configuration. From this setting, you can choose the configuration that suits your current Sector. Click one of the Arrow Buttons to select one of the 8 types of configuration.

The current configuration 001 shown indicates that using Key A, you can Write Key A, Write Key B, Read Key B, Read the Access Bits, or Write the Access Bits.



Sector Trailer Configuration

	A	B	A|B	OFF
Write Key A	□	🟩		□
Write Key B	□	🟩		□
Read Key B	□			🟩
Read Access Bits	□		🟩	
Write Access Bits	□	🟩		□

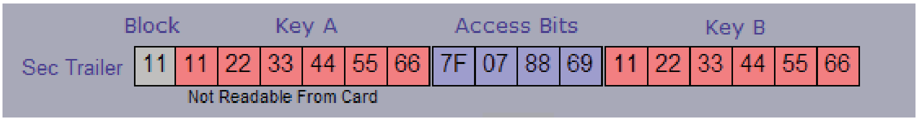
C1	C2	C3
0	1	1

In a different Sector Trailer Configuration such as 011, you can see that Key B will give us access to Write Key A, Write Key B, and Write Access Bits.

The A/B setting here means that you can use either Key A or Key B to "Read Access Bits." A setting of "OFF" means that no access is granted, in this case, to Read Key B.

Note: If you are going to use the B key as an access key for a Value or Data Block, you must configure the Read Key B to OFF. If the B key is readable it is not usable as a key. The six bytes of the B key area can be used as extra data storage if it is not needed as a key.

Authentication Keys

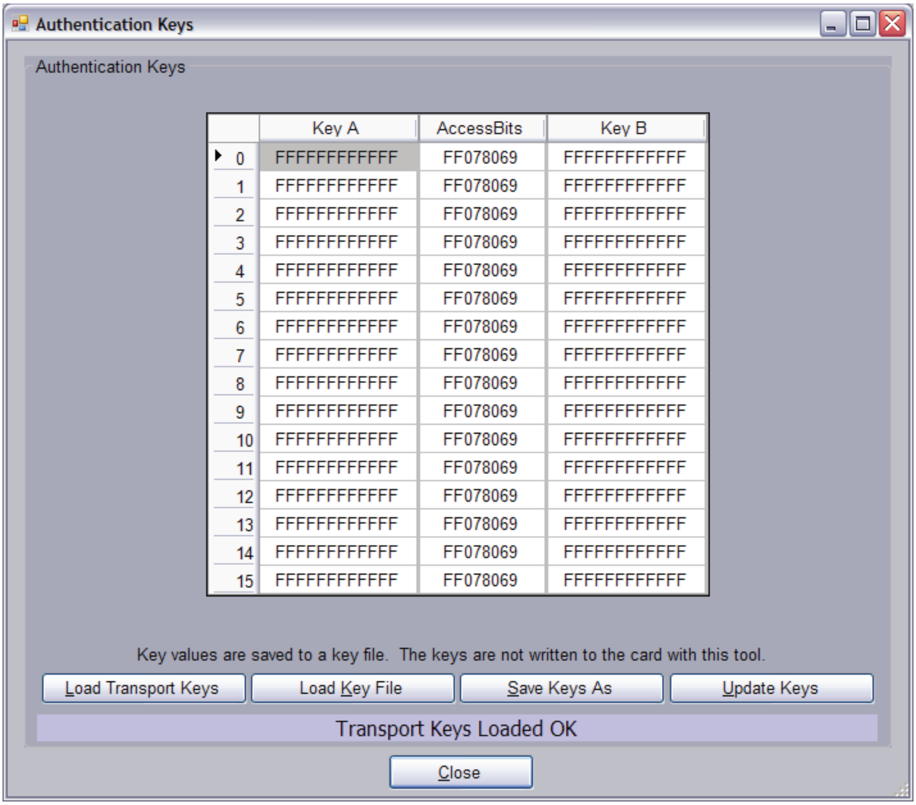


	Block	Key A						Access Bits				Key B					
Sec Trailer	11	11	22	33	44	55	66	7F	07	88	69	11	22	33	44	55	66

Not Readable From Card

Sectors that require authentication must be authenticated with a secret Key A or Key B depending on the configuration of the current sector's Access Bits. Choose the radio button corresponding to the Key Set you require for your project. The default Transport Keys (FF FF FF FF FF FF) are the keys that are provided by default from CardLogix. Once keys are changed on a card, you will need to select either the Current Project File (Current CPF) or select custom keys from a Key File by selecting Key File and clicking the box here to choose the file.

Custom Key Files



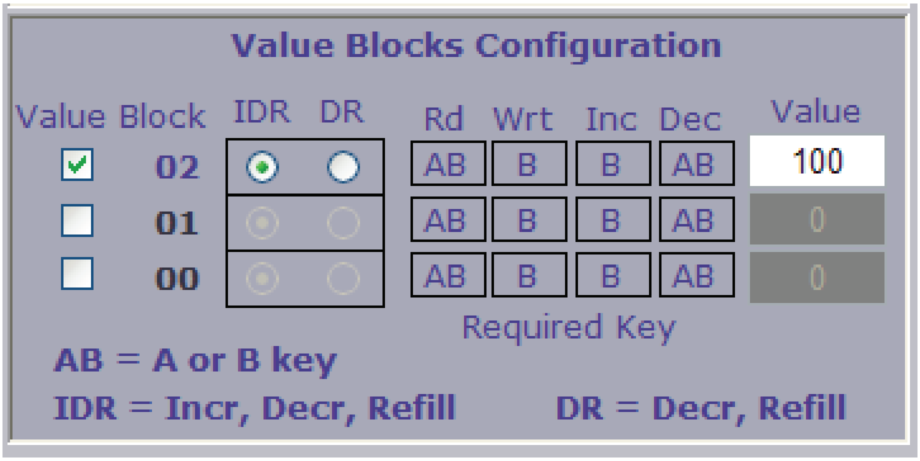
	Key A	AccessBits	Key B
▸ 0	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
1	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
2	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
3	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
4	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
5	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
6	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
7	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
8	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
9	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
10	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
11	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
12	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
13	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
14	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
15	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF

Key values are saved to a key file. The keys are not written to the card with this tool.

Transport Keys Loaded OK

To change the authentication keys, go to the menu item **`Card Functions / Authentication Keys`**. From here you can change Key A and Key B settings for the sectors of a new custom key file. Remember, the settings here are not directly written to the card, but are instead saved to a key file that can be selected later.

Value Blocks



Value Blocks Configuration

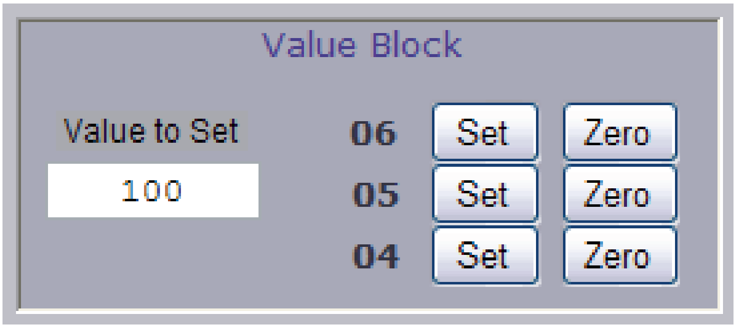
Value Block		IDR	DR	Rd	Wrt	Inc	Dec	Value
☑	02	●	○	AB	B	B	AB	100
□	01	○	○	AB	B	B	AB	0
□	00	○	○	AB	B	B	AB	0

Required Key

AB = A or B key

IDR = Incr, Decr, Refill DR = Decr, Refill

Each data block in a sector can be either a standard data block that contains hex values of data, or a value block. A value block is a special block that contains values that can have the abliity to increment, decrement, or refill. From this configuration block, mark which corresponding block that you would like to set as a value block. You can also choose whether this value block has increment/decrement/refill(IDR) ability, or just decrement/refill (DR).



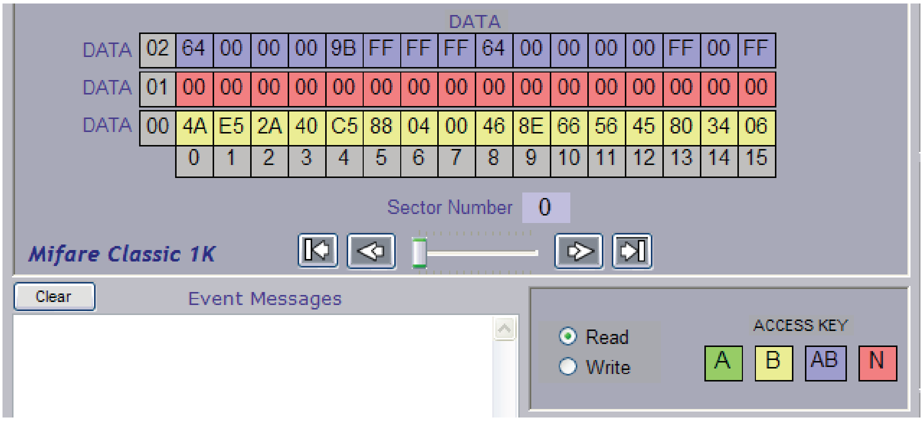
Value Block

Value to Set: 100

06	Set	Zero
05	Set	Zero
04	Set	Zero

To set the value of a value block, use the tool shown above to type in the value, then click set for the appropriate block number. The value will be placed in the value blocks configuration. You can also click zero to zero out a corresponding block.

Data Blocks



DATA

		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
DATA	02	64	00	00	00	9B	FF	FF	FF	64	00	00	00	00	FF	00	FF
DATA	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	00	4A	E5	2A	40	C5	88	04	00	46	8E	66	56	45	80	34	06

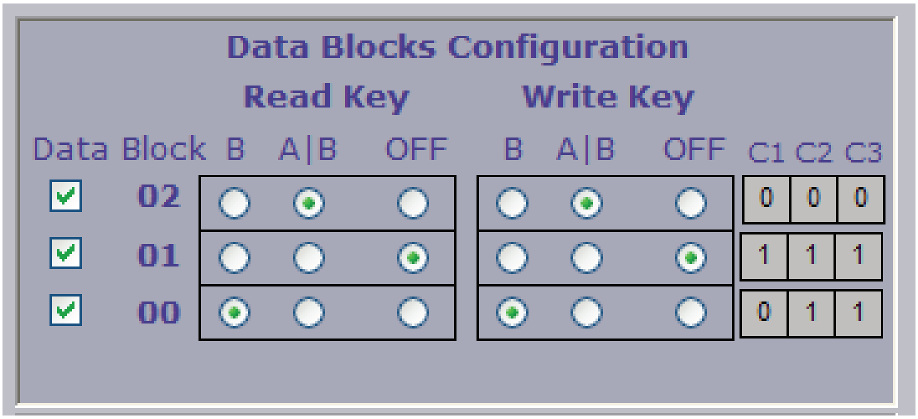
Sector Number 0

● Read
○ Write

ACCESS KEY: A | B | AB | N

The block colors above show the three data blocks for this sector. The colors represent the required key needed to read each data block. In this example, data block 00 is readable with key B (yellow color). Data block 01 is not readable (red color). Data block 02 is readable with either key A or key B (purple color) To view the access key requirements for writing, change the selector from Read to Write.

