Functional description / User manual

PR110/PR510
ST110/ST510

PR110 PR510
USER MANUAL

Rev 1.02

1/21/2009

Contents

Revision Index

Revision	Date	Author	Change Record
1.00	2009-11-02	Soemsak	- New document
1.01	2010-01-19	Noppadol	- Added “Support Software for PR110 and PR510” instruction
1.02	2010-01-21	Noppadol	- Added “FCC Statement”

About This Manual

PR110 PR510 User Manual provides general instruction for setup and operation

Service Information

If you have a problem with your equipment, please contact the Advanced ID Asia Co., LTD. Support.

116 M. 3 T.Maekhue, A.Doisaket, Chiangmai 50220, Thailand

Tel: +66 53 387316-7

Fax: +66 53 387319

E-mail: service@aae.co.th, aae@aae.co.th

Product Overview

PR110 provides the best UHF RFID reading performance, reliability, and value available. Its physical design, which combines a lightweight yet solid feel with excellent ergonomics for scanning, ensures comfortable use.

PR110 hand-held UHF RFID reader supports reading ISO18000-6C (EPC Class1 Gen2) transponders.

Getting Start

Wake Up Reader

Press “POWER” button to wake up the reader from the sleep mode.

Change Reading/Writing Mode

Press “MODE” button to change the reading/writing mode between “READ” and “WRITE”.

Change Reading/Writing Memory Bank

Press “DISPLAY” button to change the reading/writing memory bank.

If the reading/write tags type is in “EPC Class1 Gen2”, user can select reading/writing memory bank as the list below:

- EPC – to read/write the EPC number with anti-collision algorithm.
- ACCESS/KILL – to read/write access or kill code (memory bank 00) from the specific start address and data length.
- UII/EPC – to read/write UII (memory bank 01) from the specific start address and data length.
- TID – to read/write TID (memory bank 02) from the specific start address and data length.
- USER – to read/write USER (memory bank 03) from the specific start address and data length.

Note: Please check again the specification of the tag that you are attempting to read/write whether it supports reading/writing all of the banks.

Change Reading/Writing Memory Offset

Press “USER” button to change the reading/writing memory offset. User can set the new offset by changing the start address and data length.

Change Settings

Press “MENU” button to change the reader’s settings. In this mode user will find 4 submenus and bellow:

- DATE TIME – user can set date time of the reader in this mode.
- ERASE MEM – user can erase the flash memory of the reader in this mode
- VERSION – This mode shows the firmware and the hardware version of the unit
- BLUETOOTH – user can enable/disable and set the security key for Bluetooth connection in this mode.

Communication to PC

Physical layer

PR110 UHF RFID READER will be connected to the PC via USB Serial Port (FTDI chip http://www.ftdichip.com). USB Serial Port provides the conversion of Universal Serial Bus (USB) to RS232 Serial Port. The configuration of the serial port will be as below:

Baud Rate: 38400

Data Bits: 8

Rarity: None

Stop Bits: 1

Flow Control: None

Data Link Layer

The data sent to the PC is originally an array of bytes with all possible HEX values. In order to make programming of PC as well as understanding the data easier, each HEX value is converted to ASCII values.

For example: one byte 0xA3 is converted to its ASCII characters to ‘A’ = 0x41 and ‘3’ = 0x33. This means that if the original array had N bytes the n the PC packet has 2xN bytes or N words as each pair of characters compose one byte of original data.

Command structure

The command structure is as shown below:

Command Number (2 chars)	Parameter (variable)	Carriage return, <CR>(1char)

Where:

Command Number (2chars) defines which command the user wants to set, get or control the configuration in reader.

Parameter defines the data using with the command number.

Carriage Return<CR> is always 1 char of 0x0D ('\r' in C language).

For example, the massage in Data Link Layer:

1000 <CR>

That corresponds to set PR110 UHF RFID READER to read ISO18000-6B tag where "10" is the command number of "Set Tag Type"(page 12) and "00" is the parameter of "ISO18000-6B".

Command Descriptions

Error Message (FF)

Description: Error Message will be sent from the PR110 UHF RFID READER to PC when PR110 UHF RFID READER receives any unknown command number.

Type: Error command with reply.

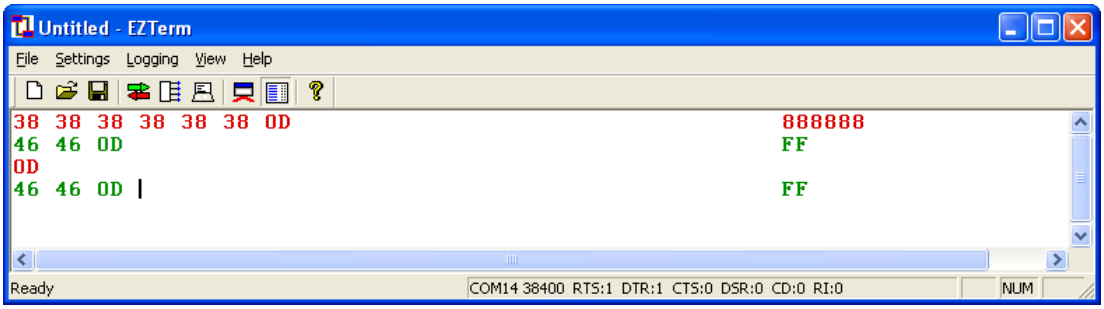
Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (variable)	Carriage return, <CR>(1char)
Any unknown command number	Any Parameters	<CR>

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
FF	-	<CR>

Example:



Note: You can download the software EZTerm to test the protocol from the websites http://www.integrityusa.com/

Get Firmware Version (00)

Description: Returns firmware version.

Type: Management command with reply.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
00	-	<CR>

Where:

"00" is the command number.

Returns from PR110 UHF RFID READER:

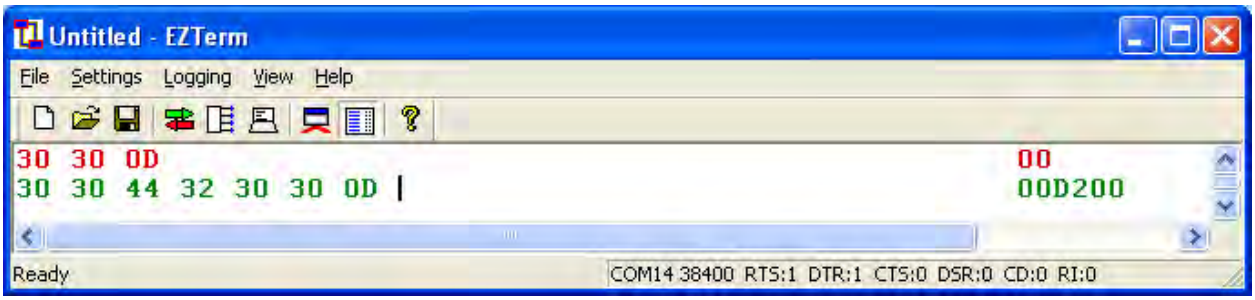
Command Number (2 chars)	Parameter (4 chars)	Carriage return, <CR>(1char)
00	Ex. 0D00	<CR>

Where:

"00" is the echo of the command number.

"0D00" is the firmware version.

Example:



Get Hardware Version (01)

Description: Returns hardware version.

Type: Management command with reply.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
01	-	<CR>

Where:

"01" is the command number.

Returns from PR110 UHF RFID READER:

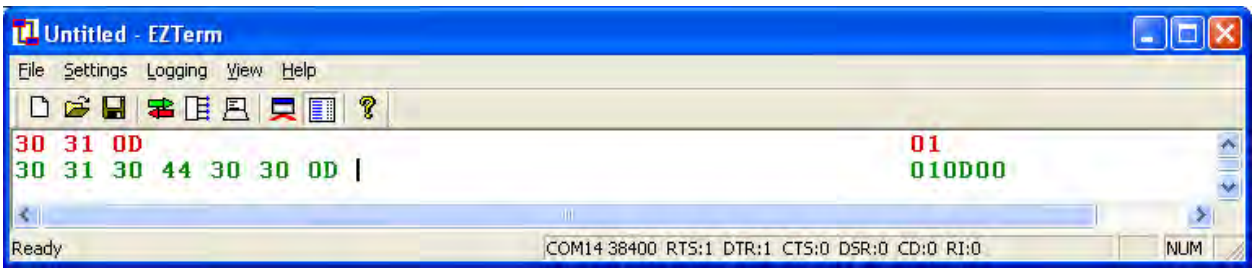
Command Number (2 chars)	Parameter (4 chars)	Carriage return, <CR>(1char)
01	Ex. 0D00	<CR>

Where:

"01" is the echo of the command number.

"0D00" is the hardware version.

Example:



Get Reader ID (02)

Description: Returns Reader ID.

