IT WORKS
Everything Works

RFID-M1

Demonstration Program

User Guide

Revision 2.10

** Subjects changed without notice

Information furnished by IT WORKS, Ltd. is believed to be accurate and reliable. However, no responsibility is assumed by IT WORKS, Ltd. for its use; nor for any infringement of patents or other rights of third parties which may result from its use.

http://www.itworks.co.th email: biosupport@itworksolutions.com

Table of Contents

1 Introduction

The Host Demo Program is prepared to show the basic features of the RFID-M1 Evaluation Board. The program is designed for Microsoft Window environment and supporting the WIN95, WIN98, WINme and WindowsNT.

2 Function Description

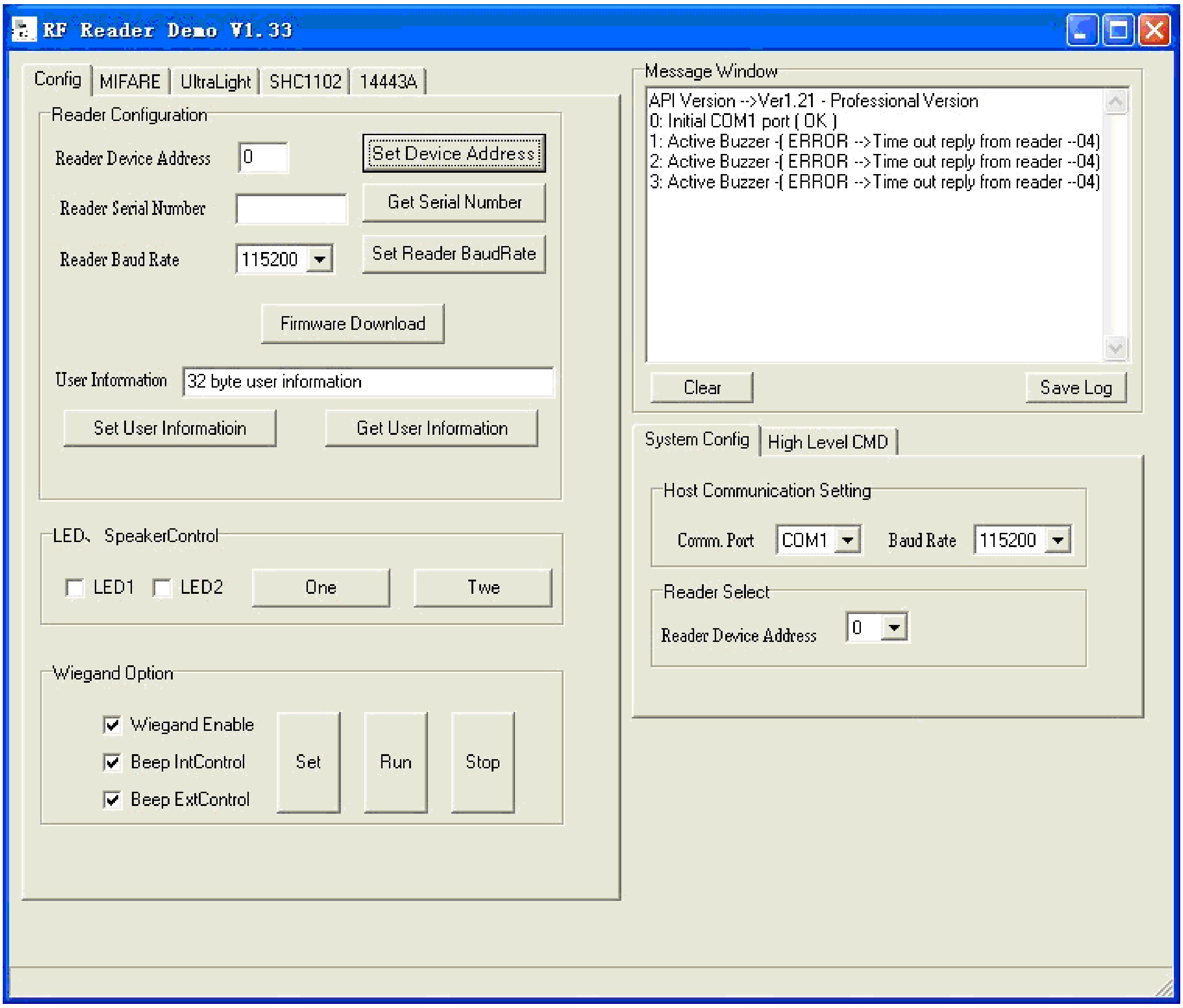
2.1 Program Configuration file

TheRC500.INI initialization file, which is pre-formatted by the factory, is used to set up the configuration of the demonstration program. This INI file must be copied to the WINDOW directory.

2.2 Layout of the Main Screens

There are three main areas with the Application Screen

2.2.1 Main Screen Layout



RF Reader Demo V1.33

Reader Device Address: 0

Reader Serial Number:

Reader Baud Rate: 115200

User Information: 32 byte user information

☐ LED1 ☐ LED2

☑ Wiegand Enable
☑ Beep IntControl
☑ Beep ExtControl

Message Window

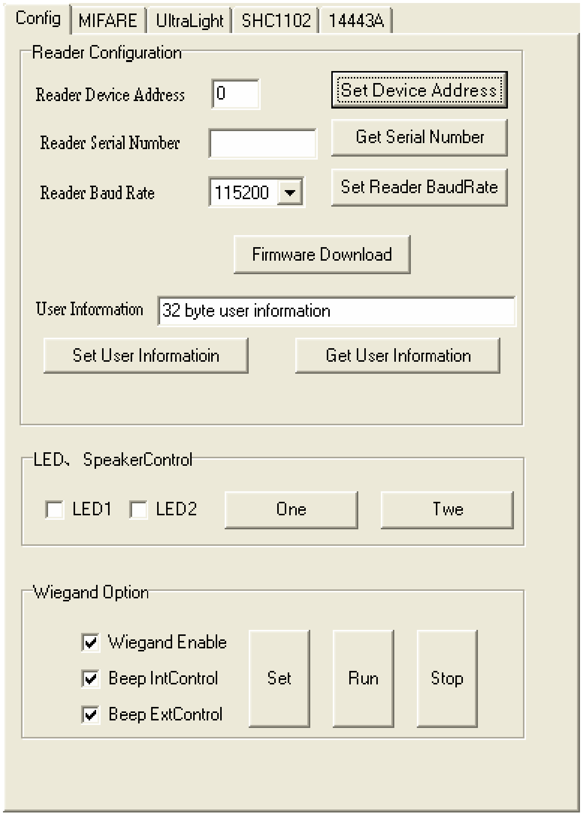
```
API Version -->Ver1.21 - Professional Version
0: Initial COM1 port ( OK )
1: Active Buzzer -[ ERROR -->Time out reply from reader --04]
2: Active Buzzer -[ ERROR -->Time out reply from reader --04]
3: Active Buzzer -[ ERROR -->Time out reply from reader --04]
```

Comm. Port: COM1
Baud Rate: 115200

Reader Device Address: 0

2.2.2 Main Function Group Layout

The left side is the Main Function Group.