Data Blocks Configuration



Data Blocks Configuration

	Data Block	Read Key B	Read Key A|B	Read Key OFF	Write Key B	Write Key A|B	Write Key OFF	C1	C2	C3
☑	02	○	●	○	○	●	○	0	0	0
☑	01	○	○	●	○	○	●	1	1	1
☑	00	●	○	○	●	○	○	0	1	1

If you have a data block in your current sector, you can use this tool to set its access properties. You can choose from the following configurations:

- Write with B Key
- Write with A or B Key
- Write Access OFF
- Read with B Key
- Read with A or B Key
- Read Access Off

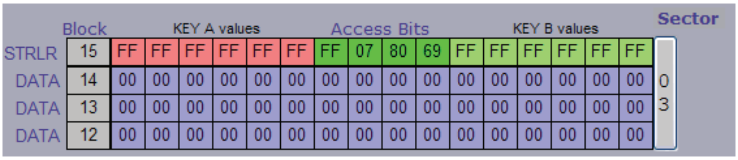
MIFARE® Plus 2K

Overview of the MIFARE® Plus 2K

The MIFARE® Plus 2K is a contactless smart card, with 2K of EEPROM of data. The card features secure access and is backward compatible with the MIFARE® Classic. The MIFARE® Plus 2K is a secure card with up to 64 Keys protecting the 32 sectors of user data.

Typical applications where the MIFARE® Plus 2K would be useful include: Transportation, stored value/loyalty programs, secure area access, and Identification.

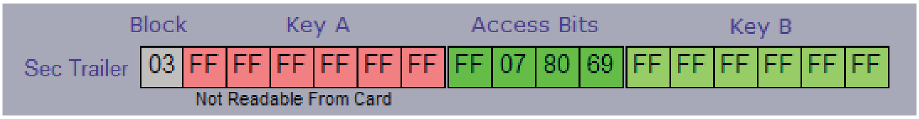
Card Structure



	Block	KEY A values						Access Bits				KEY B values						Sector
STRLR	15	FF	FF	FF	FF	FF	FF	FF	07	80	69	FF	FF	FF	FF	FF	FF	03
DATA	14	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	13	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	12	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

The MIFARE® Plus 2K consists of 32 sectors of data. Each sector consists of a sector trailer, and 3 blocks of user data.

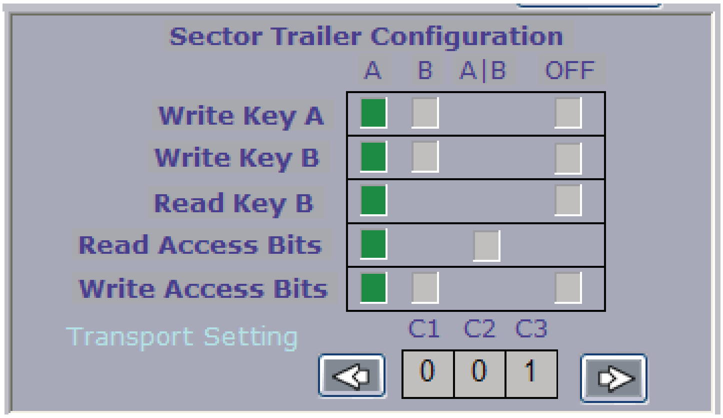
Sector Trailer



	Block	Key A						Access Bits				Key B					
Sec Trailer	03	FF	FF	FF	FF	FF	FF	FF	07	80	69	FF	FF	FF	FF	FF	FF

Not Readable From Card

A Sector Trailer is Block 03 of each Sector on a MIFARE® Plus 2K. A Sector Trailer consists of **Key A**, **Access Bits**, and and optional **Key B**. Key A is an access key, that allows reading or writing of the current Sector based on the user settings of the Access Bits. Key B can also be used as an access key for certain rights determined by the Access Bits of the card. Or it can also be used for additional user data.



Sector Trailer Configuration

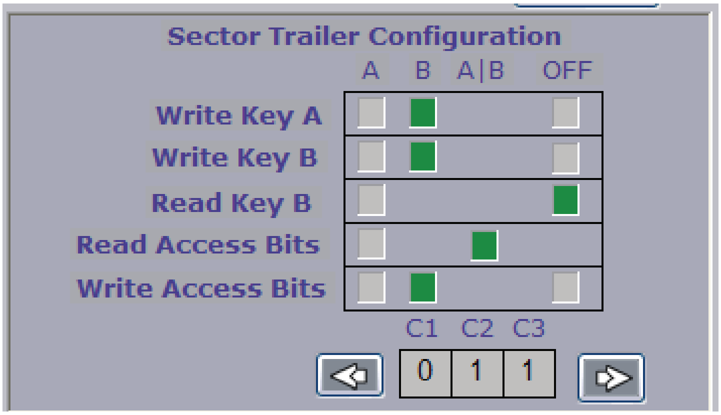
	A	B	A|B	OFF
Write Key A	🟩	□		□
Write Key B	🟩	□		□
Read Key B	🟩			□
Read Access Bits	🟩		□	
Write Access Bits	🟩	□		□

Transport Setting

C1	C2	C3
0	0	1

How you access the Sector Trailer for read or write access is determined by the Sector Trailer. The Sector Trailer Configuration utility allows users to easily change the sector trailer to their needs. You can change the value of the sector trailer by clicking on the arrow buttons to select one of 8 configuration settings.

The current configuration setting of 001 shown indicates that you can use Key A to read Key B, Write to Key A or B. In addition, with Key A you can allow reading or writing to the Access Bits.



Sector Trailer Configuration

	A	B	A|B	OFF
Write Key A	□	☑		□
Write Key B	□	☑		□
Read Key B	□			☑
Read Access Bits	□		☑	
Write Access Bits	□	☑		□

C1	C2	C3
0	1	1

With a different Sector Trailer Configuration, such as 011, we can see that Key B will give use access to Write Key A, Write Key B, and Write Access Bits. The **A/B** setting here indicates we can use either Key A or Key B to read the access bits. A setting of **OFF** means that No Access is granted, in this case, to Read Key B.

Note: If you are going to use Key B as an access Key then you must set Read Key B to OFF. If Key B is readable it is not usable as a key. If it is readable, Key B can only be used as additional user data.

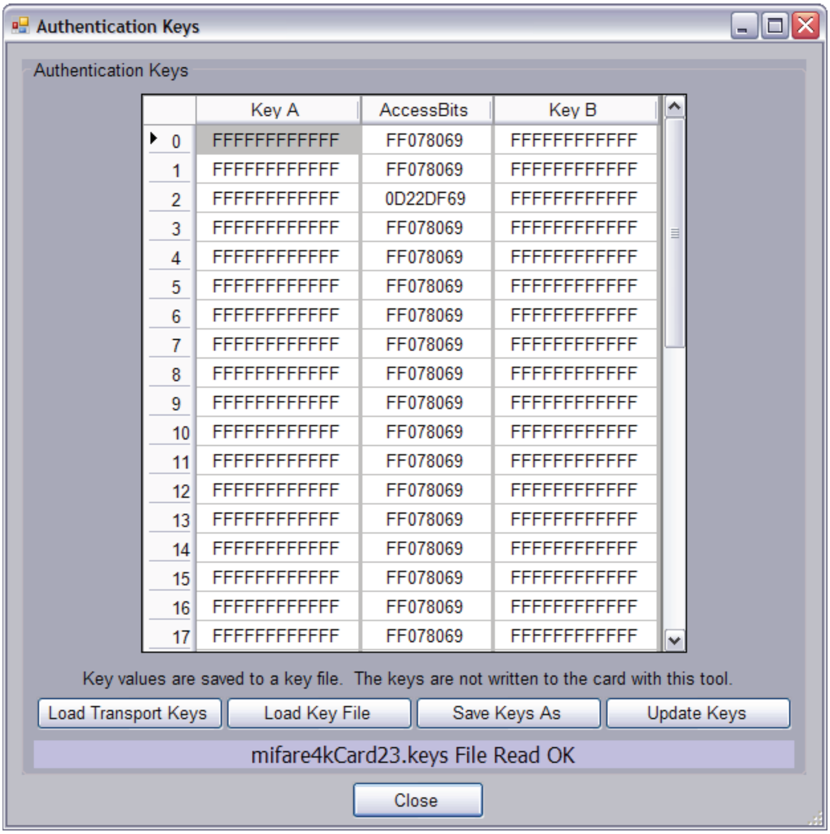
Authentication Keys

Authentication Keys

Sectors that require authentication must be authenticated with a secret Key A or Key B depending on the configuration of the current sector's Access Bits. Choose the radio button corresponding to the Key Set you require for your project. The default Transport Keys (FF FF FF FF FF FF) are the keys that are provided by default from CardLogix. Once keys are changed on a card, you will need to select either the Current Project File (Current CPF) or select custom keys from a Key File by selecting Key File and clicking the box here to choose the file.

Custom Key Files

In order to write to a card that has previously been modified with custom keys, you need to use a custom key file. To change the authentication keys, go to the menu item `Card Functions / Authentication Keys` and view the following screen. From here you can change Key A and Key B settings for the sectors of a custom key file.



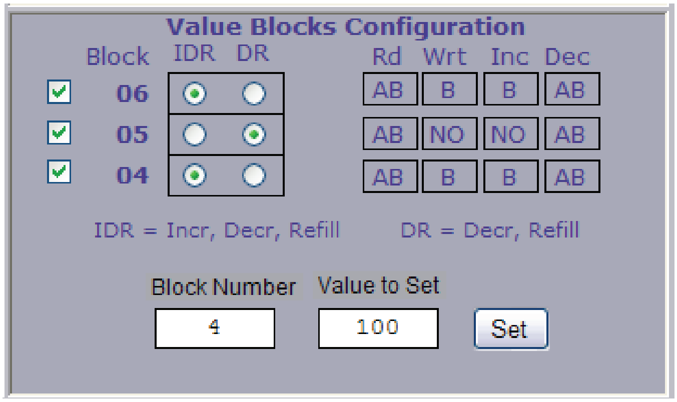
	Key A	AccessBits	Key B
0	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
1	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
2	FFFFFFFFFFFF	0D22DF69	FFFFFFFFFFFF
3	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
4	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
5	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
6	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
7	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
8	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
9	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
10	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
11	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
12	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
13	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
14	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
15	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
16	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
17	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF

Key values are saved to a key file. The keys are not written to the card with this tool.

mifare4kCard23.keys File Read OK

From here you can specify Key A and Key B settings for all sectors of a personalized MIFARE® Plus 2K card that will be saved or loaded later. Remember, the settings here are not written to the card directly, but are instead saved to a key file that can be selected later.

Value Blocks



Value Blocks Configuration

	Block	IDR	DR	Rd	Wrt	Inc	Dec
☑	06	●	○	AB	B	B	AB
☑	05	○	●	AB	NO	NO	AB
☑	04	●	○	AB	B	B	AB

IDR = Incr, Decr, Refill   DR = Decr, Refill

Block Number: 4   Value to Set: 100

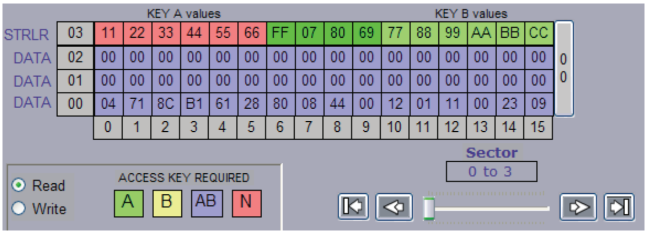
Each data block in a sector can be either a standard data block that contains hex values of data or it can be a Value Block. A Value Block is a special block that contains values that can have the ability to Increment, Decrement, or Refill. From the Value Blocks Configuration, mark which corresponding block that you would like to set as a Value Block. Choose whether you want this Value Block to Increment/Decrement/Refill or just Decrement/Refill.

To set the value of a Value Block enter the corresponding Block Number, the value you want to set it and click the Set button.

Writing or incrementing value blocks requires the use of the B key. In order to use the B key it must be unreadable. To make sure that the B key is unreadable, set Read Key B to OFF in the Sector Trailer Configuration.