Type: Management command with reply.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
02	-	<CR>

Where:

"02" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (16 chars)	Carriage return, <CR>(1char)
02	Ex. 2008091100000001	<CR>

Where:

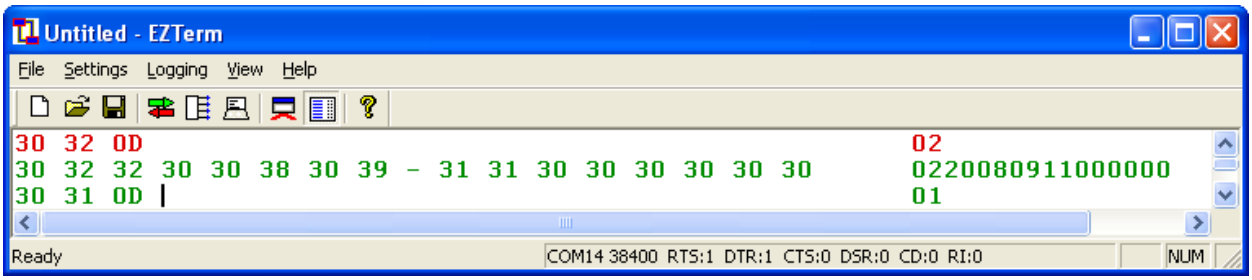
"01" is the echo of the command number.

"2008091100000001" is the produced date and serial number.

Parameter:

Parameter (16chars)	
Produced date (8 chars)	Serial number (8 chars)
20080911 – 2008/September/11	00000000 to FFFFFFFF

Example:



Get Reader Model (03)

Description: Returns Reader Model.

Type: Management command with reply.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
03	-	<CR>

Where:

"03" is the command number.

Returns from PR110 UHF RFID READER:

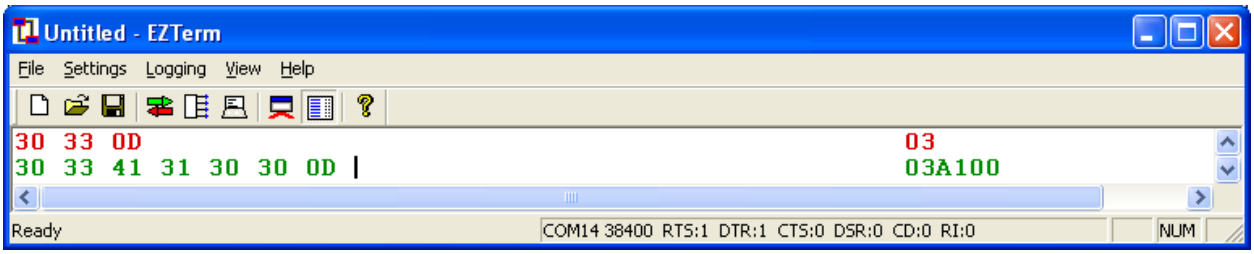
Command Number (2 chars)	Parameter (4 chars)	Carriage return, <CR>(1char)
03	Ex. A100	<CR>

Where:

"03" is the echo of the command number.

"A100" is the reader model.

Example:



Get Reader Status (04)

Description: Returns working status of the reader.

Type: Management command with reply.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
04	-	<CR>

Where:

"04" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 chars)	Carriage return, <CR>(1char)
04	Ex. 04	<CR>

Where:

"04" is the echo of the command number.

"04" is the status.

Parameter:

Parameter (2chars)
Reader status (2 chars)
00 – Initializing, Reader is initializing itself from sleep mode. 01 – Standby, Reader is ready to work. 02 – Starting RF Power, Reader is starting the RF part. 03 – Stopping RF Power, Reader is stopping the RF part. 04 – Reading, Reader is trying to read RFID tags. 05 – Writing, Reader is trying to write data to a RFID tag. 06 – Locking, Reader is trying to lock a RFID tag. 07 – Killing, Reader is trying to kill a RIFID tag

Example:

Set Tag Type (10)

Description: Defines the RFID protocol that the reader is going to read (ISO18000-6B or EPC Class1 Gen2).

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 char)	Carriage return, <CR>(1char)
10	00	<CR>

Where:

"10" is the command number.

"00" is tag type ISO18000-6B.

Parameter:

Parameter (2 chars)
Tag Type (2 chars)
00 – ISO18000-6B 01 – ISO18000-6C(EPC Class1 Gen2)

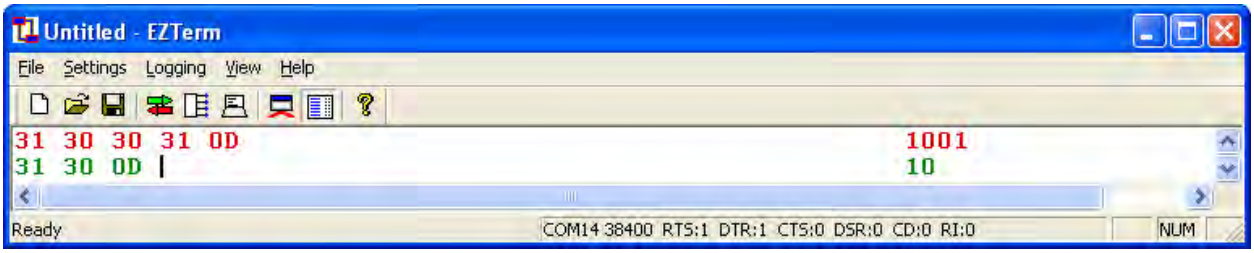
Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
10	-	<CR>

Where:

"10" is the echo of the command number.

Example:



Set Bank (11)

Description: Defines reading bank that the reader is going to read.

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 char)	Carriage return, <CR>(1char)
11	04	<CR>

Where:

"10" is the command number.

"04" is reading bank.

Parameter Case 1: if the reader was set Tag Type to ISO18000-6B.

Parameter (2 chars)
Reading bank (2 chars)
00, 01, 02, 03 – Read block data from the specific address. 04 – Read 8 bytes UID (unique ID).

Parameter Case 2: if the reader was set Tag Type to ISO18000-6C(EPC Class1 Gen2).

Parameter (2 chars)
Reading bank (2 chars)
00 – Read Reserve bank at the specific address. 01– Read UII bank at the specific address. 02– Read TID bank at the specific address. 03 – Read User Memory bank at the specific address. 04 – Read Multi EPC number with anti-collision process.

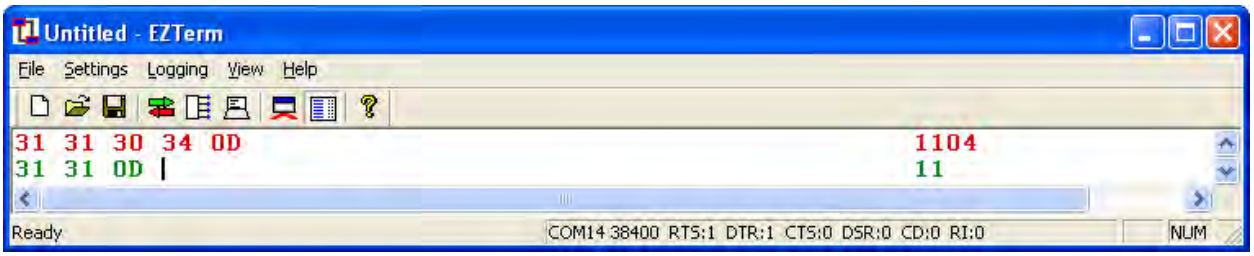
Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
11	-	<CR>

Where:

"11" is the echo of the command number.

Example:



Set Start Address (12)

Description: Defines the start address where the reader will read out from the tag memory when the reader was set Read Bank to:

- Read block data from the specific address
- Read Reserve bank at the specific address
- Read TID bank at the specific address
- Read User Memory bank at the specific address
- Read User Memory bank at the specific address

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 char)	Carriage return, <CR>(1char)
12	00	<CR>

Where:

"12" is the command number.

"00" is the start address.

Parameter:

Parameter (2 chars)
Start address (2 chars)
00 to FF

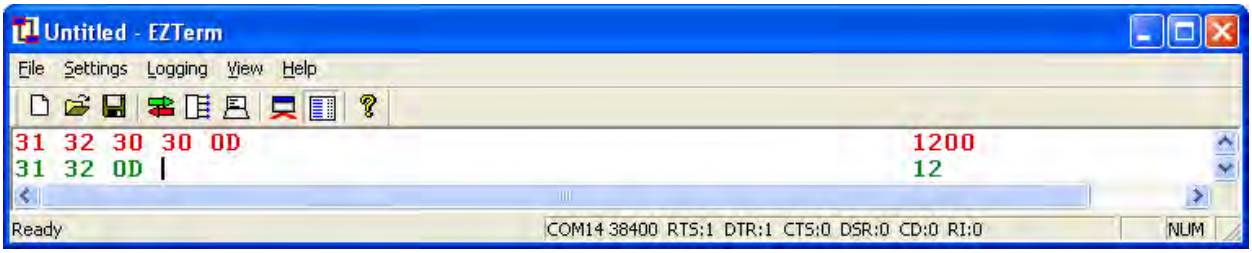
Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
12	-	<CR>

Where:

"12" is the echo of the command number.