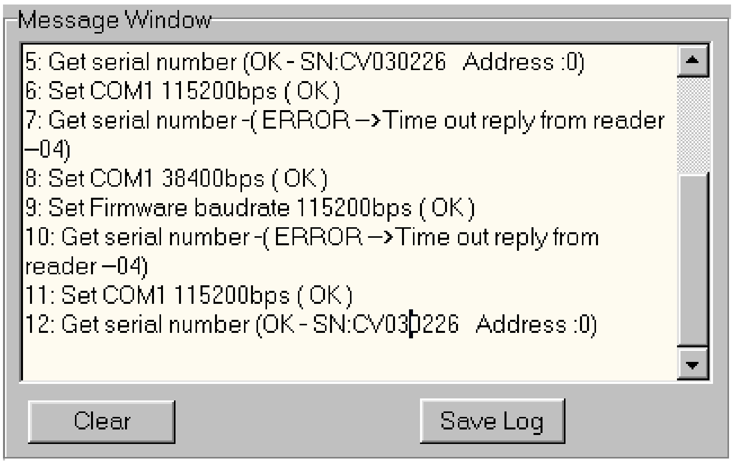
Reader Device Address: 0
Reader Serial Number:
Reader Baud Rate: 115200
User Information: 32 byte user information

☐ LED1 ☐ LED2

☑ Wiegand Enable
☑ Beep IntControl
☑ Beep ExtControl

2.2.3 Message Window

The upper right side is the message window shows the status and results of each executed command.



```
5: Get serial number (OK - SN:CV030226   Address :0)
6: Set COM1 115200bps ( OK )
7: Get serial number -( ERROR -->Time out reply from reader
--04)
8: Set COM1 38400bps ( OK )
9: Set Firmware baudrate 115200bps ( OK )
10: Get serial number -( ERROR -->Time out reply from
reader --04)
11: Set COM1 115200bps ( OK )
12: Get serial number (OK - SN:CV030226   Address :0)
```

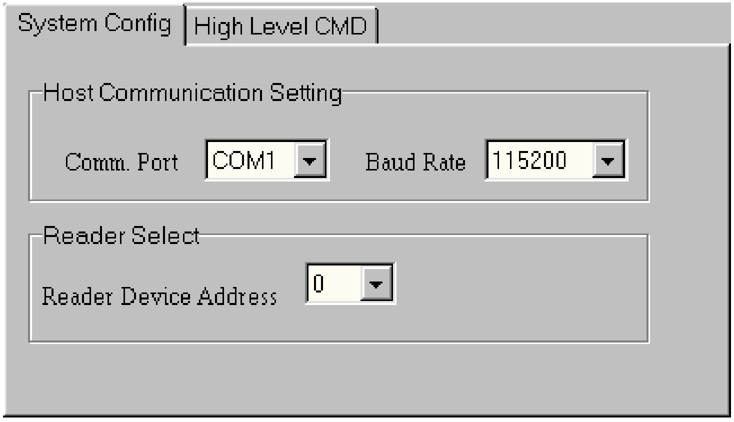
- **Clear**

  Clear the Message Window

- **Save Log**

  The messages in the Message Window will be saved as log file. The log file is a text file with the default name “log.txt”

2.2.4 Secondary Function Group



Host Communication Setting

Comm. Port: COM1

Baud Rate: 115200

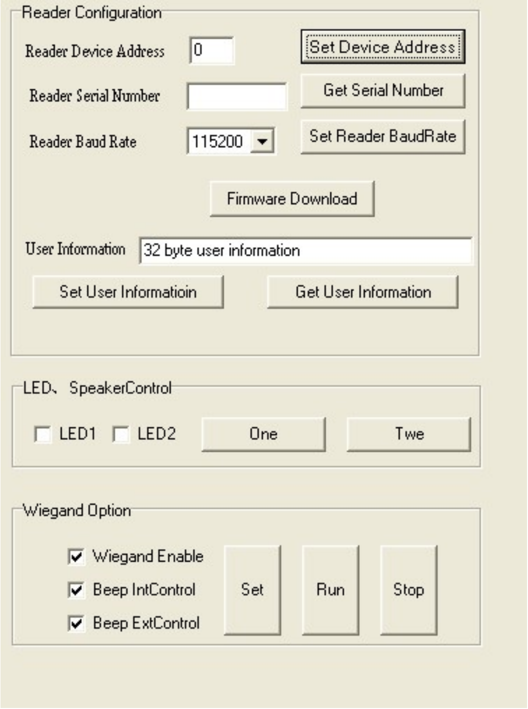
Reader Select

Reader Device Address: 0

2.3 Main Function Groups

2.3.1 Configuration

This function group is related to the reader configuration



Reader Configuration

Reader Device Address: 0

Reader Serial Number:

Reader Baud Rate: 115200

User Information: 32 byte user information

☑ Wiegand Enable

☑ Beep IntControl

☑ Beep ExtControl

- **Reader Configuration**

•Set Device Address

Set the Reader Address for multi-drop mode. The Address is stored inside the reader. The address range is from 0 to 31 but address 0 is reserved for broadcast mode. When Address "0" is used, it is supposed that the reader is connected as point-to-point mode. (In most case, the RS232 use the point-to-point mode.)

•GetSerialNum

Retrieve the Reader Address and Serial Number from the Reader. Each Reader will be programmed with a Product Serial Number by the factory.

- **SetBaudRate**

  Set the communication baud rate of the reader. The default baud rate is 115K and could be changed by the "SetBaudRate". The modified baud rate will be stored in side the Reader's EEPROM.

- **Set User Information**

  The user can write maximal 32-byte user information to the reader, which will be stored in the internal memorizer of the reader.

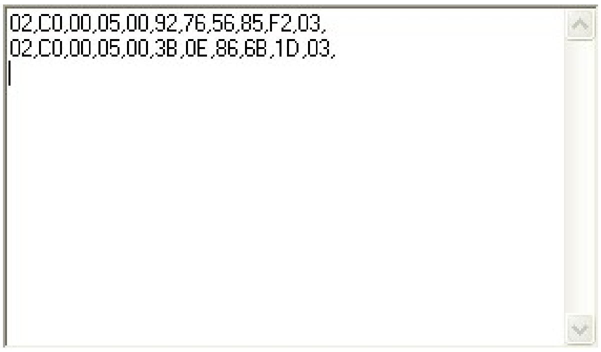
- **Get User Information**

  Get the user information stored in the internal memorizer of the reader

- **Wiegand Option**
  - ✧ **Wiegand Enable** Enable Wiegand mode
  - ✧ **Beep IntControl** Enable buzzer while reading
  - ✧ **Beep ExtControl** Enable external I/O controlled buzzer

SET Set Wiegand setting in the reader, and enter Wiegand mode accordingly

RUN Enter Wiegand mode according to the setting. The reader will detect the cards continuously, and the serial number of the card will be sent via Wiegand port and communication port when a card is detected. The data received by communication port will be shown in the message window as follows.



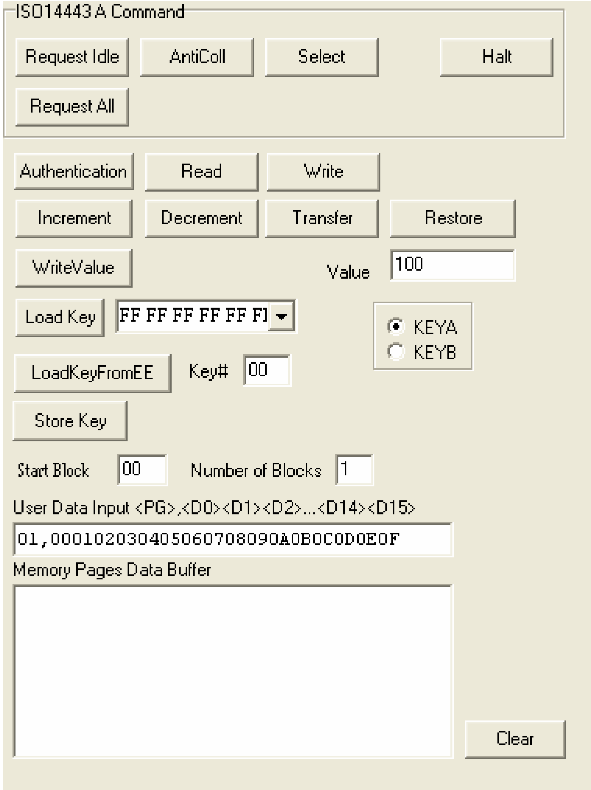
```
02,C0,00,05,00,92,76,56,85,F2,03,
02,C0,00,05,00,3B,0E,86,6B,1D,03,
```

STOP Stop Wiegand mode

Note: The SetBaudRate command only set the communication speed of the reader. The user needs to set the PC's baud rate accordingly.