***If Key B is readable it is NOT usable as a key.**

Data Blocks



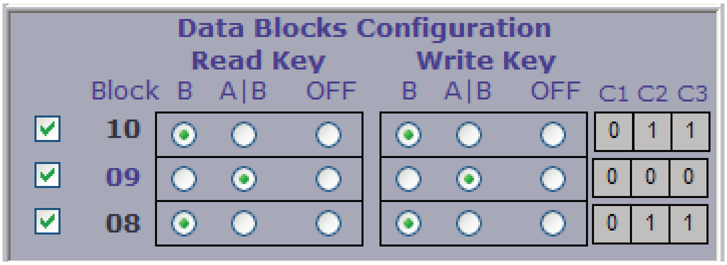
		KEY A values										KEY B values					
STRLR	03	11	22	33	44	55	66	FF	07	80	69	77	88	99	AA	BB	CC
DATA	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	00	04	71	8C	B1	61	28	80	08	44	00	12	01	11	00	23	09
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

● Read
○ Write

ACCESS KEY REQUIRED: A B AB N

Sector: 0 to 3

A card's data blocks are where primary data is stored. In the example shown above, the blue data boxes indicate that the data can be read with either the A or the B key of the matching sector. If you want to see which key is required to Write to this sector's data area, choose the Write radio selection.



Data Blocks Configuration

		Read Key			Write Key					
	Block	B	A|B	OFF	B	A|B	OFF	C1	C2	C3
☑	10	●	○	○	●	○	○	0	1	1
☑	09	○	●	○	○	●	○	0	0	0
☑	08	●	○	○	●	○	○	0	1	1

If you have a Data Block in your current sector you can use this tool to set it's access properties. You can choose from the following configurations:

- Write with B Key
- Write with A or B Key
- Write Access Off
- Read with B Key
- Read with A or B Key
- Read Access Off

Finalizing Your Card

After you have set your card configuration values, write your configuration to a MIFARE® Plus 2K card using the File menu by selecting **`File>Save to Card`**. Using your application program, thoroughly test all card functions.

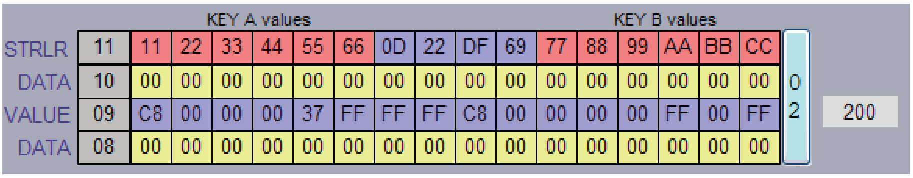
Once you are satisfied that your card structure is working correctly with your application, the next step is to send your card project file (CPF) to CardLogix to have 'check cards' made. Save your configuration settings to your project file using the file menu **`File>Save Project File`** or **`File>Save Project File As`**…

Send your project file to CardLogix. Using your project file, check cards will be made and returned to you, along with a sign-off sheet. After you sign off on the check card we can advance your design to full production. You should also contact your CardLogix sales person for information on the personalization features of your card.

If your cards contain information that needs to be secure, such information may need to be transferred using an encryption method, PGP keys or other secure means of data transfer. Please contact your CardLogix representative for further information.

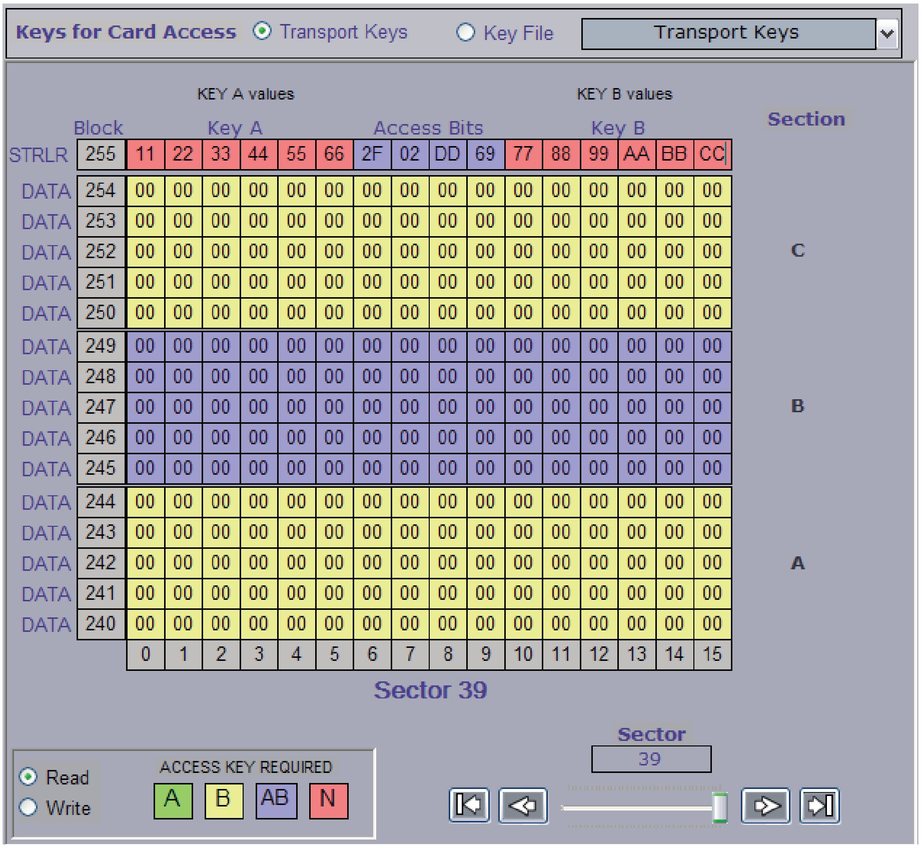
MIFARE® 4K (CLXRN032KN4) Cards

Card Structure



		KEY A values										KEY B values							
STRLR	11	11	22	33	44	55	66	0D	22	DF	69	77	88	99	AA	BB	CC		
DATA	10	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0	
VALUE	09	C8	00	00	00	37	FF	FF	FF	C8	00	00	00	00	FF	00	FF	2	200
DATA	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00		

A MIFARE® 4K (CLXRN032KN4) card consists of 40 sectors. The first 32 sectors (0 through 31) contain 3 data blocks and 1 block for a sector trailer configuration.



Keys for Card Access ● Transport Keys ○ Key File | Transport Keys

	Block	KEY A values Key A						Access Bits				KEY B values Key B						Section
STRLR	255	11	22	33	44	55	66	2F	02	DD	69	77	88	99	AA	BB	CC	
DATA	254	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	253	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	252	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	C
DATA	251	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	249	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	248	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	247	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B
DATA	246	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	245	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	244	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	243	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	242	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	A
DATA	241	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

Sector 39

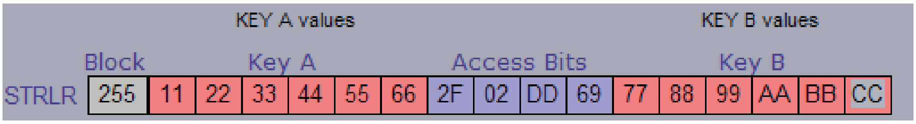
● Read
○ Write

ACCESS KEY REQUIRED: A B AB N

Sector: 39

The last 8 sectors (32 through 39) are structured with 3 groups of 5 data blocks, and 1 block for a sector trailer configuration.

Sector Trailer

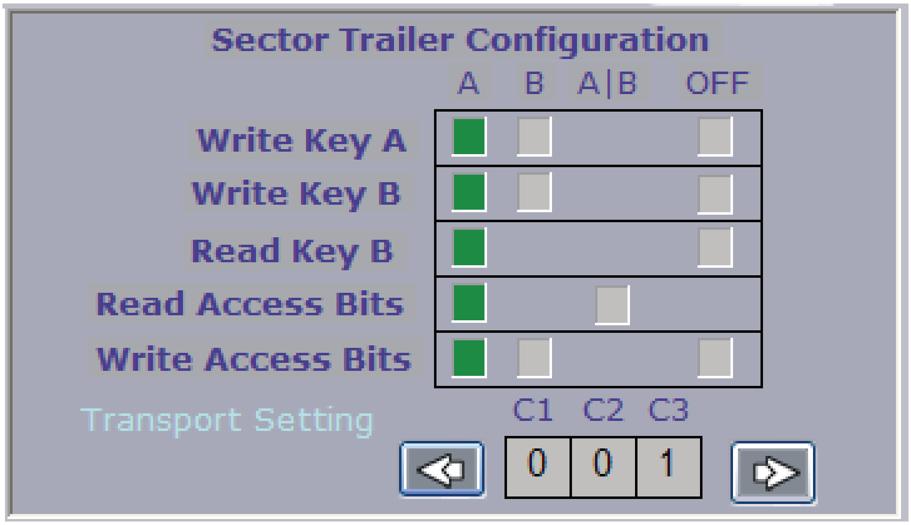


KEY A values KEY B values

	Block	Key A						Access Bits				Key B					
STRLR	255	11	22	33	44	55	66	2F	02	DD	69	77	88	99	AA	BB	CC

A sector trailer consists of two access keys, called Key A and Key B, and a 4 byte section labeled Access Bits that determines access privileges to blocks in the current sector.

Sector Trailer Configuration



Sector Trailer Configuration

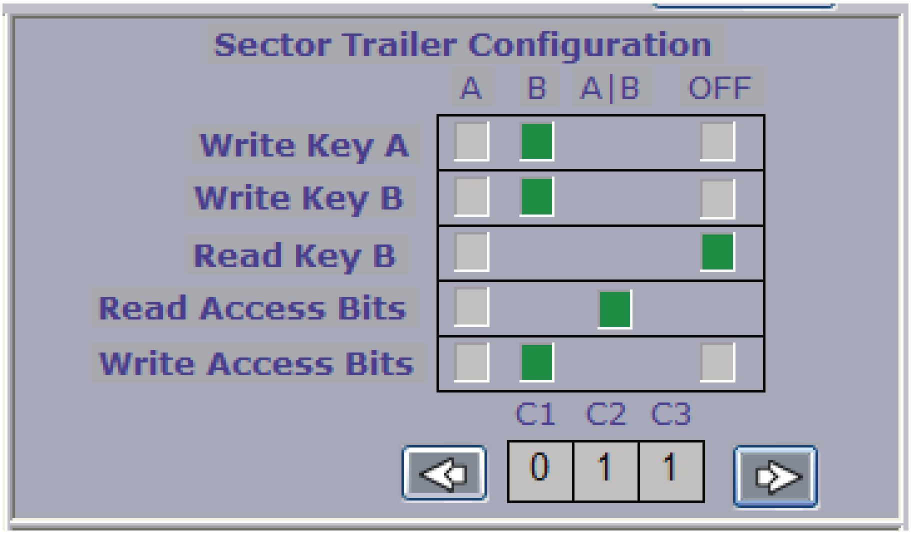
	A	B	A|B	OFF
Write Key A	■	□		□
Write Key B	■	□		□
Read Key B	■			□
Read Access Bits	■		□	
Write Access Bits	■	□		□

Transport Setting

C1	C2	C3
0	0	1

How you access the Sector Trailer For Read or Write Access is determined by the sector trailer configuration. From this setting, you can choose the configuration that suits your current sector. Click one of the arrow buttons to select one of the 8 types of configuration.

The current configuration 001 shown indicates that using Key A, you can Write Key A, Write Key B, Read Key B, Read the Access Bits, or Write the Access Bits.