Example:



Set Data Length (13)

Description: Defines the data length that the reader will read out from the tag memory where starts at the Start Address (3.9 page 16) when the reader was set Read Bank to:

- Read block data from the specific address
- Read Reserve bank at the specific address
- Read TID bank at the specific address
- Read User Memory bank at the specific address
- Read User Memory bank at the specific address

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 char)	Carriage return, <CR>(1char)
13	08	<CR>

Where:

"13" is the command number.

"08" is the data length.

Parameter:

Parameter (2 chars)
Data length (2 chars)
00 to FF

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
13	-	<CR>

Where:

"13" is the echo of the command number.

Example:

Get Tag Type (20)

Description: Get the configuration of the RFID protocol that the reader is going to read (ISO18000-6B or EPC Class1 Gen2).

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
20	-	<CR>

Where:

"20" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2chars)	Carriage return, <CR>(1char)
20	00	<CR>

Where:

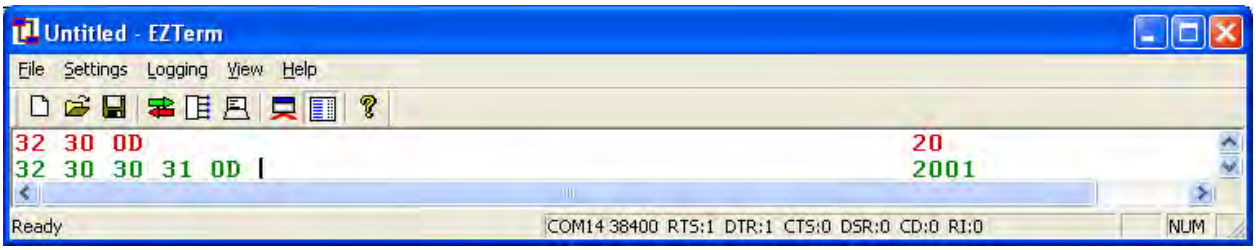
"20" is the echo of the command number.

"00" is tag type ISO18000-6B.

Parameter:

Parameter (2 chars)
Tag Type (2 chars)
00 – ISO18000-6B 01 – ISO18000-6C(EPC Class1 Gen2)

Example:



Get Bank (21)

Description: Get the configuration of the reading bank that the reader is going to read.

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
21	-	<CR>

Where:

"21" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2chars)	Carriage return, <CR>(1char)
21	04	<CR>

Where:

"21" is the echo of the command number.

"04" is reading bank.

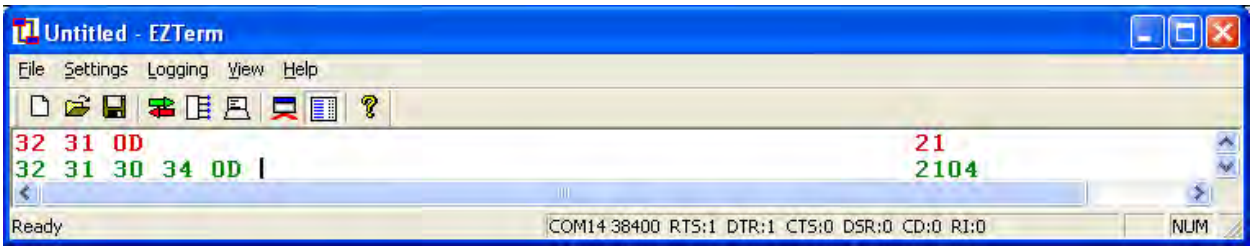
Parameter Case 1: if the reader was set Tag Type to ISO18000-6B.

Parameter (2 chars)
Reading bank (2 chars)
00, 01, 02, 03 – Read block data from the specific address. 04 – Read 8 bytes UID (unique ID).

Parameter Case 2: if the reader was set Tag Type to ISO18000-6C(EPC Class1 Gen2).

Parameter (2 chars)
Reading bank (2 chars)
00 – Read Reserve bank at the specific address. 01– Read UII bank at the specific address. 02– Read TID bank at the specific address. 03 – Read User Memory bank at the specific address. 04 – Read Multi EPC number with anti-collision process.

Example:



Get Start Address (22)

Description: Get the configuration of the start address where the reader will read out from the tag memory when the reader was set Read Bank to:

- Read block data from the specific address
- Read Reserve bank at the specific address
- Read TID bank at the specific address
- Read User Memory bank at the specific address
- Read User Memory bank at the specific address

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
22	-	<CR>

Where:

"22" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 chars)	Carriage return, <CR>(1char)
22	00	<CR>

Where:

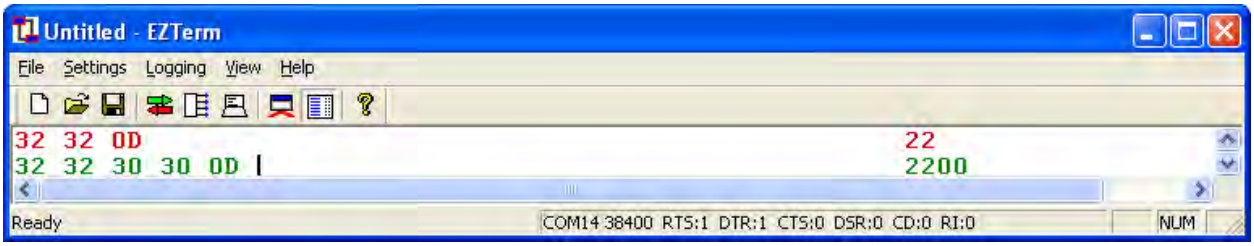
"22" is the echo of the command number.

"00" is the start address.

Parameter:

Parameter (2 chars)
Start address (2 chars)
00 to FF

Example:



Set Data Length (23)

Description: Get the configuration of the data length that the reader will read out from the tag memory where starts at the Start Address (3.9 page 16) when the reader was set Read Bank to:

- Read block data from the specific address
- Read Reserve bank at the specific address
- Read TID bank at the specific address
- Read User Memory bank at the specific address
- Read User Memory bank at the specific address

Type: RFID protocol configuration command with Parameter.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
23	-	<CR>

Where:

"23" is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 chars)	Carriage return, <CR>(1char)
23	08	<CR>

Where:

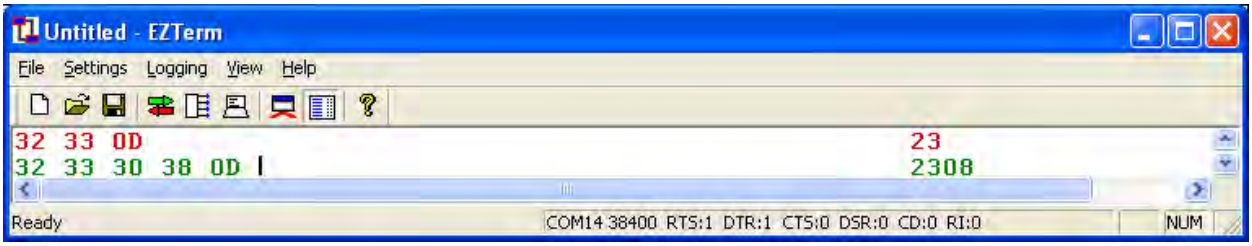
"23" is the echo of the command number.

"08" is the data length.

Parameter:

Parameter (2 chars)
Data length (2 chars)
00 to FF

Example:



Stop operation (30)

Description: Stops or cancels all of operations working with the RF part (reading, writing, locking and killing). After sending this command, the reader status will be "Stopping RF Power" and then "Standby" (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
30	-	<CR>

Where:

"30" is the command number.

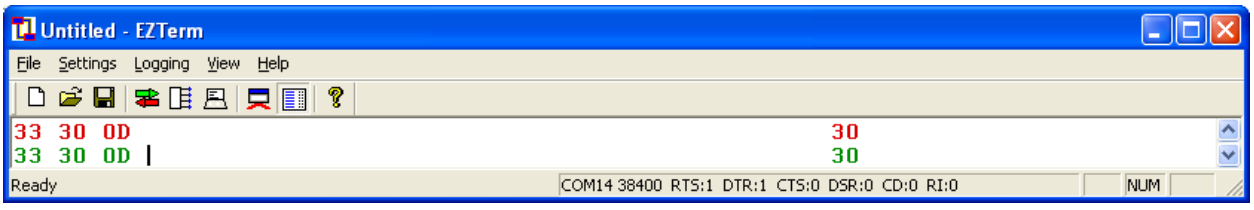
Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
30	-	<CR>

Where:

"30" is the echo of the command number.