2.3.2 MIFARE Functions

Provides the MIFARE related operations.



Value: 100

Load Key: FF FF FF FF FF FI

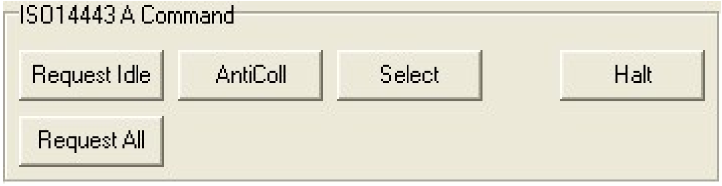
Key#: 00

Start Block: 00

Number of Blocks: 1

```
01,000102030405060708090A0B0C0D0E0F
```

- **ISO 14443 A Commands:**



- **Request Idle:**
  Request/wake up all the cards not in HALT state. The message "Request Idle [OK – 4 " will be shown in the Message Window if succeed.   Otherwise error message will be shown.

- **RequestAll**
  Request all the cards in the field. (Even cards are halted). Equal to the ISO Wake-up all command.

- **Anticoll**
  ISO14443A anti-collision command, a four bytes card UID will be returned. The Anticoll command must be run after a successful REQUEST command.

- **Select**
  ISO14443A SELECT. The card will be selected and open for further command.
  The SELECT command must be run after a successful Anticoll command. The UID returned by the previous “Anticoll” command will be used to select the card.

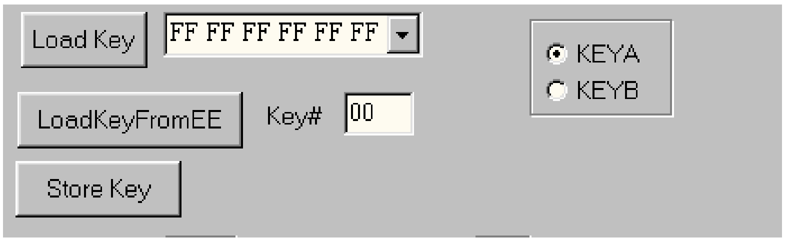
- **Halt**

  ISO14433A HALT. This command will halt the active (selected) card and that card will not respond to any further commands except the REQUESTALL or a power-on reset is applied to the card. (Turn off the reader's RF field or remove the card from the field.) [1]

- **Mifare Authentication Function**
  - **Authentication**

    Execute MIFARE's authentication process.

    The user could select to use either KEYA or KEYB for the authentication by the corresponding radio button.



FF FF FF FF FF FF

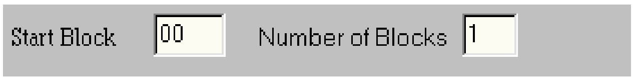
Key# 00

● KEYA
○ KEYB

Note: The correct Key must be pre-loaded into the Master Key Buffer before executes the Authentication and the further Read/Write and value related operations work properly only after the successful Authentication.

- **Mifare Read/write functions**
  - **Read**

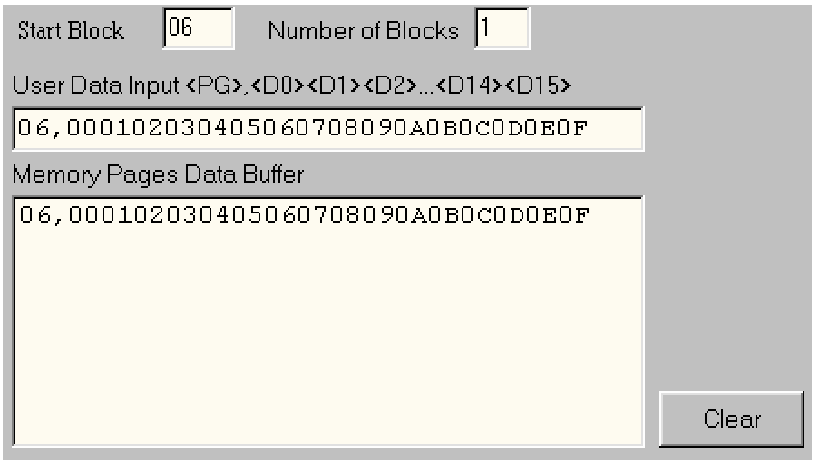
    Read memory pages (blocks) from the selected card. The starting page address (in hexadecimal format) and the number of pages to be read should be enter to the corresponding data enter fields.



Start Block 00 Number of Blocks 1

The contents of the memory blocks (up to four blocks) will be displayed on the Page Memory Data Buffer according to the following format –Block number (1 byte), read data (16 bytes).

For example, read one block from the Block 6



Start Block 06 Number of Blocks 1

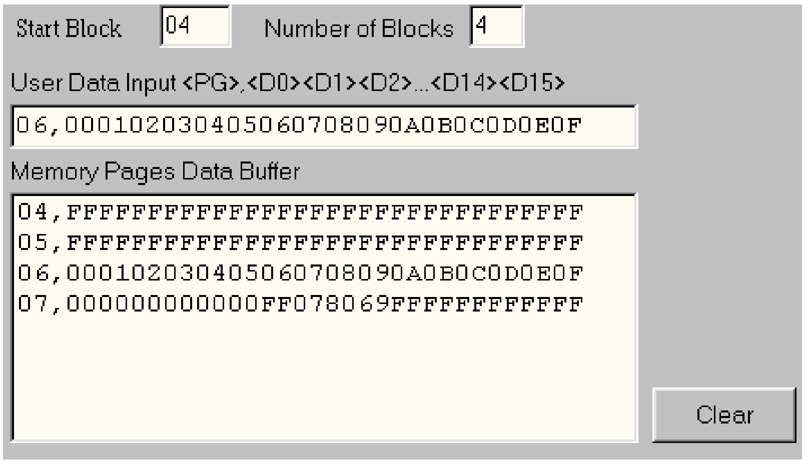
User Data Input <PG>,<D0><D1><D2>...<D14><D15>

```
06,000102030405060708090A0B0C0D0E0F
```

Memory Pages Data Buffer

```
06,000102030405060708090A0B0C0D0E0F
```

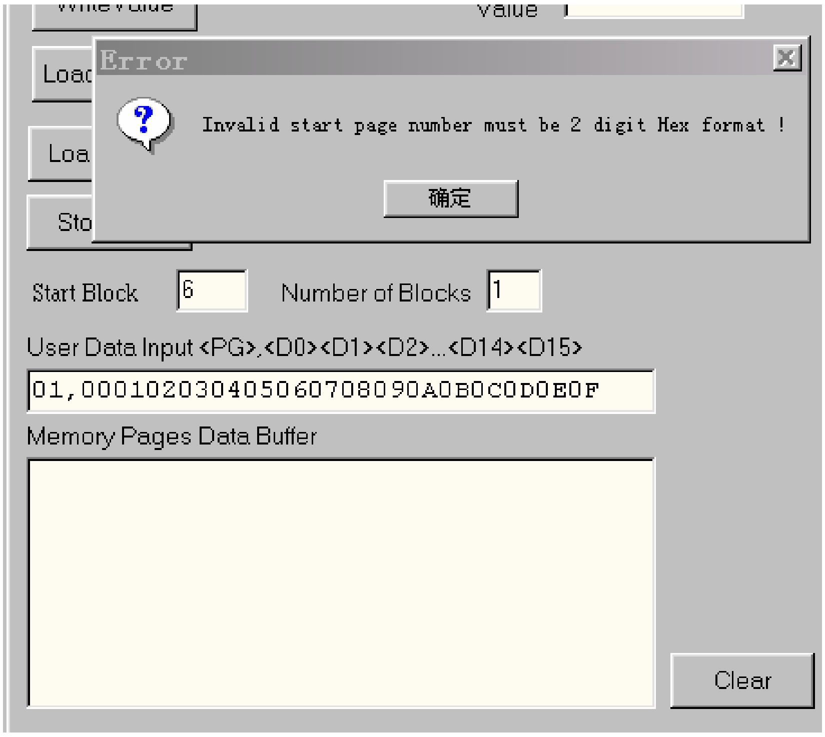
In the Memory Pages Data Buffer, the result is:
"06,000102030405060708090A0B0C0D0E0F”
where “06” is the block number and the following 16 bytes are the data read.
Reading of multiple (up to four) blocks is possible but those blocks must be within a single sector. (cross-sector reading is not allowed.)