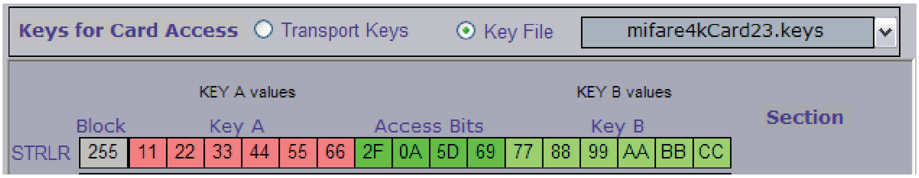
Sector Trailer Configuration

	A	B	A|B	OFF
Write Key A	□	🟩		□
Write Key B	□	🟩		□
Read Key B	□			🟩
Read Access Bits	□		🟩	
Write Access Bits	□	🟩		□

C1	C2	C3
0	1	1

In a different Sector Trailer Configuration such as 011, we can see that Key B will give us access to Write Key A, Write Key B, and Write Access Bits. The A/B setting here means that you can use either Key A or Key B to "Read Access Bits." A setting of "OFF" means that no access is granted, in this case, to read key B.

Authentication Keys

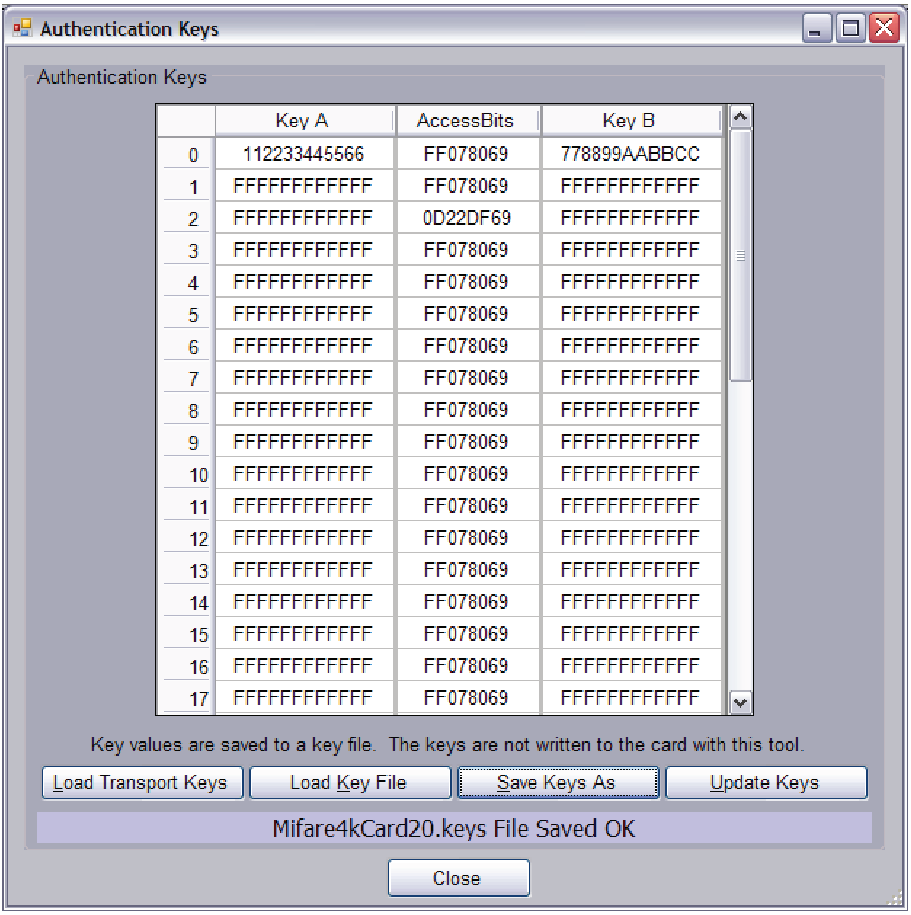


Keys for Card Access ○ Transport Keys ● Key File mifare4kCard23.keys

		KEY A values										KEY B values						
	Block	Key A						Access Bits				Key B						Section
STRLR	255	11	22	33	44	55	66	2F	0A	5D	69	77	88	99	AA	BB	CC	

A card must be authenticated with a secret Key A or Key B depending on the configuration of the current sector's access bits. Choose the radio button corresponding to the key set you require for your project. The default transport keys (FF FF FF FF FF FF) are the keys that are provided from the card manufacturer. Once keys are changed on a card, you will need to select either the current project file (current CPF) or select custom keys from a key file by selecting Key File and clicking the box here to choose the file.

Custom Key Files



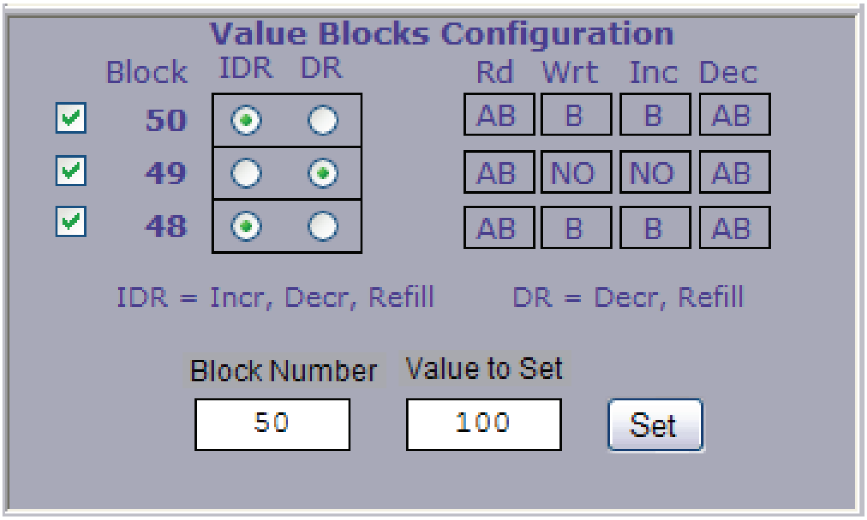
	Key A	AccessBits	Key B
0	112233445566	FF078069	778899AABBCC
1	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
2	FFFFFFFFFFFF	0D22DF69	FFFFFFFFFFFF
3	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
4	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
5	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
6	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
7	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
8	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
9	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
10	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
11	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
12	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
13	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
14	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
15	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
16	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
17	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF

Key values are saved to a key file. The keys are not written to the card with this tool.

Mifare4kCard20.keys File Saved OK

Custom key files are needed when writing to a card whose keys have been previously modified. To change the authentication keys, go to the menu item **`Card Functions/Authentication Keys`**. From here you can change Key A and Key B settings for the sectors of a custom key file.

Value Blocks



Value Blocks Configuration

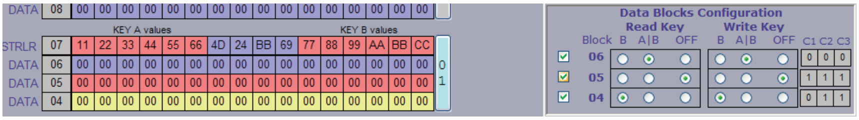
	Block	IDR	DR	Rd	Wrt	Inc	Dec
☑	50	●	○	AB	B	B	AB
☑	49	○	●	AB	NO	NO	AB
☑	48	●	○	AB	B	B	AB

IDR = Incr, Decr, Refill DR = Decr, Refill

Block Number	Value to Set	
50	100	Set

Each data block in a sector can be either a standard data block that contains hex values of data, or a value block. A Value Block is a special block that contains values that can have the ability to increment, decrement, or refill. From this configuration block, mark which corresponding block that you would like to set as a value block. You can also choose whether this Value Block has increment / decrement / refill (IDR) ability, or just decrement / refill (DR).

Data Blocks

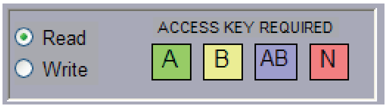


DATA	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
		KEY A values										KEY B values					
STRLR	07	11	22	33	44	55	66	4D	24	BB	69	77	88	99	AA	BB	CC
DATA	06	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	05	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	04	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

0
1

Data Blocks Configuration

		Read Key			Write Key					
	Block	B	A|B	OFF	B	A|B	OFF	C1	C2	C3
☑	06	○	●	○	○	●	○	0	0	0
☑	05	○	○	●	○	○	●	1	1	1
☑	04	●	○	○	●	○	○	0	1	1



● Read
○ Write

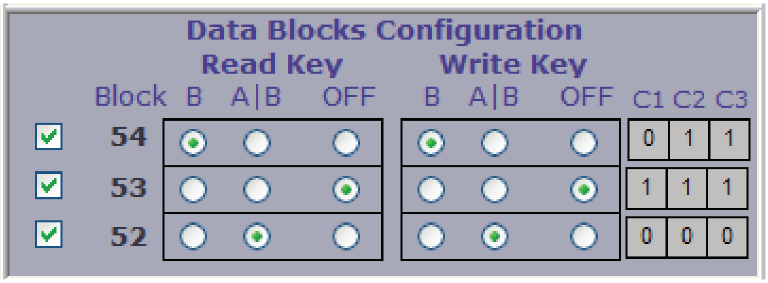
ACCESS KEY REQUIRED

A B AB N

The image above shows three data blocks for this sector. The block colors represent the required key needed to read each data block. In this example, data block 04 is readable with key B (yellow color). Data block 05 is not readable (red color). Data block 06 is readable with either key A or key B (purple color). To view the access key requirements for writing, change the selector from Read to Write.

Special Note: In the first 32 sectors of a MIFARE® 4K, the access bits affect its own sector trailer and 3 blocks of data. In the last 8 sectors of the card, the access bits affect 1 sector trailer and 3 groups of 5 blocks of data at a time.

Data Blocks Configuration



Data Blocks Configuration

		Read Key			Write Key					
	Block	B	A|B	OFF	B	A|B	OFF	C1	C2	C3
☑	54	●	○	○	●	○	○	0	1	1
☑	53	○	○	●	○	○	●	1	1	1
☑	52	○	●	○	○	●	○	0	0	0

If you have a data block in your current sector, you can use this tool to set its access properties. You can choose from the following configurations:

- Write with B Key
- Write with A or B Key
- Write Access OFF
- Read with B Key
- Read with A or B Key
- Read Access Off

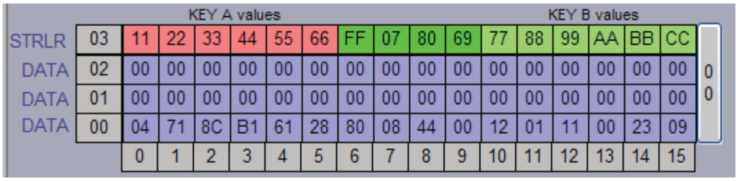
MIFARE® Plus 4K

Overview of the MIFARE® Plus 4K

The MIFARE® Plus 4K is a contactless smart card, with a 4K EEPROM. The card features secure access and is backward compatible with the MIFARE® Classic. The MIFARE® Plus 4K is a secure card with up to 72 Keys protecting the 40 sectors of user data.

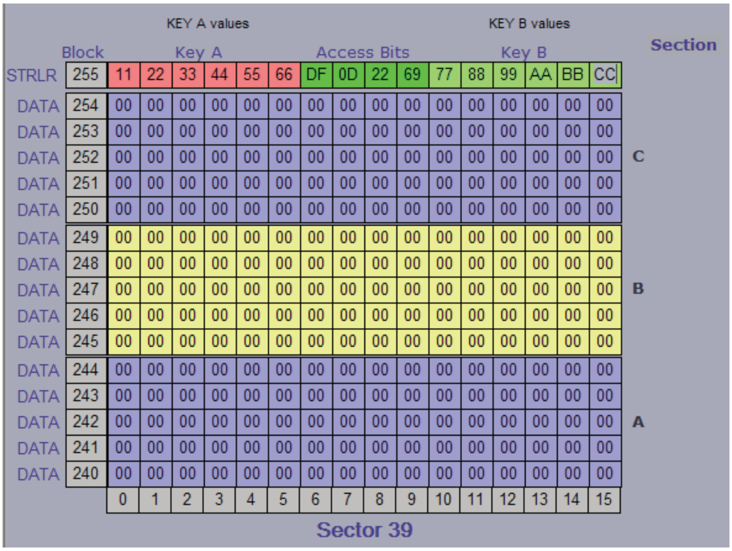
Typical applications where the MIFARE® Plus 4K would be useful include: Transportation, stored value/loyalty programs, secure area access, and Identification.