Example:



Start Auto-Reading (31)

Description: Starts reading RFID tags and continuously sends data read from tags to the PC. After sending this command, the reader status will be “Starting RF Power” and then “Reading” (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
31	-	<CR>

Where:

“31” is the command number.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
31	-	<CR>

Where:

“31” is the echo of the command number.

After: When the reader can read data from RFID tags, the reader will send automatically data to the PC.

Returns the tag data to PC:

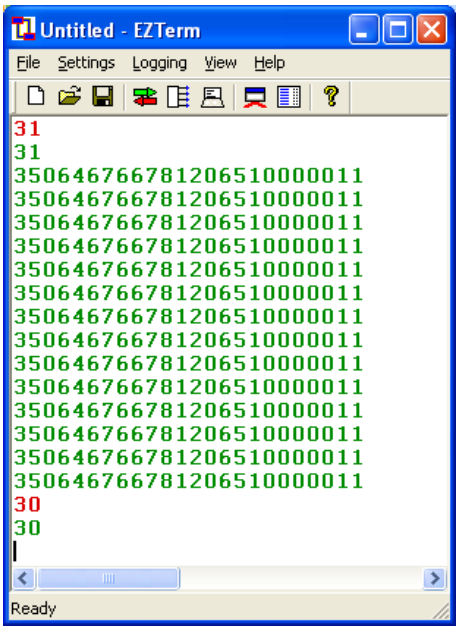
Parameter (variable)	Carriage return, <CR>(1char)
Ex. 00112233445566778899112 2 Or, E0045BA51A010000	<CR>

Where:

"00112233445566778899112 2" is the 12 bytes EPC number from an EPC C1 Gen2 tag.

"E0045BA51A010000" is the 8 bytes UID from an ISO18000-6B tag.

Example:



First the PC sent "31<CR>" to start reading.

Then, the reader read and returned the EPC number "350646766781206510000011" read from an EPC C1 Gen2 tag.

Then the PC sent "30<CR>" to stop reading.

Polling Read (32)

Description: Polling reading a single tag until success. After sending this command, the reader status will be “Starting RF Power” and then “Reading” (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 char)	Carriage return, <CR>(1char)
32	-	<CR>

Where:

“32” is the command number.

Returns from PR110 UHF RFID READER:

Parameter (variable)	Carriage return, <CR>(1char)
Ex. 001122334455667788991122 Or, E0045BA51A010000	<CR>
Returns nothing until success.	-

Where case 1: the reader can read a tag in the read range.

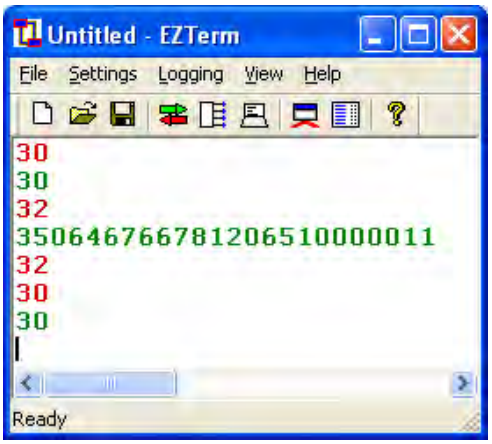
“001122334455667788991122” is the 12 bytes EPC number from an EPC C1 Gen2 tag.

“E0045BA51A010000” is the 8 bytes UID from an ISO18000-6B tag.

Where case 2: the reader has not yet got any tag in the read range.

No response.

Example:



First, the PC sent “30” to stop any running operations.

Then, the PC sent “32” to read the data from a tag.

Then, the Reader can get data from a tag in the read range and the reader returned the EPC number “350646766781206510000011” from the tag.

Then, the PC sent again “32” to read the data from a tag but this time there is no tag in read range. The reader had no response.

Then, the PC sent “30” to stop reading.

Write Data or ID (40)

Description: Writes data or ID to a tag at the specific bank (3.8, page 14) and the specific address (3.9, page 16). After sending this command, the reader status will be “Starting RF Power” and then “Writing” (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 – 64 chars)	Carriage return, <CR>(1char)
40	1122AABB Or 11	<CR>

Where:

“40” is the command number.

“1122AABB” and “11” is the **HEX** data you want to write into the tag.

Note1: 1word = 2 bytes

1 word of ISO18000-6C (EPC Class1 Gen2) is **2 bytes**. Therefore, when you want to write data into an EPC C1 Gen2 tag, you must enter the parameter in the length of 2, 4, 6, 8 and so on.

Note2: Write ID Mode

When the reader is set Reading Bank (3.8, page 14) to “Read Multi EPC number with anti-collision process”, the reader will be in “Writing ID Mode”. In this mode the reader will write the data into the UII bank (01) and start at the address 0x02 and the reader will calculate the length of EPC number and write into PC(protocol control) block in the tag automatically.

Note3: Write Data Mode

When the reader is set Reading Bank (3.8, page 14) to any others from “Read Multi EPC number with anti-collision process”, the reader will be in “Writing Data Mode”. In this mode the reader will write the data into the bank that was set in Reading Bank (3.8, page 14) and start at the address that was set in Start Address (3.9, page 16).

Returns from PR110 UHF RFID READER:

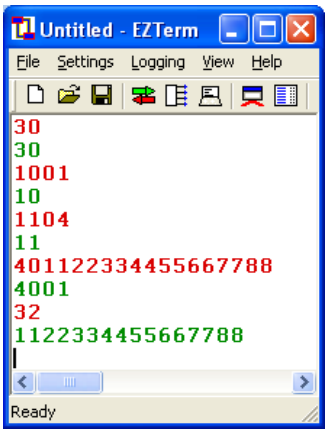
Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
40	01	<CR>

Where:

“40” is the command number.

“01” – “Success”

Example1: Write ID into the EPC C1 Gen2 tag.



First, PC sent “30<CR>” to stop running operation.

Then, PC sent “1001<CR>” to set the tag type to ISO18000-6C (EPC C1 Gen2).

Then, PC sent “1104<CR>” to set the reading bank to “Read Multi EPC number with anti-collision process”

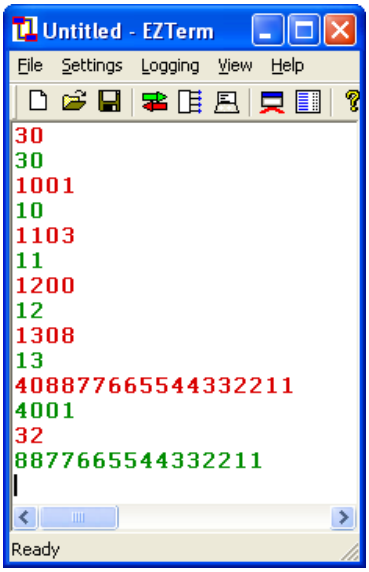
The PC sent “401122334455667788<CR>” to write the EPC number into the tag.

The reader succeeded to write the ID to the tag and returned “4001<CR>”

The PC sent “32<CR>” to read and verify the data.

The reader could get the EPC number returned from the tag “1122334455667788<CR>”.

Example2: Write Data into the EPC C1 Gen2 tag.



First, PC sent “30<CR>” to stop running operation.

The PC sent “1001<CR>” to set the tag type to ISO18000-6C (EPC C1 Gen2).

The PC sent “1103<CR>” to set the reading bank to “Read User Memory bank at the specific address”

The PC sent “1200<CR>” to set the start address to 0x00.

The PC sent “1308<CR>” to set the start address to 0x08.

The PC sent “40887766554433221 1<CR>” to write the data into the tag.

The reader succeeded to write the data to the tag and returned “4001<CR>”

The PC sent “32<CR>” to read and verify the data.

The reader could get the EPC number returned from the tag “887766554433221 1<CR>”.

Lock (50)

Description: Permanently Locks a tag at the specific bank or the specific address. After sending this command, the reader status will be “Starting RF Power” and then “Locking” (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (2 chars)	Carriage return, <CR>(1char)
50	01	<CR>

Where:

“50” is the command number.

“01” is the bank or the address will be locked.

Parameter case 1: the tag type is ISO18000-6B.

Parameter (2 chars)
Data length (2 chars)
00 to FF

Where:

“00” to “FF” is the address you want to lock.

Parameter case 2: the tag type is ISO18000-6C (EPC Class1 Gen2).

Parameter (2 chars)
Bank (2 chars)
00 – Reserve 01 – UII 02 – TID 03 - Data

Where:

"00" to "03" is the bank you want to lock.

Returns from PR110 UHF RFID READER:

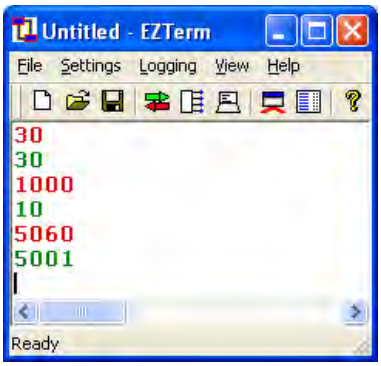
Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
50	01	<CR>

Where:

"50" is the command number.