Start Block 04
Number of Blocks 4
User Data Input <PG>,<D0><D1><D2>...<D14><D15>
06,000102030405060708090A0B0C0D0E0F
Memory Pages Data Buffer

```
04,FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
05,FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
06,000102030405060708090A0B0C0D0E0F
07,000000000000FF078069FFFFFFFFFFFF
```

NOTE: The Input of the Start Block must be entered in two-digit HEXDECIMAL format with LEADING ZERO. For example Block 6 should be “06” instead of “6” otherwise error message of “Invalid start page number” will be shown.



Error

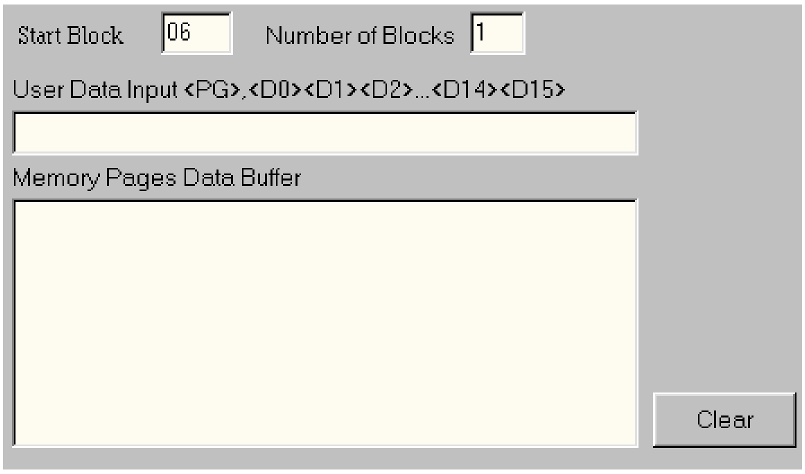
Invalid start page number must be 2 digit Hex format !

Start Block 6
Number of Blocks 1
User Data Input <PG>,<D0><D1><D2>...<D14><D15>
01,000102030405060708090A0B0C0D0E0F
Memory Pages Data Buffer

- **Write**
  The “WRITE” button writes multiple blocks of data from the Memory Data Buffer to the MIFARE card.

- **Procedure of the “Write Operation”**
  For Example: Write one block of data to Block 6 -

- Enter the Start Block Address and Number of Block to be written. (Same as the "Read" Operation.)



Start Block 06 Number of Blocks 1

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

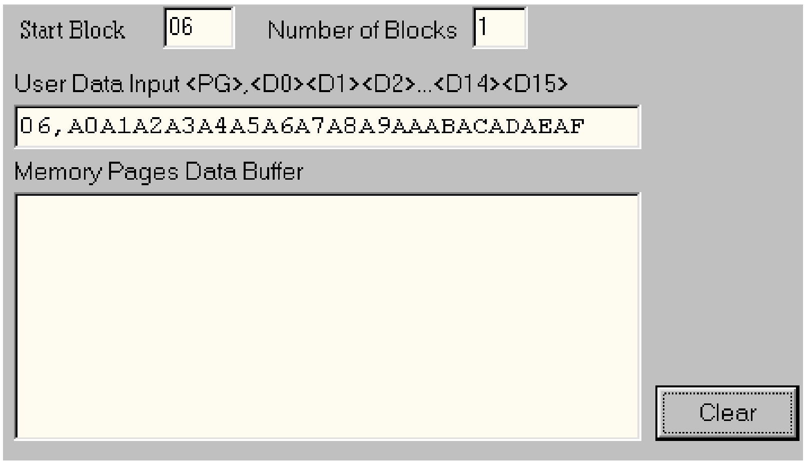
Memory Pages Data Buffer

- Enter the data to be written to the User Data Input Buffer in following the following format:

  Enter Data in Hex format:

  <page>,<D0> <D1><D2><D3> …<D14><D15>

  For example, the string: "06,A0A1A2A3A4A5A6A7A8A9AAABACADAEAF" means 16-bytes of data A0..AF will be written to block 6.



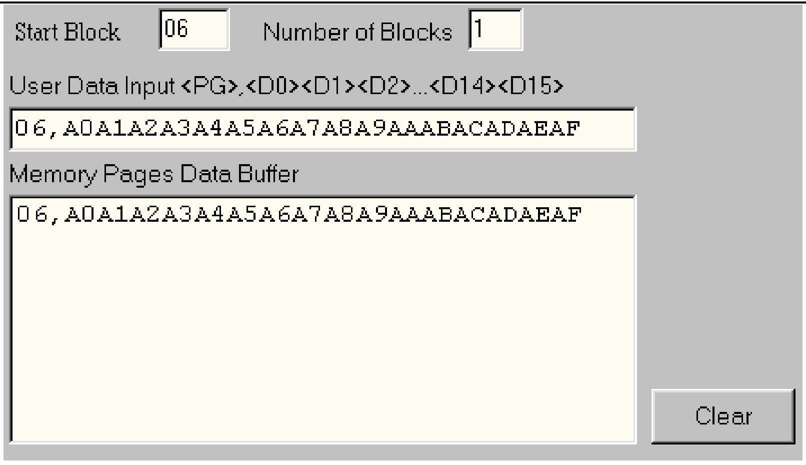
Start Block 06 Number of Blocks 1

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

06,A0A1A2A3A4A5A6A7A8A9AAABACADAEAF

Memory Pages Data Buffer

- Press "RETURN" to copy the input data from the Data Input Buffer to the Memory Pages Data Buffer. The data in the User Data Input Buffer will be transferred to the Memory Pages Data Buffer



Start Block 06 Number of Blocks 1

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

```
06,A0A1A2A3A4A5A6A7A8A9AAABACADAEAF
```

Memory Pages Data Buffer

```
06,A0A1A2A3A4A5A6A7A8A9AAABACADAEAF
```

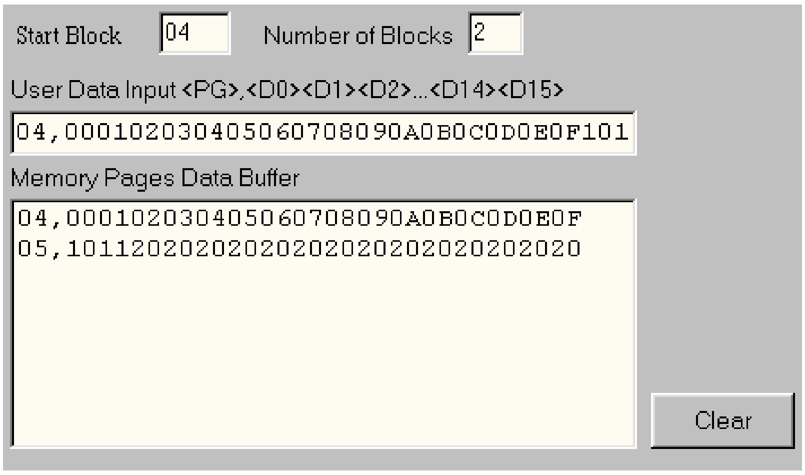
- Click the “Write” button to write the data to the card.

- **Write Multiple Blocks**

  Up to four continue Blocks (within same sector) could be written to the card by a single “Write” commands.

  The user can enter the data of the blocks to be written one by one just like writing the single block. Or the data for the multiple blocks could be type continually in the User Data Buffer. And the datum 0x20 will be patched to fill the last incomplete block.

  For Example, we would like to write data to block 4 and block 5, but there are only 18 bytes of data are inputted to the User Data Input Buffer instead of the exactly 32 bytes for two blocks.