Card Structure



		KEY A values										KEY B values					
STRLR	03	11	22	33	44	55	66	FF	07	80	69	77	88	99	AA	BB	CC
DATA	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
DATA	00	04	71	8C	B1	61	28	80	08	44	00	12	01	11	00	23	09
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

The MIFARE® Plus 4K consists of 40 sectors of data. The first 32 sectors consist of a sector trailer, and 3 blocks of user data as shown.

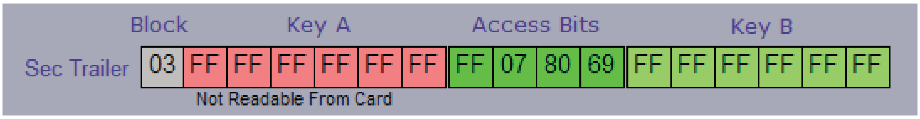


		KEY A values										KEY B values						
	Block	Key A						Access Bits				Key B						Section
STRLR	255	11	22	33	44	55	66	DF	0D	22	69	77	88	99	AA	BB	CC	
DATA	254	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	253	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	252	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	C
DATA	251	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	250	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	249	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	248	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	247	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	B
DATA	246	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	245	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	244	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	243	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	242	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	A
DATA	241	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
DATA	240	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

Sector 39

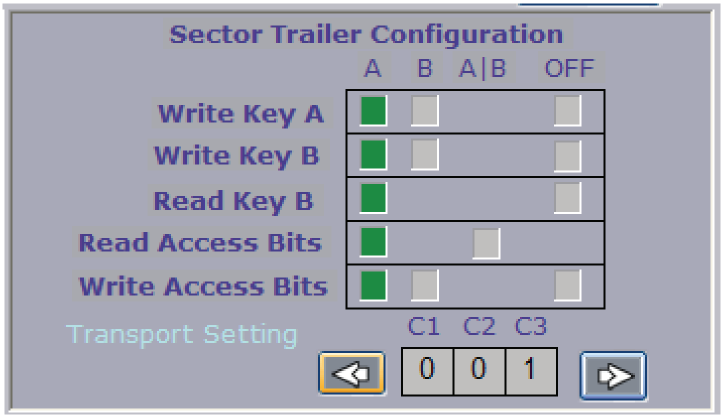
The final 8 sectors of a MIFARE® Plus 4K consist of a sector trailer and 15 blocks of user data.

Sector Trailer



	Block	Key A						Access Bits				Key B					
Sec Trailer	03	FF	FF	FF	FF	FF	FF	FF	07	80	69	FF	FF	FF	FF	FF	FF
		Not Readable From Card															

A Sector Trailer is the final block of each Sector on a MIFARE® Plus 4K. A Sector Trailer consists of **Key A**, **Access Bits**, and and optional **Key B**. Key A is an access key, that allows reading or writing of the current Sector based on the user settings of the Access Bits. Key B can also be used as an access key for certain rights determined by the Access Bits of the card. Or it can also be used for additional user data.



Sector Trailer Configuration

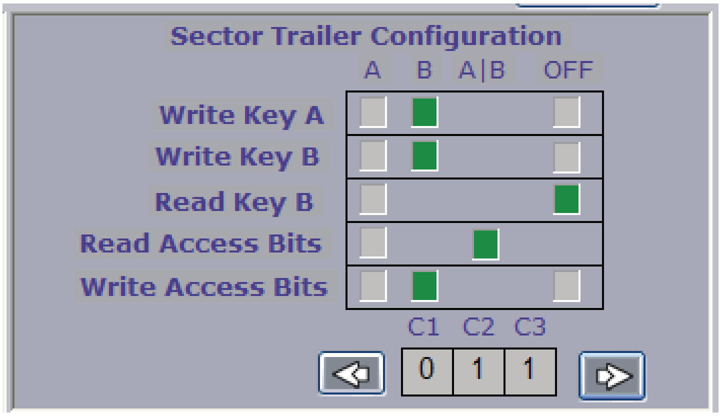
	A	B	A|B	OFF
Write Key A	🟩	□		□
Write Key B	🟩	□		□
Read Key B	🟩			□
Read Access Bits	🟩		□	
Write Access Bits	🟩	□		□

Transport Setting

C1	C2	C3
0	0	1

How you access the Sector Trailer for read or write access is determined by the Sector Trailer. The Sector Trailer Configuration utility allows users to easily change the Sector Trailer to their needs. You can change the value of the Sector Trailer by clicking on the arrow buttons to select one of 8 configuration settings.

The current configuration setting of 001 shown indicates that we can use Key A to read Key B, Write to Key A or B. In addition, key A will allow reading or writing to the Access Bits.



Sector Trailer Configuration

	A	B	A|B	OFF
Write Key A	□	🟩		□
Write Key B	□	🟩		□
Read Key B	□			🟩
Read Access Bits	□		🟩	
Write Access Bits	□	🟩		□

C1	C2	C3
0	1	1

With a different Sector Trailer Configuration, such as 011, we can see that Key B will give use access to Write Key A, Write Key B, and Write Access Bits. The **A/B** setting here indicates we can use either Key A or Key B to read the access bits. A setting of **OFF** means that No Access is granted, in this case, to Read Key B.

Note: If you are going to use Key B as an access Key then you must set Read Key B to OFF. If Key B is readable it is not usable as a key. If it is readable, Key B can only be used as additional user data.

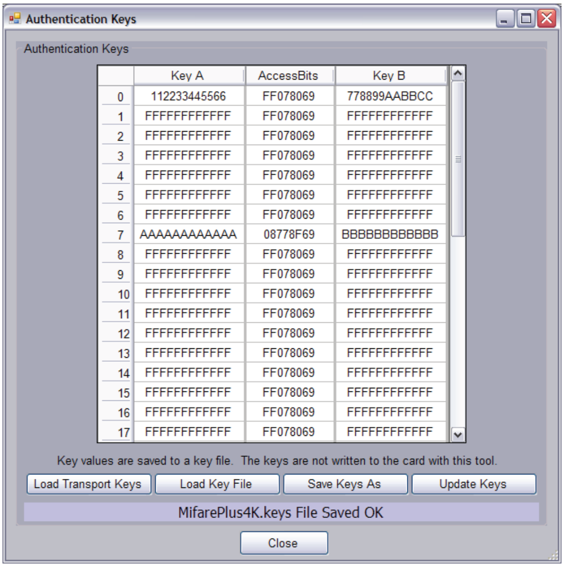
Authentication Keys

Authentication Keys

Sectors that require authentication must be authenticated with a secret Key A or Key B depending on the configuration of the current sector's Access Bits. Choose the radio button corresponding to the Key Set you require for your project. The default Transport Keys (FF FF FF FF FF FF) are the keys that are provided by default from CardLogix. Once keys are changed on a card, you will need to select either the Current Project File (Current CPF) or select custom keys from a Key File by selecting Key File and clicking the box here to choose the file.

Custom Key Files

In order to write to a card that has previously been modified with custom keys, you need to use a custom key file. To change the authentication keys, go to the menu item **`Card Functions / Authentication Keys`** and view the following screen.



Authentication Keys

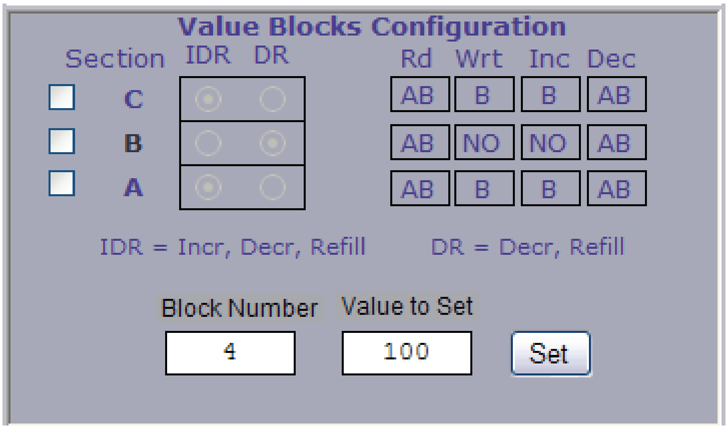
	Key A	AccessBits	Key B
0	112233445566	FF078069	778899AABBCC
1	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
2	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
3	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
4	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
5	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
6	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
7	AAAAAAAAAAAA	08778F69	BBBBBBBBBBBB
8	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
9	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
10	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
11	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
12	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
13	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
14	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
15	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
16	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF
17	FFFFFFFFFFFF	FF078069	FFFFFFFFFFFF

Key values are saved to a key file. The keys are not written to the card with this tool.

MifarePlus4K.keys File Saved OK

From here you can specify Key A and Key B settings for all sectors of a custom key file of a personalized MIFARE® Plus 4K card that will be saved or loaded later. Remember, the settings here are not written to the card directly, but are instead saved to a key file that can be selected later.

Value Blocks



Value Blocks Configuration

	Section	IDR	DR	Rd	Wrt	Inc	Dec
□	C	●	○	AB	B	B	AB
□	B	○	●	AB	NO	NO	AB
□	A	●	○	AB	B	B	AB

IDR = Incr, Decr, Refill   DR = Decr, Refill

Block Number: 4   Value to Set: 100

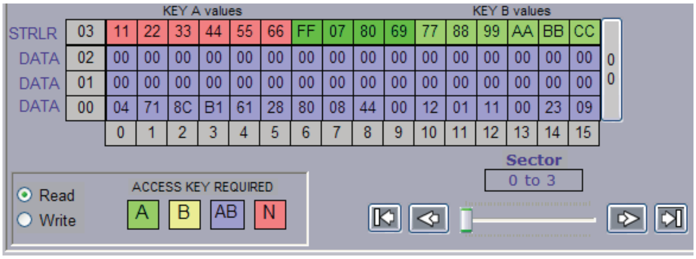
Each data block in a sector can be either a standard data block that contains hex values of data or it can be a Value Block. A Value Block is a special block that contains values that can have the ability to Increment, Decrement, or Refill. From the Value Blocks Configuration, mark which corresponding block that you would like to set as a Value Block. Choose whether you want this Value Block to Increment/Decrement/Refill or just Decrement/Refill.

To set the value of a Value Block enter the corresponding Block Number, the value you want to set it and click the Set button.

Writing or incrementing value blocks requires the use of the B key. In order to use the B key it must be unreadable. To make sure that the B key is unreadable, set Read Key B to OFF in the Sector Trailer Configuration

***If Key B is readable it is NOT usable as a key.**

Data Blocks



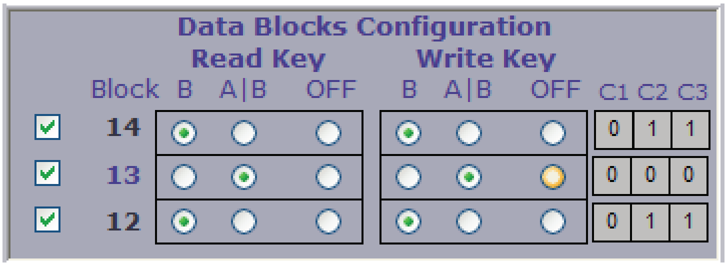
		KEY A values										KEY B values						
STRLR	03	11	22	33	44	55	66	FF	07	80	69	77	88	99	AA	BB	CC	
DATA	02	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0
DATA	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	0
DATA	00	04	71	8C	B1	61	28	80	08	44	00	12	01	11	00	23	09	
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	

● Read
○ Write

ACCESS KEY REQUIRED: A B AB N

Sector: 0 to 3

A card's data blocks are where primary data is stored. In the example shown above, the blue data boxes indicate that the data can be read with either the A or the B key of the matching sector. If you want to see which key is required to Write to this sector's data area, choose the Write radio selection.