"01" – "Success"

Example:



First, PC sent “30<CR>” to stop running operation.

The PC sent “1000<CR>” to set the tag type to ISO18000-6B.

The PC sent “5060<CR>” to lock the tag at the address 0x60

The reader returns “5001<CR>” when succeeded.

Kill (51)

Description: Permanently kill an EPC C1 Gen2 tag. After sending this command, the reader status will be “Starting RF Power” and then “Killing” (3.6, page 10).

Type: Reader operation command.

Transmits from PC to PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (4 chars)	Carriage return, <CR>(1char)
51	11223344	<CR>

Where:

“51” is the command number.

“11223344” is the kill code.

Returns from PR110 UHF RFID READER:

Command Number (2 chars)	Parameter (0 chars)	Carriage return, <CR>(1char)
51	01	<CR>
Or returns nothing		

Where:

"51" is the command number.

"01" – "Success"

Or returns nothing

Support Software for PR110 and PR510

This chapter describes how to use PC software “ST-PR Demo Software” to control and download data from PR110 and PR510

Software requirement

Below are programs required to install on your PC before setup ST-PR Demo Software, if these programs already installed you can skip this step and go to next section.

- Microsoft Dot Net Framework 2.0 or above <Download>

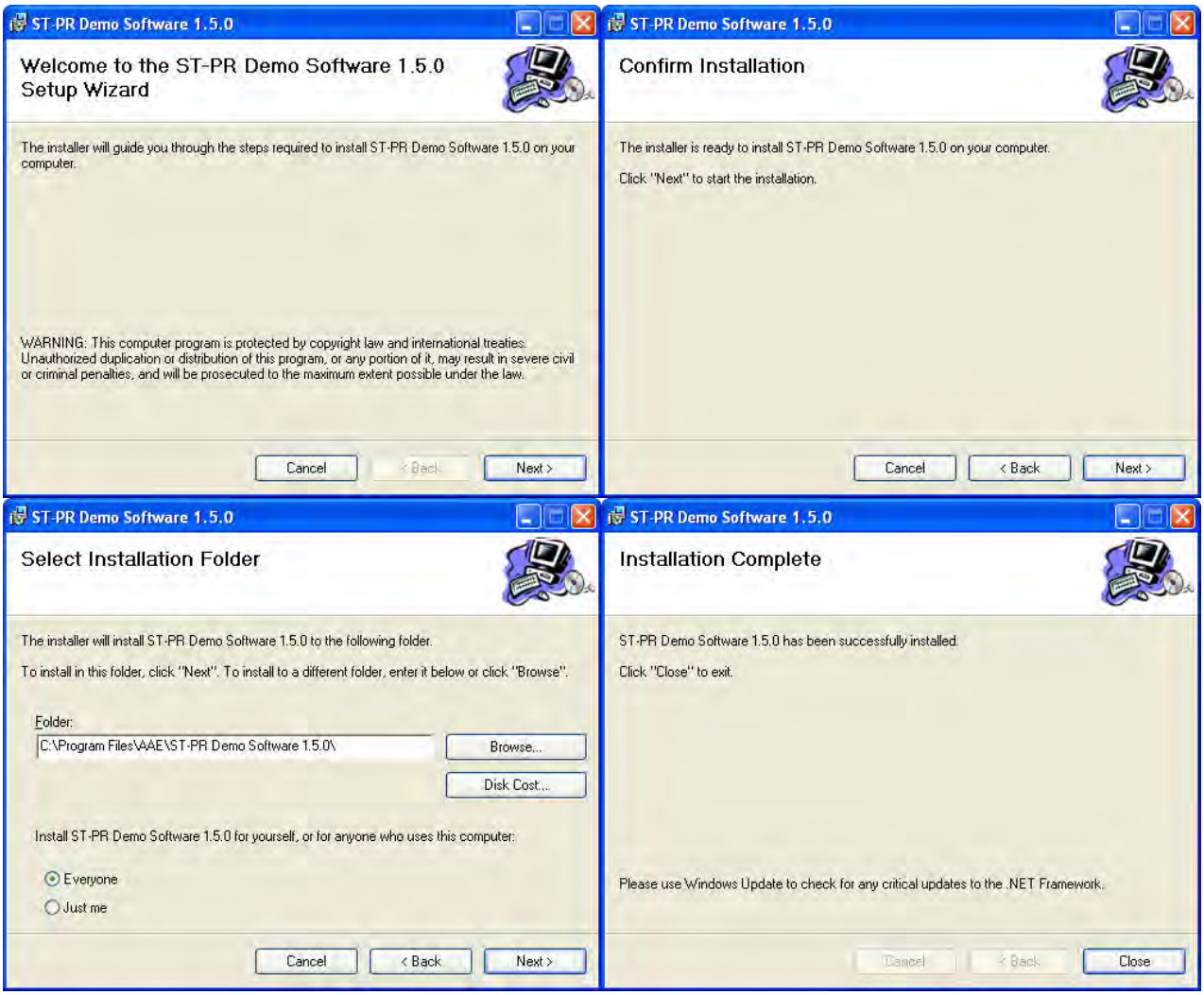
- Windows Installer 3.1 <Download>

ST-PR Demo Software

1. Installation ST-PR Demo Software

- To install ST-PR Demo Software, double click “Setup ST100 Demo Software 1.x.msi” icon.

- Follow instructions in the installation dialog until installation complete.



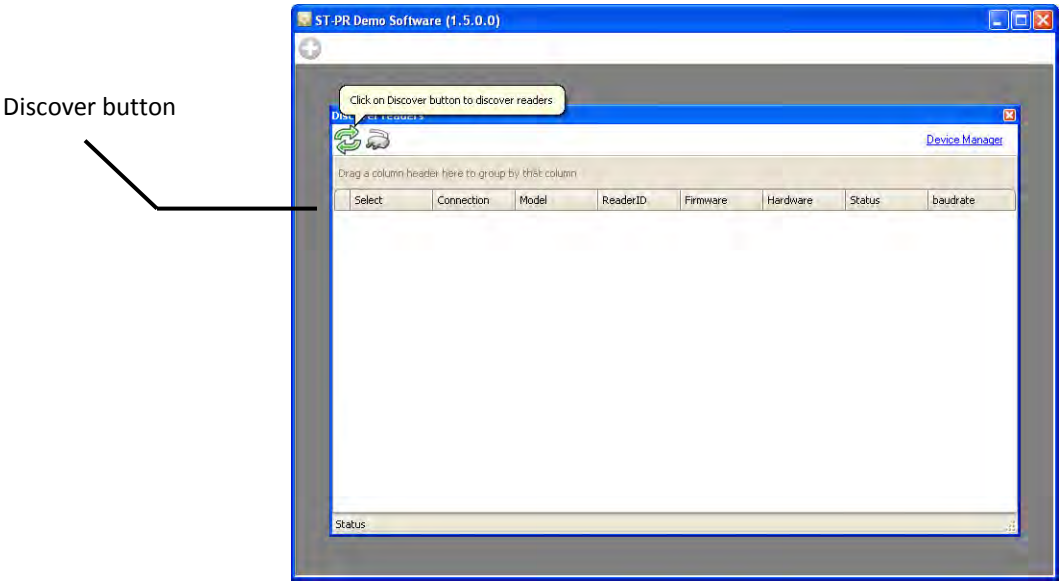
2. Using ST-PR Demo Software

To start “ST-PR Demo Software”,
- Connect your RFID Reader with PC
- Switch on the Reader
- Double click ST-PR Demo Software 1.x desktop shortcut or in Start menu shortcut.