Start Block 04 Number of Blocks 2

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

```
04,000102030405060708090A0B0C0D0E0F101
```

Memory Pages Data Buffer

```
04,000102030405060708090A0B0C0D0E0F
05,10112020202020202020202020202020
```

Then click the “Write” button to write the data to the card.

.

- **Enter Data in text mode**

  Instead of entering the hexadecimal numbers to the Data Input Buffer, the user can also enter ASCII string directly by the following data format.

  <page>”data_string

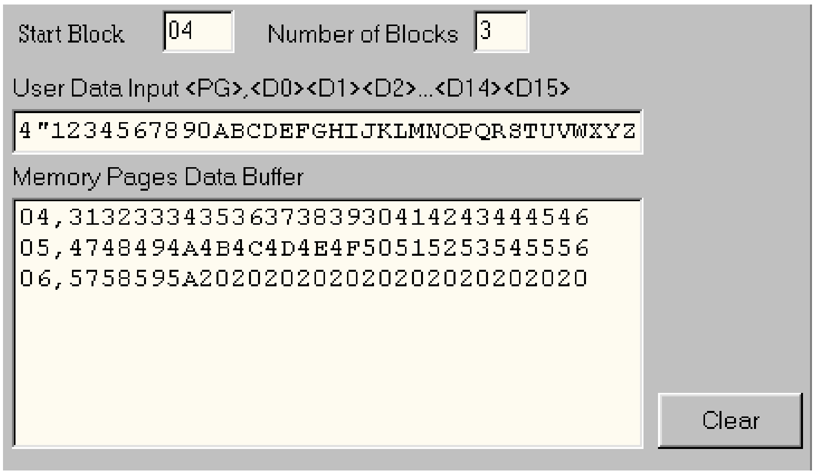
  page – indicate the start block number.

  “ – to prompt the program the following is a string

  data_string – the input data in ASCII string format.

  - The input string could be up to 256 characters.
  - As the same as the hexadecimal mode, characters of 0x20 will be filled to last

block.



Start Block 04 Number of Blocks 3

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

```
4"1234567890ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

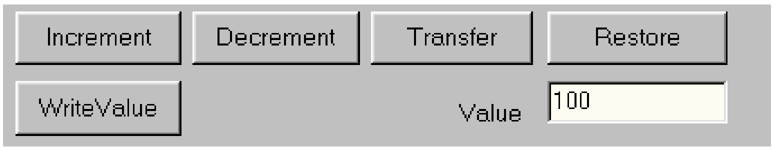
Memory Pages Data Buffer