Data Blocks Configuration

		Read Key			Write Key					
	Block	B	A|B	OFF	B	A|B	OFF	C1	C2	C3
☑	14	●	○	○	●	○	○	0	1	1
☑	13	○	●	○	○	●	○	0	0	0
☑	12	●	○	○	●	○	○	0	1	1

If you have a Data Block in your current sector you can use this tool to set it's access properties. You can choose from the following configurations:

- Write with B Key
- Write with A or B Key
- Write Access Off
- Read with B Key
- Read with A or B Key
- Read Access Off

Finalizing Your Card

After you have set your card configuration values, write your configuration to a MIFARE® Plus 4K card using the File menu by selecting **`File>Save to Card.`** Using your application program, thoroughly test all card functions.

Once you are satisfied that your card structure is working correctly with your application the next step is to send your card project file (CPF) to CardLogix to have 'check cards' made. Save your configuration settings to your project file using the file menu **`File>Save Project File`** or **`File>Save Project File As`**…

Send your project file to CardLogix. Using your project file, check cards will be made and returned to you, along with a sign-off sheet. After you sign off on the check card we can advance your design to full production. You should also contact your CardLogix sales person for information on the personalization features of your card.

If your cards contain information that needs to be secure, such information may need to be transferred using an encryption method, PGP keys or other secure means of data transfer. Please contact your CardLogix representative for further information.

MIFARE® 8K DESFire EV1

Overview of the 8K DESFire EV1

The MIFARE® DESFire EV1 cards from CardLogix are ideal for companies wanting to implement secure multi-application smart cards. The cards are ideal for many types of markets where transactional security is required, such as transportation, stored value/loyalty, and access control applications.

The card fully supports fast and highly secure data transmission in a flexible memory organization and is interoperable with existing reader infrastructures. The chip design gives the integrator freedom to abstract their files with very granular control over data security.

The MIFARE® DESFire EV1 is based on open global standards for both contactless interface and cryptographic methods. It is compliant to all 4 levels of ISO/IEC 14443A. The chip is certified to EAL4+. Due to the file-based organization of the card, ISO 7816-4 commands can be executed like a standard CPU contact smart card.

CardLogix DESFire Cards have an on-chip backup management system support, and optional three pass authentication. A DESFire EV1 card can store up to 28 different applications and 32 files per application.

Applications

The 8 KB Non Volatile memory is organized using a flexible file system. This file system allows a maximum of 28 different applications on one MIFARE® DESFire EV1. Each application provides up to 32 files. Each application is represented by a 3 byte Application IDentifier (AID). You can think about the AID as a dedicated file in ISO 7816 Style. For each Application you have the following options:

Rename: This allows you to choose a different AID for the Application. AID values are entered in hex bytes. Each AID must be unique and can range from 000001 to FFFFFF.

Delete: Permanently delete the application and any files it may contain.

Add: Add one of five file types to this application. The next available file ID will automatically be chosen. You can rename the file ID after it has been created. Valid values for a file ID range from 00 to 1F.

Edit Access Keys: Each Application can have up to 14 unique secret keys available. To personalize the values of the application keys, click on the Edit Access Keys button. Key #00 of the application is considered the Application Master Key. The AMK always exists and is critical to certain application features such as changing the app keys once they have been personalized.

File Types

The files within an application can be any of the following types:

Standard Data Files

Plain unformatted user data that can be allocated to a user defined size. This type of file is useful for applications that contain a wide variety of information, in various formats. It is suggested for use where the data in this file is written by once at issuance and does not change since it does not have an anti-tear feature. This type of file would also be most useful if you require the most storage space since there is very little overhead with this particular type of file. Please note for data integrity: This type of file is typically used for write once and read only data.

Backup Data Files

Same as a standard data file, however, there is an anti-tear backup feature which makes it use twice as much storage. This type of file would be useful for instances where a kiosk modifies the user card and there is a chance of the customer removing the card before the transaction has been completed. Contactless cards by their nature typically require this type of backup. This is due to a customers behavior of moving a card in and out of range of the reader antenna that supplies power to the card and maintains the transaction.

Value Files with Backup

Files for storage and manipulation of 32-bit signed integer values within a specified range. This type of file is useful in cases where a single value is needed, one that can fit within a predetermined range. This type of file could be used for a points or value system.

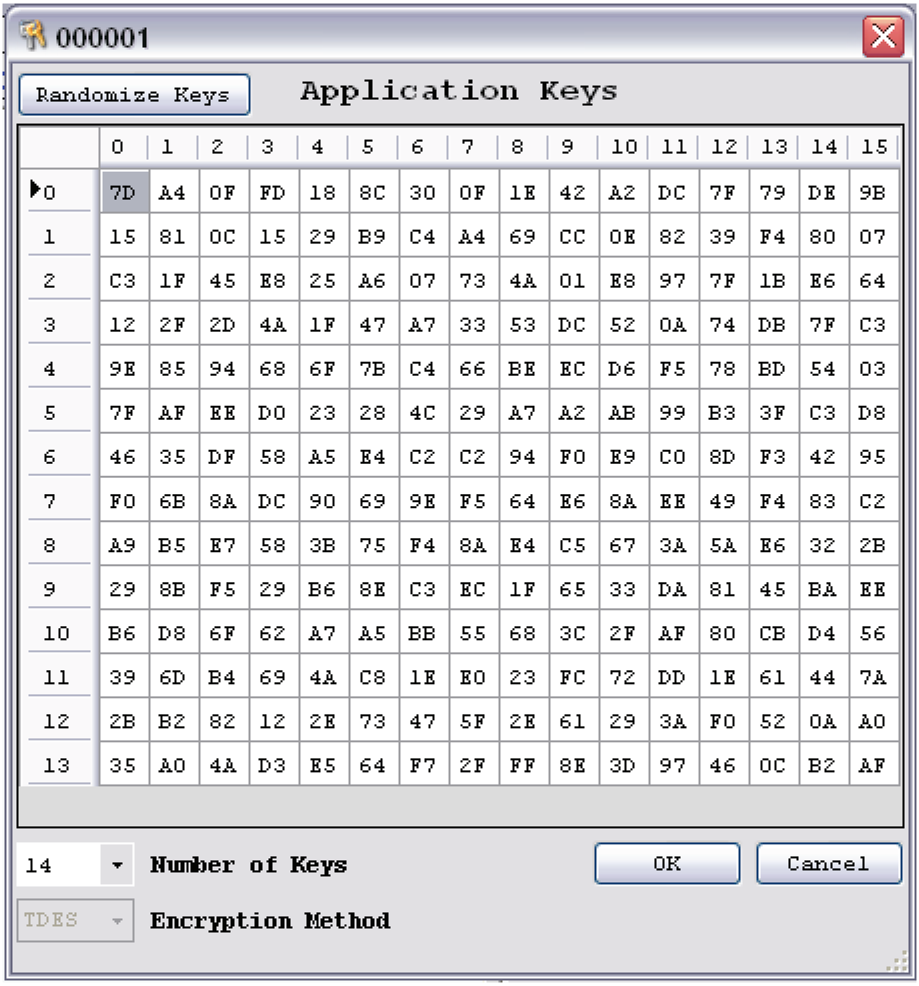
Linear Record Files with Backup

File for storage and manipulation of similar structural data, for example loyalty programs within an existing application on the MIFARE® DESFire EV1; once the file is filled completely with data records, further writing to the file is not possible unless it is cleared.

Cyclic Record Files with Backup

File for multiple storage of similar structural data, for example, logging transactions within an existing application on the MIFARE® DESFire EV1; once the file is completely filled with records, the MIFARE® DESFire EV1 automatically overwrites the oldest record, allowing for continuous memory usage. This is usually used for recent financial transaction records or logging of other card functions to and from a terminal before a card is synced up to a master database.

Application Keys



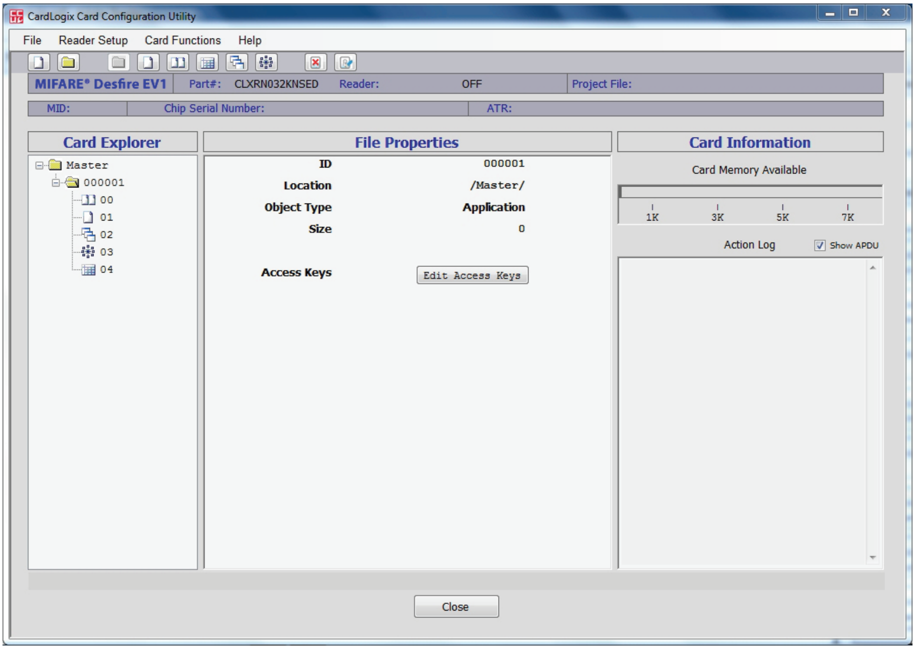
Application Keys

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
▶0	7D	A4	0F	FD	18	8C	30	0F	1E	42	A2	DC	7F	79	DE	9B
1	15	81	0C	15	29	B9	C4	A4	69	CC	0E	82	39	F4	80	07
2	C3	1F	45	E8	25	A6	07	73	4A	01	E8	97	7F	1B	E6	64
3	12	2F	2D	4A	1F	47	A7	33	53	DC	52	0A	74	DB	7F	C3
4	9E	85	94	68	6F	7B	C4	66	BE	EC	D6	F5	78	BD	54	03
5	7F	AF	EE	D0	23	28	4C	29	A7	A2	AB	99	B3	3F	C3	D8
6	46	35	DF	58	A5	E4	C2	C2	94	F0	E9	C0	8D	F3	42	95
7	F0	6B	8A	DC	90	69	9E	F5	64	E6	8A	EE	49	F4	83	C2
8	A9	B5	E7	58	3B	75	F4	8A	E4	C5	67	3A	5A	E6	32	2B
9	29	8B	F5	29	B6	8E	C3	EC	1F	65	33	DA	81	45	BA	EE
10	B6	D8	6F	62	A7	A5	BB	55	68	3C	2F	AF	80	CB	D4	56
11	39	6D	B4	69	4A	C8	1E	E0	23	FC	72	DD	1E	61	44	7A
12	2B	B2	82	12	2E	73	47	5F	2E	61	29	3A	F0	52	0A	A0
13	35	A0	4A	D3	E5	64	F7	2F	FF	8E	3D	97	46	0C	B2	AF

14 Number of Keys

TDES Encryption Method

The MIFARE® 8K DESFire EV1 contains one primary Card Master Key that modifies various card features. Each application within the MIFARE® DESFire EV1 also contains one Application Master Key and up to 13 additional secret keys for use by that application's files.