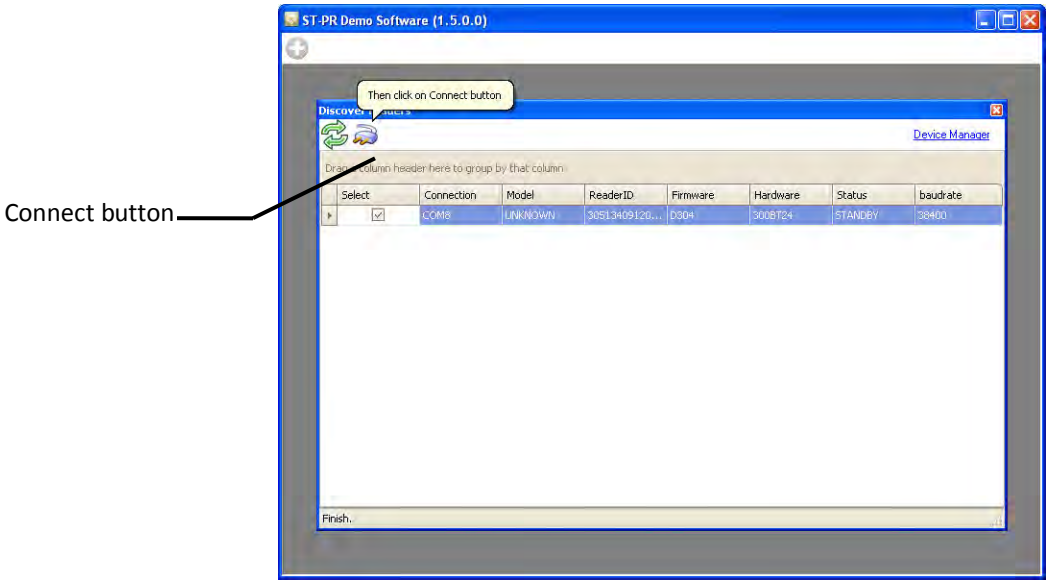
3. Readers Connection

To connect and communicate with your reader,

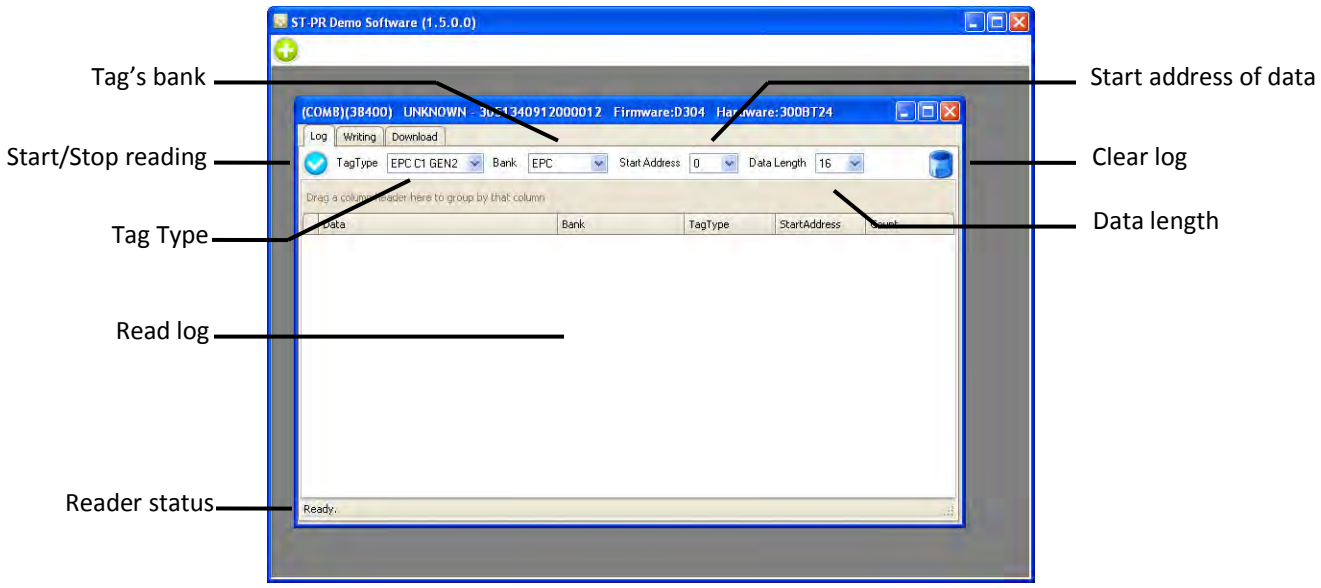
- Click “Discover” button on the top of Discover readers dialog



- Select your reader and click “Connect” button



4. Read tag data using Log tab

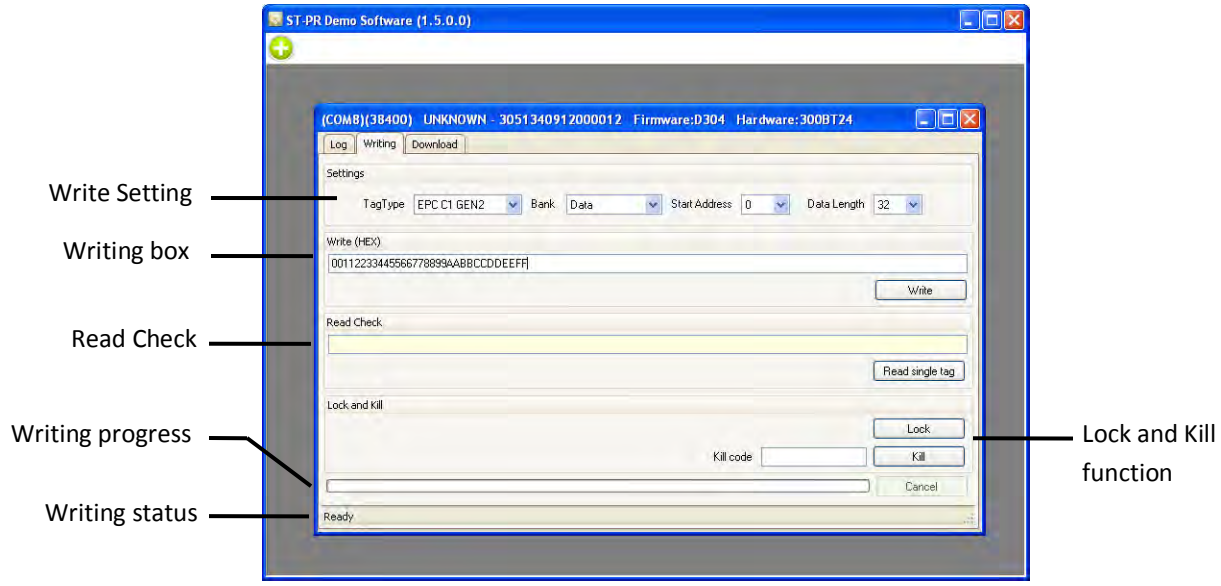


This tab uses for control read command of the Reader following,

- To read data from tags, click **Start reading button** and click **Stop reading button** to stop reading
- To read data from the specific bank using **Bank** dropdown, specific data address using **Start address** dropdown and specific data length with **Data Length** dropdown (Data length depends on the tag bank)

5. Write data to a tag using Writing tab

This tab uses for control writing function. You can write data in each bank of tag, read check, lock and kill tag.

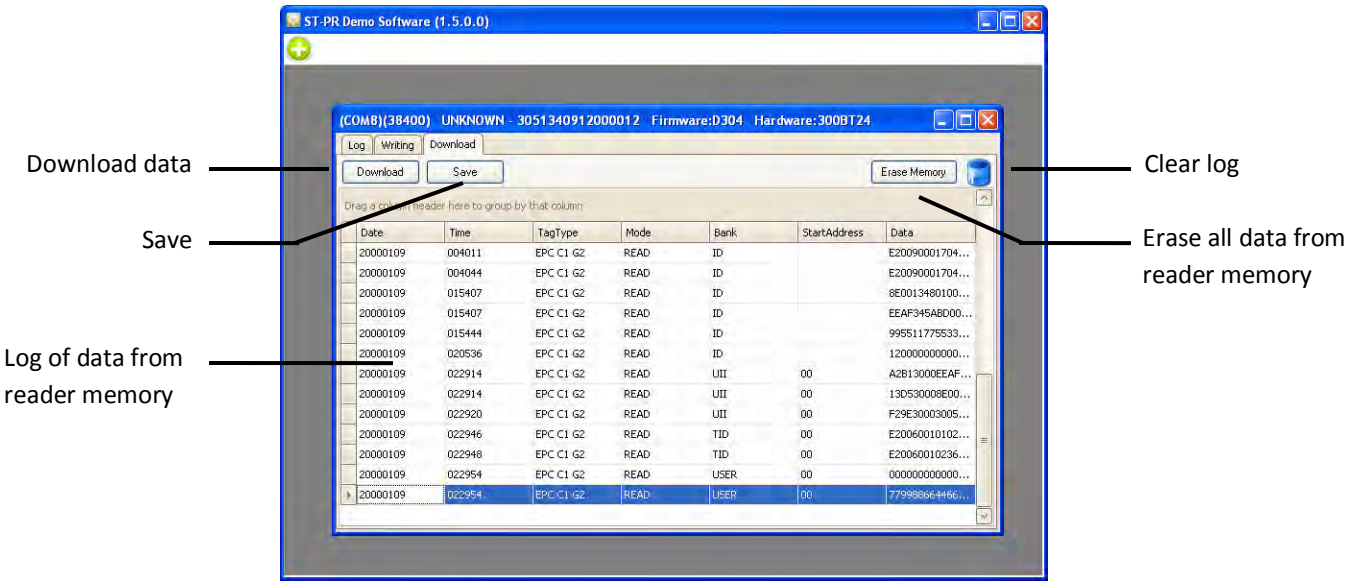


To write data to a tag

- Set Tag Type, specific Bank, Start address and Data length
- Input data depends on data length setting in Settings box
- Put a tag in the reader range and click **Write** button. "Success" message will show in the status bar after finish writing.