```
04,31323334353637383930414243444546
05,4748494A4B4C4D4E4F50515253545556
06,5758595A202020202020202020202020
```

- The user could copy the data from the Memory Pages Data Buffer back to the User Data Input
  Buffer by double click the corresponding page in the Memory Pages Data Buffer.

NOTE: When writes with multiple blocks, the user must be careful no to corrupt the sector trailer.

- **Mifare Value Related Functions:**



Value 100

- **Increment and Decrement**
  Increase or decrease with the value amount to/from the MIFARE value block. The block must be pre-formatted as MIFARE's value block format.
  Note: the Increment and Decrement functions only hold the calculated value in the internal register. The actual value is not written back to the value block until the Transfer function is executed. Please refer the MIFARE's data sheet for the detail.

- **Transfer**
  Transfer the calculated (by command increment or decrement) value amount from the internal register to the value block. The Transfer function should be called to write back the value block after the increment/decrement command.

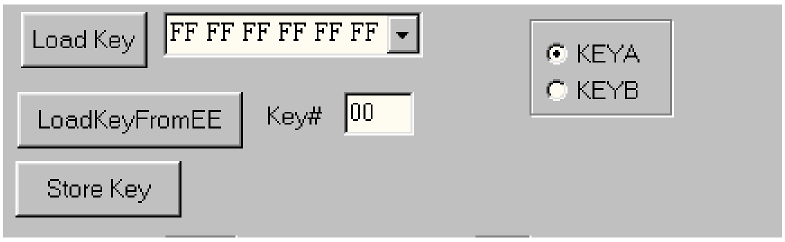
- **Restore**
  Restore the value from the Value Block to the internal register. Then the value could be transferred to another value block by using the Transfer command.

- **WriteValue**
  Format a normal memory block (not in Mifare Value format) to a value block format with the initial value.

- **ReadValue**

Read back the value from the value block

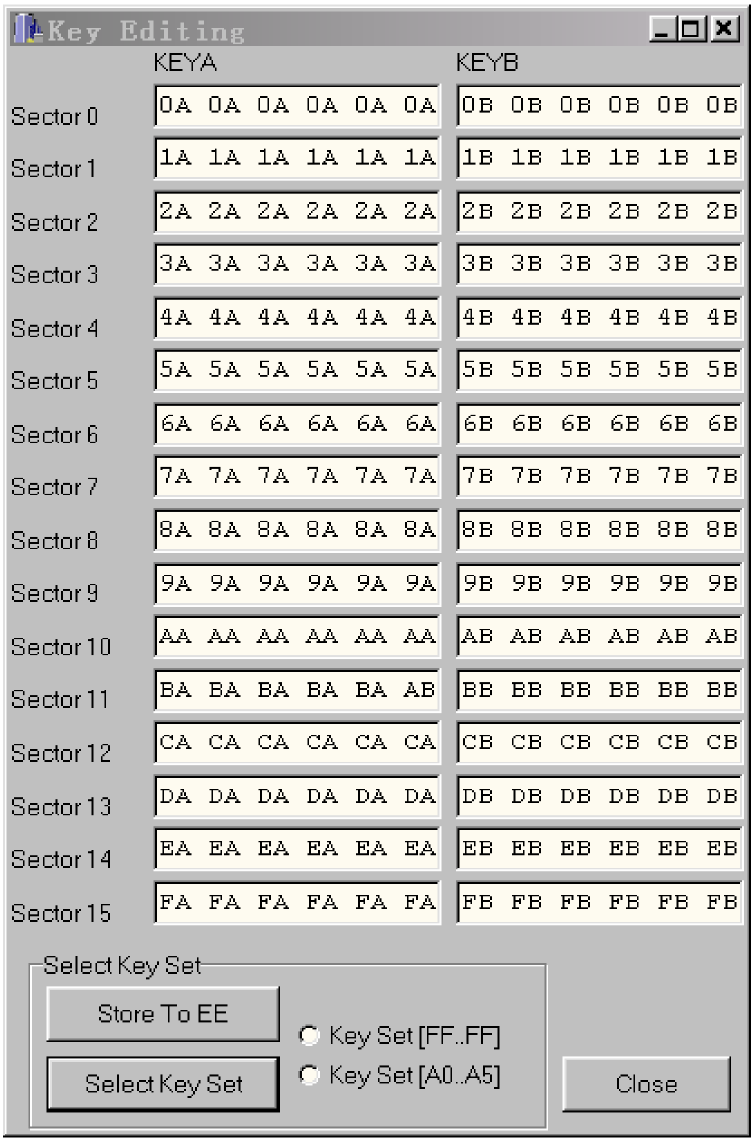
- **Mifare Key Loading functions:**



- **LoadKey**
  Directly load a key string to the MIFARE reader chip’s Master Key Buffer.

- **LoadKeyFromEE**
  Basically has the same function as the LoadKey, but the Key is loaded from the MFRC500 Reader chip’s Internal EEPROM instead of the key string inputted from the EDIT BOX by user. There are sixteen sets of keys (KEYA and KEYB) stored in the reader chip’s internal EEPROM. One of the sixteen keys could be selected by the “Key#” and the “KEYA/KEYB” Radio Button determines to load KeyA or KeyB.

- **Store Key**
  Store the sixteen Keys to the MFRC500 Reader chip’s internal EEPROM. The user could click “Store Key” button to pop up the Key Editing window.



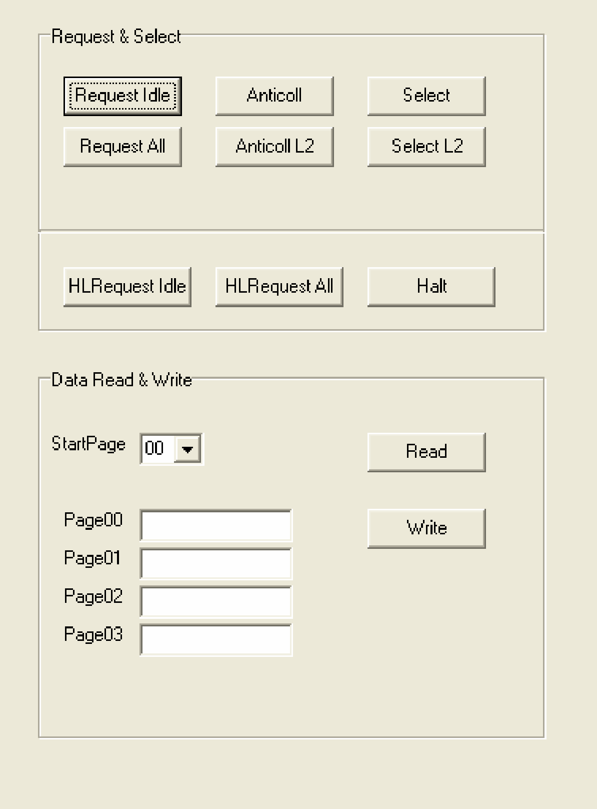
	KEYA	KEYB
Sector 0	0A 0A 0A 0A 0A 0A	0B 0B 0B 0B 0B 0B
Sector 1	1A 1A 1A 1A 1A 1A	1B 1B 1B 1B 1B 1B
Sector 2	2A 2A 2A 2A 2A 2A	2B 2B 2B 2B 2B 2B
Sector 3	3A 3A 3A 3A 3A 3A	3B 3B 3B 3B 3B 3B
Sector 4	4A 4A 4A 4A 4A 4A	4B 4B 4B 4B 4B 4B
Sector 5	5A 5A 5A 5A 5A 5A	5B 5B 5B 5B 5B 5B
Sector 6	6A 6A 6A 6A 6A 6A	6B 6B 6B 6B 6B 6B
Sector 7	7A 7A 7A 7A 7A 7A	7B 7B 7B 7B 7B 7B
Sector 8	8A 8A 8A 8A 8A 8A	8B 8B 8B 8B 8B 8B
Sector 9	9A 9A 9A 9A 9A 9A	9B 9B 9B 9B 9B 9B
Sector 10	AA AA AA AA AA AA	AB AB AB AB AB AB
Sector 11	BA BA BA BA BA AB	BB BB BB BB BB BB
Sector 12	CA CA CA CA CA CA	CB CB CB CB CB CB
Sector 13	DA DA DA DA DA DA	DB DB DB DB DB DB
Sector 14	EA EA EA EA EA EA	EB EB EB EB EB EB
Sector 15	FA FA FA FA FA FA	FB FB FB FB FB FB

- **Select Key Set**
  By active the corresponding Radio Button the user could select one of the two commonly used keys sets. The old MIFARE cards may use the [A0..A5, B0..B5] key set as its default transport key but for the most new MIFARE chips the [FF..FF,FF..FF] key set may be used. The "Select Key Set" button selects the key set according the marked radio button and shows them in the Edit Box. The use r can still change the keys shown in the Edit Boxes.
- **Store To EE**
  Save the 16 Key sets to the MFRC500 chip's internal EEPROM.

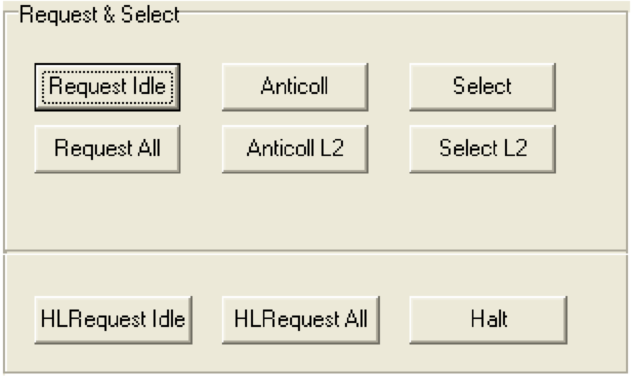
. UltraLight Functions

UltraLight

User interface is shown as followed.



Request&Select operation



Request Idle	ISO14443A	REQA command	
Request All	ISO14443A	WAKE-UP command	
Anticoll	ISO14443A	ANTICOLLISION command	（UID =4byte）
Select	ISO14443A	SELECT command	（UID =4 byte）