MIFARE® Desfire EV1 Part#: CLXRN032KNSED Reader: OFF Project File:

MID: Chip Serial Number: ATR:

- Master
  - 000001
    - 00
    - 01
    - 02
    - 03
    - 04

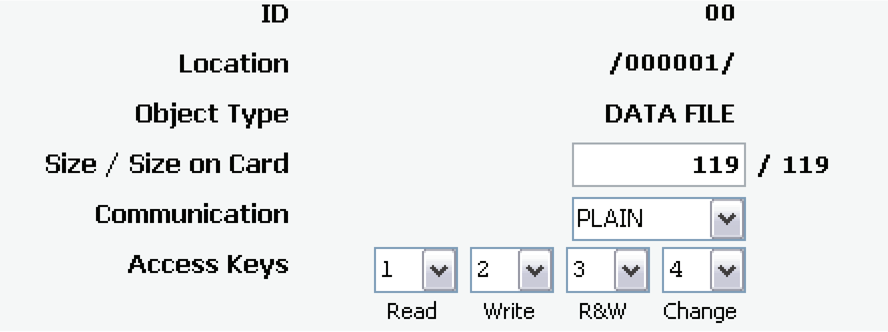
ID	000001
Location	/Master/
Object Type	Application
Size	0
Access Keys	Edit Access Keys

Card Memory Available

1K 3K 5K 7K

Action Log ☑ Show APDU

In order to change an Application's secret keys, left click the Application ID in the left pane window. In the middle pane, click the **`Edit Keys`** button. A window, like the one shown above, will pop up allowing you to edit that application's key values. If you do not have values in mind for the secret keys you can click the **`Randomize Keys`** button to quickly generate random values for all the keys.



ID	00			
Location	/000001/			
Object Type	DATA FILE			
Size / Size on Card	119 / 119			
Communication	PLAIN			
Access Keys	1	2	3	4
	Read	Write	R&W	Change

Key Value	Definition
0	Application Master Key
1-13	User Keys
14	Open / Free Access
15	Deny Access

Files on the EV1 card have an Access Keys property that specifies four key numbers required to Read, Write, Read&Write and for Changing the Access Keys property itself. These keys can be specified by choosing them on the File Properties window. Key 14 can be used to allow Open Access. Additionally, Key 15 is a special key that is used to Deny access. For example, if you wanted a read only file, you could choose key #0 (The Application's Master Key) for Read Access, and key 15 for Write and key 15 for R&W access.

Finalizing Your Card

After you have set your card configuration values, write your configuration to a MIFARE® DESFire EV1 card using the File menu by selecting File->Save to Card. Using your application program, thoroughly test all card functions.

Once you are satisfied that your card structure is working correctly with your application the next step is to send your card project file (CPF) to CardLogix to have 'check cards' made. Save your configuration settings to your project file using the file menu File->Save Project File or File->Save Project File As…

Send your project file to CardLogix. Using your project file, check cards will be made and returned to you, along with a sign-off sheet. After you sign off on the check card we can advance your design to full production. You should also contact your CardLogix sales person for information on the personalization features of your card.

If your cards contain information that needs to be secure, such information may need to be transferred using an encryption method, PGP keys or other secure means of data transfer. Please contact your CardLogix representative for further information.

This page left intentionally blank.

Appendix A: ISO 7816 Error Codes

Each operation returns a status code on completion. The possible codes are as follows:

ISO Error Codes

9000	Operation completed successfully
6E00	CLA not supported
6D00	INS not supported
6300	Warning, value read is different from value written (Write Binary)
6500	Error, unable to erase (no "carry" bits available, Erase Binary and Restore)
6B00	Error, invalid key value (Internal Authentication)
6700	Error, wrong length (Internal Authentication)

Card Type 1 Error Codes

9000	Operation completed successfully
6200	Warning, location(s) protected
6500	Error, Data not written correctly, or no read issued after power up
6687	Error, No More Retries, CLXSB008KB4 inaccessible (Verify)
6688	Error, Invalid PSC presented (Verify)
6700	Error, wrong length (Internal Authentication)
6D00	INS not supported
6E00	CLA not supported

Card Type 2 Error Codes

9000	Operation completed successfully
6300	Warning, data written does not match data read back (Write Binary)
6687	Error, No More Retries, CLXSB008KB4 inaccessible (Verify)
6688	Error, Invalid PSC presented (Verify)
6700	Error, wrong length (Internal Authentication)
6B00	Invalid Parameter (Erase Binary, odd address specified)
6D00	INS not supported
6E00	CLA not supported

Card Type 3 Error Codes

9000	Operation completed successfully
6500	Error, Data not written correctly, may be protected memory
6687	Error, No More Retries, CLXSB008KB4 inaccessible (Verify)
6688	Error, Invalid PSC presented (Verify)
6700	Error, wrong length (Internal Authentication)
6B00	Invalid Parameter (Erase Binary, odd address specified)
6D00	INS not supported
6E00	CLA not supported

Card Type 4 Error Codes

9000	Operation completed successfully
6200	Warning, Card not responding, data may be written
6300	Warning, Card not responding, data may be written
6500	Error, Data not written correctly
6700	Error, wrong length (Internal Authentication)
6D00	INS not supported
6E00	CLA not supported

Card Type 5 Error Codes

9000	Operation completed successfully
6200	Warning, Card not responding, data may be written
6300	Warning, Card not responding, data may be written
6500	Error, Data not written correctly
6700	Error, wrong length (Internal Authentication)
6D00	INS not supported
6E00	CLA not supported

Card Type 6 Error Codes

9000	Operation completed successfully
6200	Warning, Card not responding, data may be written
6300	Warning, Card not responding, data may be written
6500	Error, Data not written correctly
6700	Error, wrong length (Internal Authentication)
6D00	INS not supported
6E00	CLA not supported
6B00	Validation Error

Appendix B: Memory Address Range

The following table lists the address ranges for the various card types. This table should be used as a guide using the CLX_ReadCard and CLX_WriteCard commands. When using these commands, the offset and size parameters should be kept within a valid memory range. When the Page Mode is enabled (CLX_SetPageMode command), the address of a CLX_WriteCard command must be on page boundaries.

Table C-1: Card Address Ranges

Card Type	Part Number	Memory Range	Memory Size	Page Size
1	CLXSB000KB1**	N/A	N/A	N/A
2	CLXSB000KB2*	N/A	N/A	N/A
3	CLXSB000KB6*	N/A	N/A	N/A
4	CLXSA001KA1	0 - 127	128	8
5	CLXSA001KK1	0 - 126	127	N/A
6	CLXSA002KA2	0 - 255	256	8
7	CLXSA002KE1	0 - 255	256	8
10	CLXSA004KA6	0 - 511	512	16
11	CLXSA004KD2	0 - 511	512	N/A
12	CLXSA004KE2	0 - 511	512	16
13	CLXSA008KA7	0 - 991	992	16
14	CLXSA008KB3	0 - 127	128	N/A
16	CLXSA008KE3	0 - 1023	1024	16
17	CLXSA016KA8	0 - 2047	2048	16
20	CLXSA064KA3	0 - 8191	8192	32
21	CLXSA064KE4	0 - 8191	8192	32
22	CLXSA064KE6	0 - 8191	8192	32
23	CLXSA256KA5	0 - 32767	32768	64
24	CLXSA001MD1	0 - 1048575	1048576	64
25	CLXSA001KK1	0 - 126	127	8
26	CLXSA128KA4	0 - 127	128	32

**Card type is not currently supported. **Card type 2 (CLXSA000KB1), uses B1 API calls only.*

This page left intentionally blank.

Appendix C: Reader Command Matrix

The following tables can be used to cross-reference which commands work with which type of reader. The numbers listed at the top of each table represent a specific reader type.

	Reader Type				
General API Commands	**1**	**9**	**10**	**13**	**14**
CLX_OpenReader	X	X	X	X	X
CLX_CloseReader	X	X	X	X	X
CLX_CloseAll	X	X	X	X	X
CLX_APIVersion	X	X	X	X	X

	Reader Type				
General Reader Commands	**1**	**9**	**10**	**13**	**14**
CLX_GetError	X	X	X	X	X
CLX_CardInserted	X	X	X	X	X
CLX_ResetReader	X	X	X	X	X
CLX_SetReaderLed	X	X			
CLX_GetReaderVersion	X	X	X	X	X
CLX_GetReaderStatus	X	X		X	
CLX_LatchReader				X	
CLX_UnLatchReader				X	

	Reader Type				
Magnetic Stripe Card Commands	**1**	**9**	**10**	**13**	**14**
CLX_StartMagCard	X	X	X	X	
CLX_GetMagData	X	X	X	X	
CLX_StopMagCard	X	X	X	X	

	Reader Type				
Encryption & Hashing Commands	**1**	**9**	**10**	**13**	**14**
CLX_3DESEncrypt	X	X	X	X	X
CLX_3DESDecrypt	X	X	X	X	X
CLX_AESDecrypt	X	X	X	X	X

Encryption & Hashing Commands	Reader Type 1	9	10	13	14
CLX_AESEncrypt	X	X	X	X	X
CLX_DESEncrypt	X	X	X	X	X
CLX_DESDecrypt	X	X	X	X	X
CLX_Sha	X	X	X	X	X
CLX_Sha1	X	X	X	X	X
CLX_ShaInternal	X	X	X	X	X

Memory Card Commands	Reader Type 1	9	10	13	14
CLX_ChangePassword	X	X		X	X
CLX_EraseBinary	X	X		X	X
CLX_ReadCard	X	X		X	X
CLX_ReadCardtoFile	X	X		X	X
CLX_SetCardPageMode	X	X			
CLX_SetCardType	X	X		X	X
CLX_VerifyPin	X	X		X	X
CLX_WriteCard	X	X		X	X
CLX_WriteCardFromFile	X	X		X	X

Microprocessor Card Commands	Reader Type 1	9	10	13	14
CLX_C0ExternalAuth	X	X	X	X	X
CLX_C0GetResponse	X	X	X	X	X
CLX_C0InternalAuth	X	X	X	X	X
CLX_CardOn	X	X	X	X	X
CLX_CardOff	X	X	X	X	X
CLX_ExternalAuth7816	X	X	X	X	X
CLX_ExternalAuth7816Ex	X	X	X	X	X
CLX_GenerateKey	X	X	X	X	X
CLX_GetChallenge7816	X	X	X	X	X
CLX_GetResponse7816	X	X	X	X	X
CLX_InternalAuth7816	X	X	X	X	X
CLX_Invalidate7816	X	X	X	X	X
CLX_PurseDep	X	X	X	X	X
CLX_PurseDep_Secure	X	X	X	X	X

	Reader Type				
Microprocessor Card Commands	**1**	**9**	**10**	**13**	**14**
CLX_PurseInit	X	X	X	X	X
CLX_PurseVal	X	X	X	X	X
CLX_PurseVal_Secure	X	X	X	X	X
CLX_PurseWdl	X	X	X	X	X
CLX_PurseWdl_Secure	X	X	X	X	X
CLX_Read7816	X	X	X	X	X
CLX_ReadBinary7816	X	X	X	X	X
CLX_ReadSecure	X	X	X	X	X
CLX_Rehab7816	X	X	X	X	X
CLX_Select	X	X	X	X	X
CLX_Select7816	X	X	X	X	X
CLX_SetEncryption	X	X	X	X	X
CLX_UpdateBinary7816	X	X	X	X	X
CLX_UpdateSecure	X	X	X	X	X
CLX_Verify7816	X	X	X	X	X
CLX_Verify7816Ex	X	X	X	X	X
CLX_Write7816	X	X	X	X	X
CLX_WriteBinary7816	X	X	X	X	X
CLX_WriteSecure	X	X	X	X	X

	Reader Type				
B5 Card Commands	**1**	**9**	**10**	**13**	**14**
CLX_ReadB5Card	X	X		X	X
CLX_ReadB5ProtectionBits	X	X		X	X
CLX_VerifyB5Password	X	X		X	X
CLX_WriteB5Card	X	X		X	X
CLX_WriteB5ProtectionBits	X	X		X	X

	Reader Type				
K1 Card Commands	**1**	**9**	**10**	**13**	**14**
CLX_EraseK1AZ	X	X		X	
CLX_GetK1EC	X	X		X	
CLX_VerifyK1Password	X	X		X	

K2 Card Commands	**Reader Type** **1**	**9**	**10**	**13**	**14**
CLX_EraseK2AZ1	X	X			
CLX_EraseK2AZ2	X	X			
CLX_GetK2EC	X	X			
CLX_VerifyK2Password	X	X			

Appendix D: Glossary

AC: Alternating Current

ACK: Acknowledge. Used in a communications protocol to Acknowledge correct receipt of a message.