6. Download data from Reader's memory using Download tab

There are functions to download data from reader memory in this tab. Downloaded data is able to export in the XLS format using save menu and this tab also provides user to erase all data in reader memory.

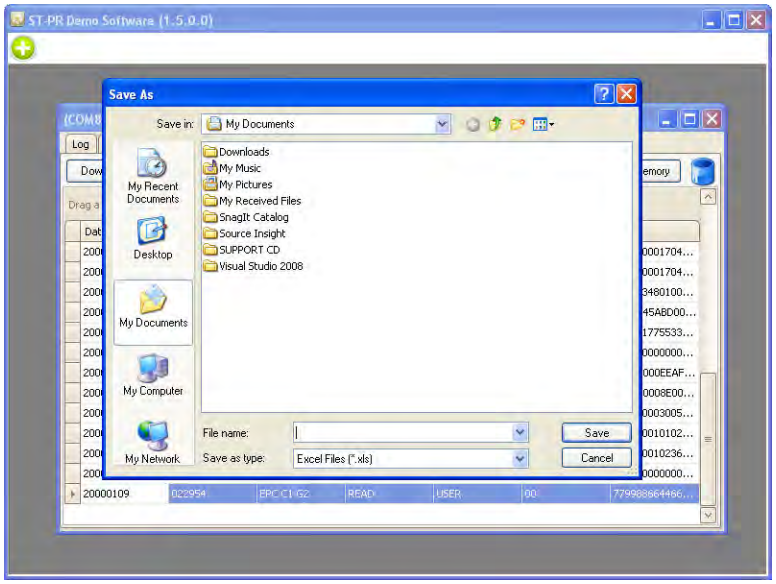


To download data from reader memory

- Click **Download** button, data will be downloaded to log table. After finish downloading, you can save the data in log in XLS format.

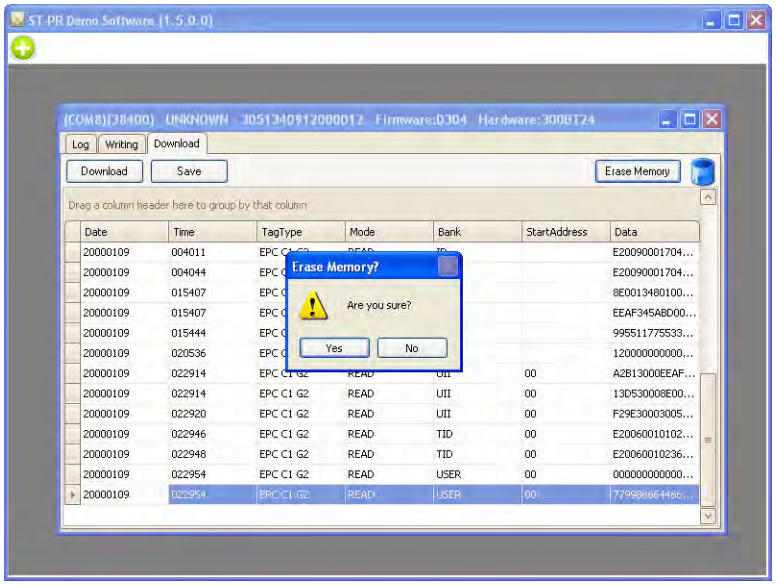
To save downloaded data

- Click **Save** button. Save dialog will appear.
- Select save location, input file name and click Save



To erase all data from reader memory

- Click **Erase memory** button
- Erase Memory dialog will appear, click Yes to confirm or click No to cancel erase memory and return to Download tab



To clear downloaded data in log table, click Clear icon

FCC Statement

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 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.

To assure continued compliance, any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment. (Example - use only shielded interface cables when connecting to computer or peripheral devices).

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and
(2) This device must accept any interference received, including interference that may cause undesired operation.

FCC Radiation Exposure Statement:

(if mobile device (generally 20 cm distance) is applicable, MPE calculation)

“This equipment complies with FCC RF radiation exposure limits set forth for an uncontrolled environment. The antenna(s) used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter”.

Declaration of Conformity for USA

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and

(2) This device must accept any interference received, including interference that may cause undesired operation.

Usually this is followed by the following FCC caution: Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

For **Canada** the following text has to be included in the user manual: (See RSS-210 clause 5.11)

Operation is subject to the following two conditions:

1) This device may not cause interference and

2) This device must accept any interference, including interference that may cause undesired operation of the device

If the antenna is detachable the following sentences according to RSS-210 clause 5.5/5.11 need to be included:

“This device has been designed to operate with an antenna having a maximum gain of .. dB. Antenna having a higher gain is strictly prohibited per regulations of Industry Canada. The required antenna impedance is 50 ohms”.

“To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the EIRP is not more than required for successful communication”.

ST110 ST510

User Manual and Communication Protocol

REV 1.02
1/21/2010

Contents

Revision Index

Revision	Date	Author	Change Record
1.00	2009-11-28	Soemsak	- New document
1.01	2010-01-12	Noppadol	- Added (6) Using ST110 Reader Tool
1.02	2010-01-21	Noppadol	- Added “FCC Statement”

About Advanced ID Asia Engineering Co., Ltd.

Advanced ID Asia Engineering Co., Ltd. is a provider of RFID reader hardware solution in the LF, HF, UHF segments of the RFID marketplace.

Advanced ID Asia Engineering Co., Ltd.

116 Moo 3 T.Maekhue, A.Doisaket,

Chiangmai 50220, Thailand.

Tel. +66 53 387316-7, Fax. +66 53 387319

Website: http://www.aae.co.th

Email: sales@aae.co.th

1 INTRODUCTION

This document describes the communication protocol that is used for the ST110 and ST510. The protocol was designed to work with ST110, ST510 and independent of which interface is used for the communication. So it is a very small and essential protocol but has the possibility to be enhanced. In the following chapters the structures and the values of the protocol are described.

2 PACKET TRANSMISSIONS

This protocol does not depend on the interface which is used for the communication. So with the use of every interface the data structure is the same.

3 PACKET STRUCTURE

All exchanged packets follow the same structure as shown below:

Start Byte	1 Byte
Command	2 Byte
Length Start Byte	1 Byte
Length	1 Byte
Payload Start Byte	1 Byte
Payload	Variable
Checksum Start Byte	1 Byte
Checksum	1 Byte

Start Byte: The start bytes are used to signalize the start of a new packet.

They are always **0x52**, **0x46**, **0x45** (ASCII: R, F, E).

Command1-2: The two command fields describe which command should be executed. These commands are described in the chapter 5.

Length & Payload: These two fields contain the payload and the length of this packet. The length indicates the count of characters in the payload field. If length is zero, the payload start byte and payload can / must be left out.

Checksum: The checksum is just a simple XOR connection of all data before.

All other start bytes are used to synchronize the protocol and to reduce the probability of the misinterpretation of the byte sequence.

Example: (Save-Settings-Permanent)

52 46 45 01 0302 02 01 03 00 04 cs

Start bytes: appear in every message, except the payload start byte (0x03)

commands: always appear in a row of two bytes

length: one byte, max length is 255

payload: includes data or parameter like modes, IDs, return results etc…

cs: checksum, xor conjunction of all preceding bytes of that messages

4 RETURN CODES

The return codes of execute functions are listed below:

RFE_SUCCESS	0x00
RFE_ERR_OP_NOT_SUPPORTED	0x50
RFE_ERR_UNKOWN_ERR	0x51
RFE_ERR_ON_EXEC_OP	0x52
RFE_ERR_COULD_NOT_WRITE	0x53
RFE_ERR_WRONG_PARAM_COUNT	0x54
RFE_ERR_WRONG_PARAM	0x55

This table should be used as an enum with the name RFE_RET_VALUE. This type is used in the following description. It is used as an unsigned char.

5 VARIABLE DICTIONARIES

5.1 Reader Control

Command Byte 1		Command Byte 2	
Reader-Common	0x01	Get-Serial Number	0x01
		Get-Reader Type	0x02
		Get-Hardware Revision	0x03
		Get-Software Revision	0x04
		Get-Bootloader Revision	0x05
Reader-RF	0x02	Get-Attenuation (dBm)	0x01
		Get-Frequency	0x02
Reader-Control	0x03	Reboot	0x01
		Reserved	0x02
		Set-Antenna-Power	0x03
		Set-Attenuation (dBm)	0x04
		Set-Frequency	0x05
		Reserved	0x20
		Reserved	0x21
		Reserved	0x30
		Reserved	0x31
Reader-Tag-Mode	0x04	Reserved	0x01
		Reserved	0x02
		Reserved	0x03

5.2 Tag Control

Command Byte 1		Command Byte 2	
Tag-Functions	0x50	Inventory-Single	0x01
		Inventory-Cyclic	0x02
		Read-From-Tag	0x03
		Write-To-Tag	0x04
		Lock-Tag	0x05

5.3 Special Command

Command Byte 1		Command Byte 2	
Tag-Functions	0x70	Special Customer Requests	0x01

5.4 Detailed Description

5.4.1 Get-Serial-Number (01-01)

This command returns the serial number of the reader.