Anticoll L2	ISO14443A	ANTICOLLISION command	（UID =7byte）
Select L2	ISO14443A	SELECT command	（UID =7byte）
Halt	ISO14443A	HALT command	

High level command

HLRequest Idle

The steps of this function as followed.

Request Idle ------ Anticoll -------- Select ------ Anticoll L2 ------- Select L2

When this command is finished, UltraLight card will enter ACTIVE status from IDLE status, and return 7

Byte UID.

HLRequest All

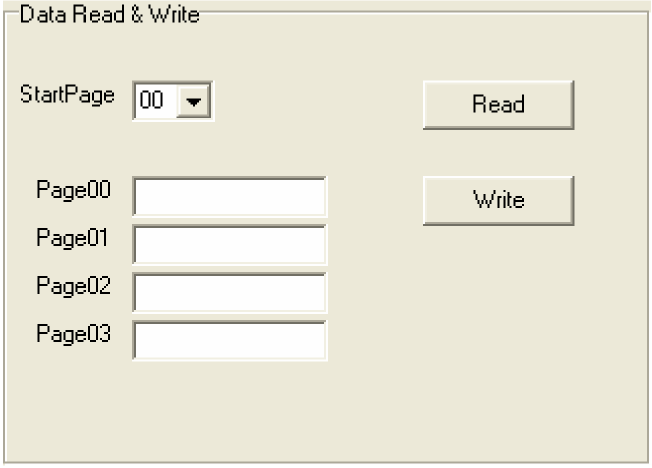
The steps of this function as followed.

Request All ------ Anticoll -------- Select ------ Anticoll L2 ------- Select L2

When this command is finished, UltraLight card will enter ACTIVE status from IDLE status or HALT status, and return 7 Byte UID.

Date Read&Write

User interface as followed



Read

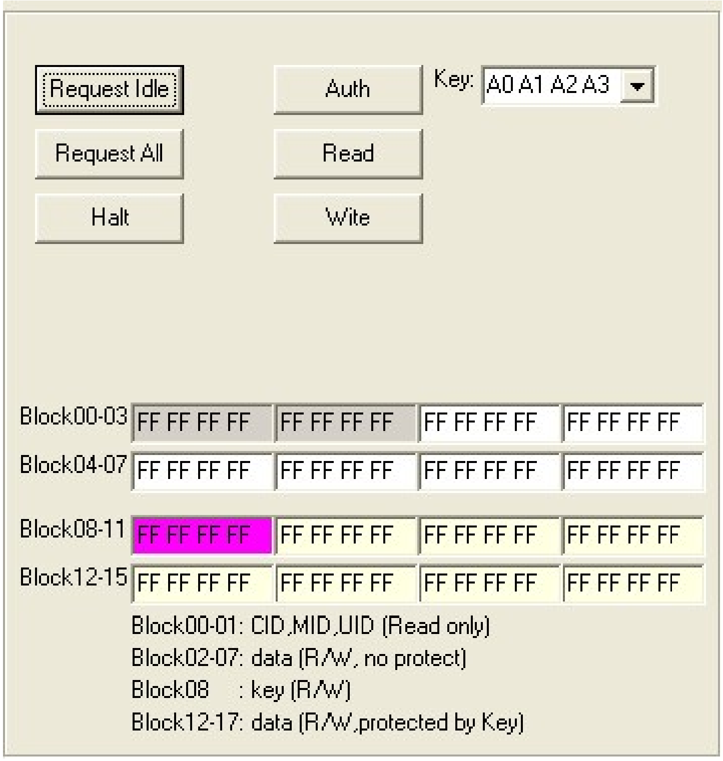
Operate UltraLight card in ACTIVE status. The reader will return 4-page content after StartPage. Each page includes 4 byte.

Write

Operate UltraLight card in ACTIVE status. Only the content of Startpage will be written to UltraLight card. Only one 4-Byte page will be written each time.

SCH1102 functions

This function integrates all the operations to China Huahong card SCH1102 as followed.



Key: A0 A1 A2 A3

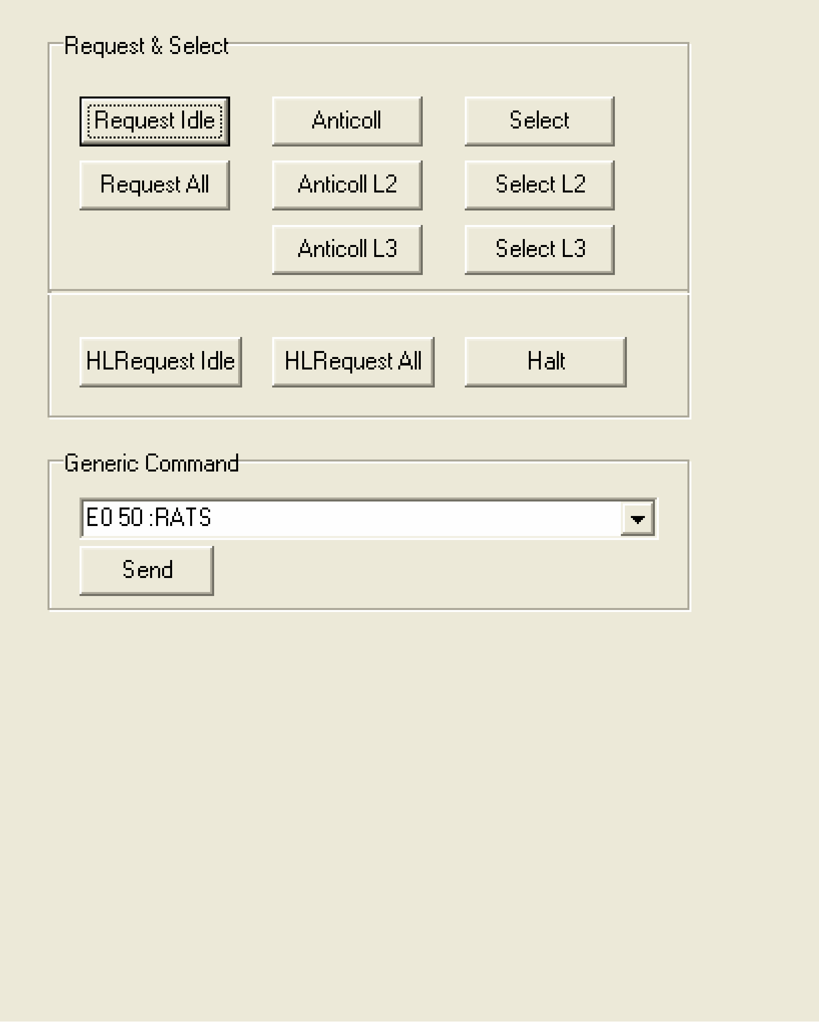
Block00-03	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF
Block04-07	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF
Block08-11	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF
Block12-15	FF FF FF FF	FF FF FF FF	FF FF FF FF	FF FF FF FF

Block00-01: CID,MID,UID (Read only)
Block02-07: data (R/W, no protect)
Block08 : key (R/W)
Block12-17: data (R/W,protected by Key)

Request Idle ISO14443 REQA command
Request All ISO14443 WAKE-UP command
Halt ISO14443 HALT command
Auth Key authentication, the card is 4-byte key protected
Read Read command
Block 00~07 read without key authentication
Block 09~15 key authentication is needed to read the card
Write Write command
Block00, Block01 the block can’t be modified
Block03 ~ Block15 key authentication is needed to erase or modify the block

14443A Functions

The function can finish all the operations to ISO14443 TYPE A.



Generic Command: E0 50 :RATS

Request & Select

Request Idle	**ISO14443A**	**REQA**
Request All	**ISO14443A**	**WAKE-UP**
Anticoll	**ISO14443A**	**Anticoll (UID=4byte)**
Select	**ISO14443A**	**Select (UID= 4byte)**
Anticoll L2	**ISO14443A**	**Anticoll (UID=7 byte)**
Select L2	**ISO14443A**	**Select (UID=7 byte)**
Anticoll L3	**ISO14443A**	**Anticoll (UID=10 byte)**
Select L3	**ISO14443A**	**Select (UID=10 byte)**
Halt	**ISO14443A**	**Halt**

HLRequest Idle

This function will finish the following operations.

Run Request Idle command, and finish Anticoll and Select command automatically according to the returned card type.

For Mifare one card, the reader will run as followed.

Request Idle---- Anticoll----- Select

Mifare one card will enter ACTIVE status from Idle status and return 4 Byte UID.

For UltraLight card, the reader will run as followed.

Request Idle---- Anticoll----- Select------ Anticoll L2------ Select L2

UltraLight card will enter ACTIVE status from Idle status and return 7 Byte UID.

HLRequest All

This function will finish the following operations.

Run Request All command, and finish Anticoll and Select command automatically according to the returned card type.

For Mifare one card, the reader will run as followed.

Request All---- Anticoll----- Select

Mifare one card will enter ACTIVE status from IDLE status or HALT status, and return 4 Byte UID.

For UltraLight card, the reader will run as followed.

Request All---- Anticoll----- Select------ Anticoll L2------ Select L2

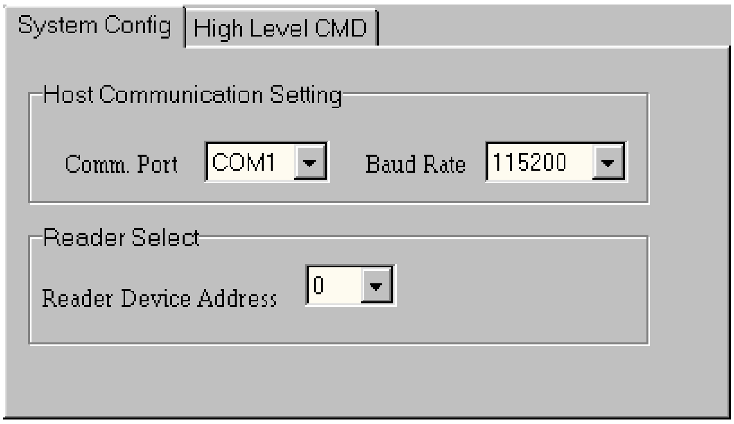
UltraLight card will enter ACTIVE status from IDLE status or HALT status, and return 7Byte UID.

Generic Command

This command channel will send commands to all ISO 14443A cards directly. Data will be returned only when the card must support the command sent. The returned data will be showed in Message Window.