Answer to Reset: The response an ICC Card returns when the proper power sequence is applied. Defined in ISO 7816-3 for Microprocessor Cards. Definition for Synchronous Cards is not as well defined.

APDU: An Application Protocol Data Unit contains either a command message or a response message, sent from the interface device to the card; or conversely, from the card to the interface device.

ASCII: A character set used by many computers.

Asynchronous Cards: Also known as Microprocessor Cards. ICC Cards which have a microprocessor and function according to ISO 7816-3 specifications for Microprocessor Cards. Asynchronous refers to the fact that they communicate using an asynchronous communications technique.

ATR: Answer to Reset. Elementary File indicating operating characteristics of card.

Authentication: The process of assuring that one, or both, parties to a transaction are who they say they are.

BCC: Block Check Character. Used in many communications protocols to detect errors in transmission.

BPS: Bits Per Second, abbreviated in either upper case or lower case. Refers to the number of bits which can be sent on a communications path in one second.

Card Seated: Refers to a card which is actually inserted fully into a card Connector such that the switch at the back of the reader slot changes state because of the contact with the card.

Card Connector: Any connector designed to receive an ICC.

Challenge: Some of the security schemes used with Smart Cards require a random number to be associated with key manipulations. The random number, which of course changes with every transaction/session, assures that no two occurrences of the same transaction will look the same, thus avoiding replay of secure transactions.

CLA: Class Byte. This is one of the bytes used in a APDU.

Communications Protocol: A set of rules governing the structure, sequencing, and validation of messages between two or more points on a communications media.

CTS: A hardware signal from the host to the Axiohm Model 152 reader which allows the host to block transmission of data from the Model 152.

DB-9: This is the kind of connector used to connect to the host. If you were running a card reader using a PC as a host, this connector would mate to a “9 pin com port”.

DC: Direct Current.

DF: Dedicated File. A file containing file control information and optionally, memory available for allocation. It may be the parent of EFs and/or DFs.

DLL: Dynamic Link Library.

EEPROM: Electrically Erasable Programmable Read Only Memory. Most Smart Cards store user data in EEPROM, which can be erased and re-programmed numerous times. See the card manufacturer’s specifications for information on the number of programming cycles available with a particular card.

EF: Elementary File. A set of data units or records which share the same file identifier, such as an ATR. An Elementary File cannot be the parent of another file.

EOT: End of Transmission. This byte is used in many communications protocols to signify the end of a transmission. In the ASCII character set it is defined to have the value 04H.

Erase: When talking about smart cards, erase usually means setting data bits to all ones. This is because EEPROM programming changes bits from the erased (all ones) state to zeroes a bit at a time, but cannot change single bits from zero to one. Currently available EEPROMs require at least one complete byte to change to ones (erasure) in order to change a single bit to one. Some EEPROMs erase in blocks of 2 or more bytes.

ETX: End of Text. This byte is used in many communications protocols to signify the end of a transmission. In the ASCII character set it is defined to have the value of 03H.

F/D Ratio: F stands for Frequency, D stands for Divisor. In ISO 7816-3, these terms are used as a ratio (along with an oscillator frequency) to determine the actual speed of the smart card interface. ISO 7816-3 defines a default F of 372 with a default D of 1. When used with a standard oscillator frequency a speed of 9600 bps is the result.

Field: An area in the CardAppz® database file that tracks just one type of item, i.e. a city, state, ZIP code, and so on. See also database and record.

File: A document saved to a computer disk.

Fuse: Many Memory Cards have a way of making an irreversible change to the card structure so that future modifications to certain portions of the card (i.e. card serial number) are impossible. Usually the technique is referred to as a fuse.

GeldKarte: A variant of the ISO 7816-3, Amendment 1, T=1 protocol.

GSM: Global System Mobile; A cell phone card standard.

HEX: Hexadecimal, base 16. Some numbers in this manual are followed by H (e.g. 03H). This notation is to denote a HEX number.

Host: The device connected to the reader via the communications cable. The host controls all operations of the reader.

ICC: Integrated Circuit Card. Any card which acts as a carrier for an Integrated Circuit. Most particularly, cards which conform to ISO 7816 standards.

INS: Instruction. This is one of the bytes used in a APDU.

ISO 7816: This international standard is used as a guideline by many smart card manufacturers. It defines standards (mechanical, electrical, operational) for a integrated circuit cards with contacts. Other standards apply to ICCs without contacts.

Key: Many smart card security schemes require the use of a "key" to prove either a reader has legitimate access to a smart card, or that a reader should accept a smart card as valid. Keys are passwords via the application.

Lc: One of the parameters which may be used in a APDU, it codes the length of data being transferred to the smart card.

Le: One of the parameters which may be used in a APDU, it codes the length of data expected to be returned by the smart card.

LED: Light Emitting Diode, the visible lights on the front of the reader.

LSAM: Local Secure Application Module. This is a Secure Application Module inserted in the LSAM Connector located inside the reader.

LSAM Connector: This Connector (if installed) is found inside the reader where it is not easily accessible to the general public. It receives the LSAM which allows the reader to be used for security applications. One disadvantage of having a SAM in an LSAM Connector is the possibility of the whole reader being stolen. Such an occurrence could (if the logical security scheme is inadequate), expose the owner to possible fraud. If the SAM is used to contain actual cash from transactions, the loss of the entire reader could mean loss of the revenue currently residing in the LSAM.

mAL: Milliampere.

Memory Cards: Also known as Synchronous Cards or Serial Cards. These cards do not have a microprocessor. They contain simple (relatively), circuitry which allows the card to read, write and update data. There are a variety of security mechanisms available on many cards.

Microprocessor Cards: Also known as Asynchronous Cards. ICC Cards which have a microprocessor and function according to ISO 7816-3 specifications for Microprocessor Cards.

MF: Master File. The mandatory, unique Dedicated File representing the root of the file structure.

ms: milliseconds

Multi-drop: Refers to techniques for multiple computers/devices to be attached to a single communications line and be able to communicate with each other coherently.

N/A: Not Applicable.

NAK: Negative Acknowledge. This term is used, usually when talking about a communications protocol, to designate how one party on a communications line tells another party that a particular message was not received correctly. Typically, when a sender receives a NAK, the sender retransmits the correct message.

OSB: Operation Status Byte. This byte is present in all responses from the reader when using either of the TLP-224 protocols available. It informs the host of the final status of the operation.

P1: One of the parameters used in a APDU. Specific usage depends on the APDU being used.

P2: One of the parameters used in a APDU. Specific usage depends on the APDU being used.

P3: One of the parameters used in a APDU. It is used to code either Lc or Le. If both Lc and Le are zero, P3 is zero.

PIN: Personal Identification Number. Many smart cards have security mechanisms with presentation of a PIN to authorize usage of the card.

Procedure Byte: This byte is part of the card level communications for microprocessor cards using the T=0 protocol. It used to regulate the flow of data from the card to the reader. The Axiohm Model 152 handles all procedure bytes, relieving the host of having to even know that procedure bytes exist.

Record: A collection of fields in a database that defines one entity. (Each card contains 1 record with the exception of the M.O.S.T.® Card family). See also Database and Field.

RFU: Reserved for Future Use. When sending a command to the reader, any field documented as RFU should be filled with zeroes.

RS232: An electrical specification of a communications system which is used between parties on a communications line. The Axiohm model 152 is RS232-compatible.

SAM: Secure Application Module. Many Smart Card applications require security to protect against fraud. Many security schemes are implemented via SAMs which are smart cards which make security algorithms available and supply a secure place to store keys. It is quite difficult to discover the value of a key stored in a SAM.

SAM Box: This accessory, available for connection to many readers allows the reader access for up to seven additional SAMs. The SAM box may be stored under the counter or in some other secure location.

Secure Application Module: See SAM.

Serial Cards: See Memory Cards.

Smart Card: Any card with implanted, programmable, integrated circuitry.

SOH: Start of Header. This byte is used in many communications protocols to signify the start of a transmission. In the ASCII character set it is defined to have the value 01H.

SW1 and SW2: These bytes are defined in ISO 7816 to be the last two bytes of any APDU response. They convey status information about the card operation.

Synchronous Cards: See Memory Cards.

T=0: One of the protocols defined in ISO 7816 for communicating with microprocessor cards. This protocol is byte-oriented, with error correction and recovery techniques applied on a byte-by-byte basis.

T=1: One of the protocols defined in ISO 7816 for communicating with microprocessor cards. This protocol is block oriented, with error correction and recovery techniques applied to whole messages.

TLP-224: One of the communication protocols supported by the Axiohm Model 152 and 171 readers for communication between the reader and the host. This protocol is used by several manufacturers of smart card readers. If you use only the core set of commands in your application, your unit may be plug compatible with units from other manufacturers.

TLP-224 Turbo: A proprietary variant of the TLP-224 communications protocol which reduces the transmission time required to exchange messages between the host and reader.

Transport Code: A passwording technique used by many manufacturers of smart cards to assure that cards cannot be tampered with, or diverted to other destinations, for fraudulent purposes. There are many names used by manufacturers of this technique, but they are usually similar in nature and the level of protection provided.

Update: This function is used the second time you write to an Elementary File (EF). Most Smart Cards use EEPROM for data storage. If the new value to be stored at a location has any one bit where the old value had zero bits, the byte (or maybe a larger section of storage space) must be erased prior to writing the new value. In many cards the operation called "update" performs an erase before writing the new value to the card. Consult your card documentation to determine the exact nature of the "update" operation for your card.

User Connector: The card connector visible to the user. When the Axiohm Model 152 goes through a power cycle, this Connector is selected.

USI2: One of the communications protocols supported by the Axiohm Model 152 & 171 for communications between the reader and the host. This protocol is unique to American Magnetics. This protocol allows fuller usage of the features of the Model 152, 154, 171 units.

Write: This is the function to use when you write to an Elementary File (EF) for the first time. (All subsequent writes are done via the UPDATE function). Most smart cards use EEPROM for data storage, if the new value to be stored at a location has any one bits where the old value had zero bits, the byte (or maybe a larger section of storage space) must be erased prior to writing the new value. If the byte is not erased, only bits which change from a one to a zero will be changed. In many cards the operation called "write" only changes one bits to zeroes. Consult your card documentation to determine the exact nature of the "write" operation for your card.

Notes

Notes

Notes

Quality

CardLogix Corporation is absolutely committed to providing defect free products and services to our customers in partnership with equally committed integration partners and authorized resellers.

RoHS Compliant LEAD FREE 2002/95/EC

ISO 9001:2000 CERTIFIED A14381

- California C Corporation
- CA Resale# SREAA 97-124323
- D&B# 867418899
- SIC Codes# 3577, 3089, 5162
- UNSPCSC Code# 32101617
- Harmonized Code# 8542.10.0000
- NAICS Codes# 334119, 326199, 334418, 334519, 42261, 51421
- CAGE Code# 1KV39
- Congressional District# 47

16 Hughes, Suite 100 · Irvine, CA 92618 · United States
Phone +1 949 380-1312 · Fax +1 949 380-1428
www.cardlogix.com · sales@cardlogix.com

7000011A