2.4 Secondary Function Group

The Secondary Function Group mainly provides two categories of functions –the System Configuration and High Level Commands.



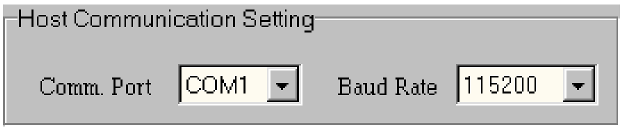
Comm. Port COM1

Baud Rate 115200

Reader Device Address 0

2.4.1 System Config

- **Set Host Communication**



Comm. Port COM1

Baud Rate 115200

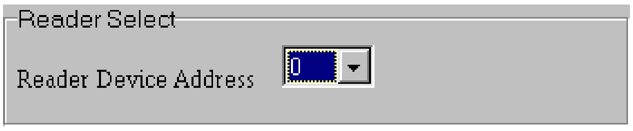
- **COM Port**

  Select the PC serial COM port.

- **Baud Rate**

  Select the Communication Baud rate of the serial port.

- **Reader Select**
  - **Device Address**

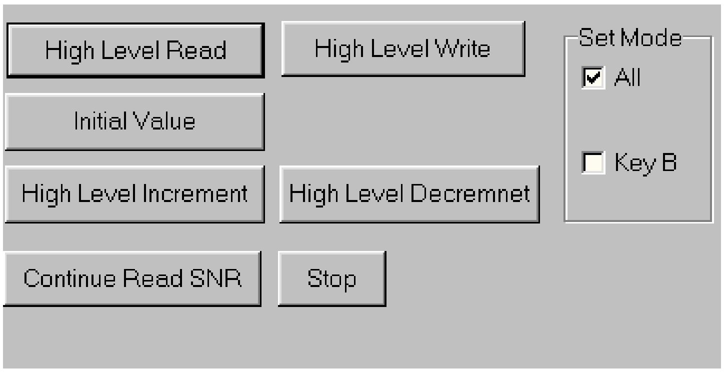


Reader Device Address 0

In multi-drop, the use can select the reader to be communicated by the corresponding address Device. For point to point mode, the Device Address stored within the Reader will be ignored if you select the Device Address as “0’. The Version number of the reader will be shown in the Status Bar at the bottom of the program screen after the right device address is selected.

2.4.2 High Level Commands

This function pages provide the High Level Commands to process the Read/Write and Value related MIFARE commands. The items in the “Set Mode” Check Box Group determine the operating mode of the High Level Command.

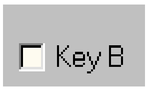


Set Mode: ☑ All □ Key B

There are two items in the “Set Mode ”Box

☑ All

Select to use either “Request All” or “Request Idle” mode for the High Level Command. Checked to use the “Request All” Mode.


□ Key B

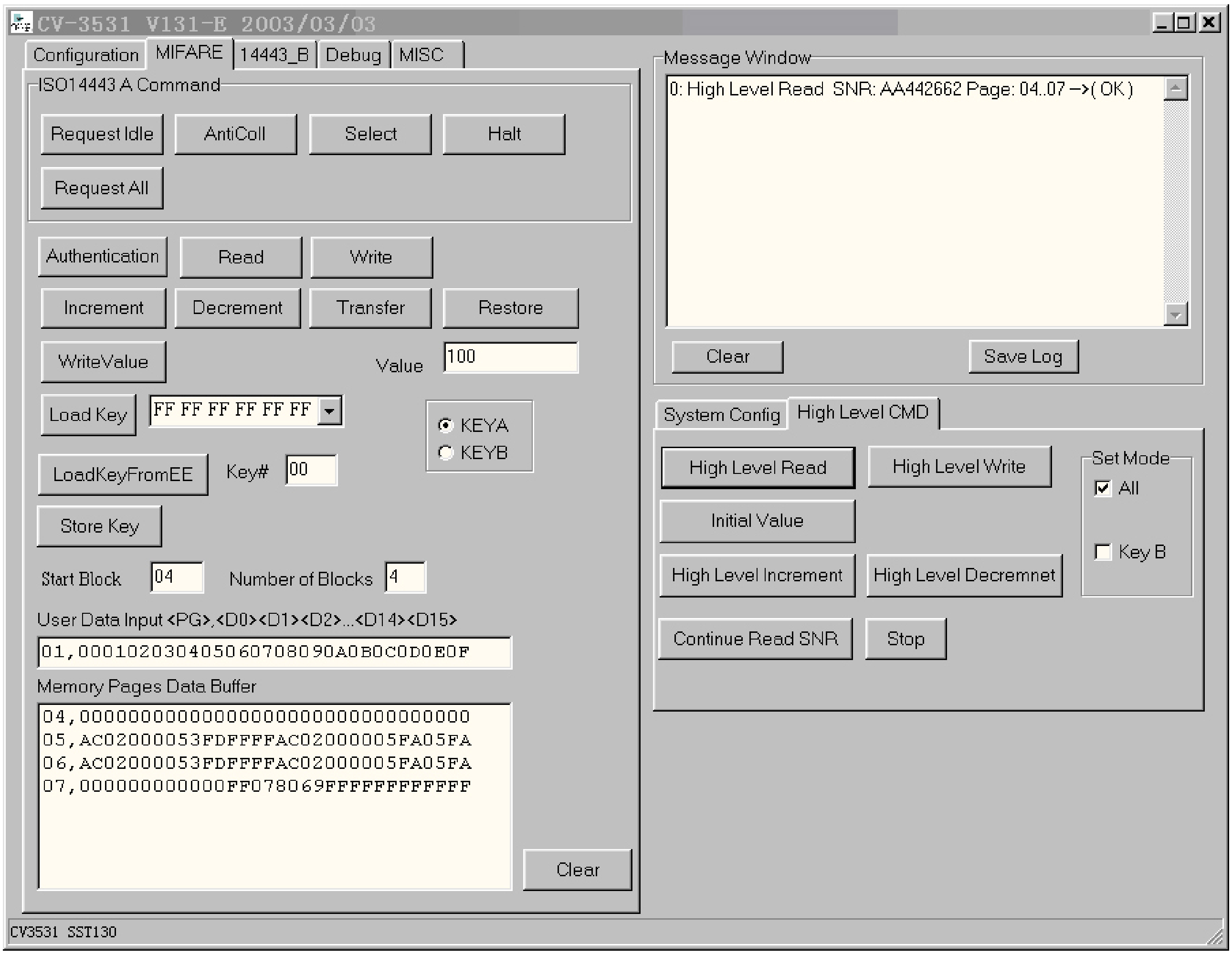
Select to use either “Key Al” or “Key B” for the High Level Command. Checked to use the “Key R”

- **High Level Read/High Level Write**

The operation is same as the Read/Write Command; firstly need to define the number of blocks to be read/written and the start address of the blocks to be processed. But the high level commands integrated the low level commands - “REQUEST, Anti-Collision, Select, Authentication” in a single commands.

The high level commands calculates the corresponding key set, which should be stored into the MFRC500’s internal EEPROM by the “Store Key”, according to the Start Block Address (i.e. the corresponding Sector Number) and use it for the Authentication. The Serial Number of the card to be read/written will be shown in the Message Window. The content of the blocks read will be also displayed in the Memory Page Buffer. The data to be written by the High Level Write Commands should be available in the Memory Pages Buffer. (Just as the same as the Write operation.)

For Example, following is the screen of the result of the High Level Read command.



Value 100

Load Key FF FF FF FF FF FF

● KEYA
○ KEYB

Key# 00

Start Block 04 Number of Blocks 4

User Data Input <PG>,<D0><D1><D2>...<D14><D15>

01,000102030405060708090A0B0C0D0E0F

Memory Pages Data Buffer

```
04,00000000000000000000000000000000
05,AC02000053FDFFFFAC02000005FA05FA
06,AC02000053FDFFFFAC02000005FA05FA
07,000000000000FF078069FFFFFFFFFFFF
```

Message Window

```
0: High Level Read  SNR: AA442662 Page: 04..07 ->( OK )
```

Set Mode
☑ All
□ Key B

CV3531 SST130

- **Initialize Value**
  The selected sector (according to the Address of the Start Block) will be formatted as a sector value with the value in the "Value" Edit Box.
  The Block 1 of the formatted value sector will be used as the prim nary value block and the Block 2 will be used as the value backup block. And the Block0 is free for the user use.

- **High Level Decrement/Increment**
  Same as the High Level Read/Write command but handle the increment and decrement operation instead of the read and write operation

- **Continue Read SNR/Stop**
  Click the "Continue Read SNR" button to read the card serial number and show it on the Message Window continually.
  Click the "Stop" button to break the "Continue Read SNR" command