Field-Value: String with variable length

Example:

PC -> Reader: 52 46 45 01 0101 02 00 04 cs

Reader->PC: 52 46 45 01 0101 02 04 03 03000015 04 cs

dataLength = 0x04 -> 4 Bytes

serialNumber = 0x03 0x00 0x00 0x15 -> “03-00-00-15”

5.4.2 Get-Reader-Type (01-02)

This command returns the reader type of the reader.

Field-Value: String with variable length

Example:

PC -> Reader: 52 46 45 01 0102 02 00 04 cs

Reader->PC: 52 46 45 01 0102 02 04 03 81010101 04 cs

dataLength = 0x04 -> 4 Bytes

readerType = 0x81 0x01 0x01 0x01 -> 81-01-01-01 -> PUR RM1

5.4.3 Get-Hardware-Revision (01-03)

This command returns the hardware revision of the reader. The version number is split into blocks of 4 bit. One of these blocks represents a decimal character. There are always two characters in front of the point and two after the point. So the first two bytes of the hardware revision are not used.

Field-Value: String with variable length

Example:

PC -> Reader: 52 46 45 01 0103 02 00 04 cs

Reader->PC: 52 46 45 01 0103 02 04 03 00010015 04 cs

dataLength = 0x04 -> 4 Bytes

hardwareRev = 0x00 0x01 0x00 0x15

5.4.4 Get-Software-Revision (01-04)

This command returns the software revision of the reader. The version number is split into blocks of 4 bit. One of these blocks represents a decimal character. There are always two characters in front of the point and two after the point. The first two byte of the software reversion define the application version and the second two bytes the version of the used kernel.

Field-Value: String with variable length

Example:

PC -> Reader: 52 46 45 01 0104 02 00 04 cs

Reader->PC: 52 46 45 01 0104 02 04 03 00030041 04 cs

dataLength = 0x04 -> 4 Bytes

softwareRev = 0x00 0x03 0x00 0x41

5.4.5 Get-Bootloader-Revision (01-05)

This command returns the bootloader revision of the reader. The version number is split into blocks of 4 bit. One of these blocks represents a decimal character. There are always two characters in front of the point and two after the point. So the first two bytes of the bootloader revision are not used.

Field-Value: String with variable length

Example:

PC -> Reader: 52 46 45 01 0105 02 00 04 cs

Reader->PC: 52 46 45 01 0105 02 04 03 00020005 04 cs

dataLength = 0x04 -> 4 Bytes

bootloaderRev = 0x00 0x00 0x01 0x05

5.4.6 Get-Attenuation (02-01)

This command returns the maximal potential and the current attenuation in dBm.

Parameters: none

Return Values: RFE_RET_VALUE **status**, unsigned short **maxAttenuation**, unsigned short **currentAttenuation**

Example:

PC -> Reader 52 46 45 01 0201 02 00 04 cs

Reader->PC 52 46 45 01 0201 02 07 03 00 000F 000A 04 cs

dataLength = 0x07 -> 7 Bytes

status = SUCCESS

maxAttenuation = 0x000F -> 15 dBm

currentAttenuattion = 0x000A -> 10 dBm

5.4.7 Get-Frequency (02-02)

This command returns the current frequency and the maximum count of frequencies that can be set.

enum **HOPPING_MODE** (unsigned char)	
STATIC_UP	0x00
STATIC_DOWN	0x01
RANDOM	0x02

Parameters: none

Return Values: RFE_RET_VALUE **status**, HOPPING_MODE **mode**, unsigned char **maxFrequencyCount,** unsigned char **frequencyCount**, unsigned char **frequency** [frequencyCount] [3]

Example:

PC -> Reader 52 46 45 01 0204 02 00 04 cs

Reader->PC 52 46 45 01 0204 02 0A 03 00 01 08 02 0D4094 0D3CAC 04 cs

dataLength = 0x0A -> 10 Bytes

status = 0x00 -> RFE_SUCCESS

mode = 0x01 -> STATIC_DOWN

maxFrequencyCount = 0x08 -> 8 Frequencies

frequencyCount = 0x02 -> 2 Frequencies

frequency1 = 0x0D4094 -> 868500 kHz

frequency2 = 0x0D3CAC -> 867500 kHz

5.4.8 Reboot (03-01)

This function can be used to reboot the reader.

Parameters: None

Return Values: None; the reader is rebooted immediately.

Example:

PC -> Reader 52 46 45 01 0301 02 00 04 cs

5.4.9 Set-Antenna-Power (03-03)

This function can be used to set the antenna power on and off. The values are listed below:

enum **ANTENNA_POWER** (unsigned char)	
ANTENNA_ON	0x00
ANTENNA_OFF	0x01

Parameters: AntennaPower mode

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Set antenna power off.

PC -> Reader 52 46 45 01 0303 02 01 03 00 04 cs

dataLength = 0x01 -> 1 Bytes

mode = 0x00 -> ANTENNA_OFF

Reader -> PC 52 46 45 01 0303 02 01 03 00 04 cs

dataLength = 0x01 -> 1 Bytes

status = 0x00 -> RFE_SUCCESS

5.4.10 Set-Attenuation (03-04)

This command can be used to set the attenuation of the reader in dBm. The maximal attenuation value can be found in the documentation of the reader or can be read from the reader using the "*Get-Attenuation*" command.

Parameters: unsigned short **value**

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Set output power to 10 dB

PC -> Reader 52 46 45 01 0304 02 02 03 000A 04 cs

dataLength = 0x02 -> 2 Bytes

value = 0x000A -> 10 dBm

Reader->PC 52 46 45 01 0304 02 01 03 00 04 cs

dataLength = 0x01 -> 1 Bytes

status = 0x00 -> RFE_SUCCESS

5.4.11 Set-Frequency (03-05)

This command can be used to set frequency at which the reader should operate. It is also possible to specify more than one frequency. This makes only sense with passive reader. The reader will then be hopping from one to another frequency. The mode how the reader should hop through the frequencies can also be specified. These modes are available:

enum **HOPPING_MODE** (unsigned char)	
STATIC_UP	0x00
STATIC_DOWN	0x01
RANDOM	0x02

Every frequency is transferred in three bytes. These three bytes specify the frequency in kHz.

Parameters: HOPPING_MODE **mode,** unsigned char **frequencyCount,** unsigned char **frequency** [frequencyCount] [3]

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Set output power to 10 dB

PC -> Reader 52 46 45 01 0305 02 08 03 02 02 0D4094 0D3CAC 04 cs

dataLength = 0x08 -> 8 Bytes

mode = 0x02 -> Random Hopping

frequencyCount = 0x02 -> 2 Frequencies

frequency1 = 0x0D4094 -> 868500 kHz

frequency2 = 0x0D3CAC -> 867500 kHz

Reader->PC 52 46 45 01 0305 02 01 03 00 04 cs

dataLength = 0x01 -> 1 Bytes

status = 0x00 -> RFE_SUCCESS

5.4.12 Inventory-Single (50-01)

This function can be used to make a single inventory round. The reader then returns a list of tags that are in its field. With some interfaces the data length for transmission is limited, so if the return packet would be too large, it is separated into more packets. Therefore two counters are transmitted; one that indicates how many tags were found and one that indicates how many tag ids are transmitted in this packet.

The reader can send more additional information then only the tag id. Therefore the structure TagInfo is used. This structure has a variable length that is dependent of the information sent from the reader. The type of additional information is dependent of the used reader. This information and the specific identifier can be found in the documentation of the used reader.

The first byte of the structure indicates the length of the whole structure. After this the tag id is sent. This is splitted in a start byte, a length indicator and the id itself. After that additional information can be added by the reader. In the example below, the RSSI value of the tag is added.

Length-of-TagInfo	1 Byte	
StartByte-Tag ID	1 Byte	0x01
ID-Length	1 Byte	
ID	Length	
StartByte-RSSI	1 Byte	0x02
RSSI	2 Byte	

Parameters: None

Return Values: RFE_RET_VALUE **status**, unsigned char **idCount**, unsigned char **packetIdCount**, TagInfo **tags** [packetIdCount]

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Do a single inventory. The reader sends the TagInfo with only the TagId.

PC -> Reader 52 46 45 01 5001 02 **00** 04 cs

Reader -> PC 52 46 45 01 5001 02 12 03 00 01 01 0E 01 0C 300833b23333014035050000 04 cs

dataLength = 0x12

status = 0x00 -> RFE_SUCCESS

idCount = 0x01 -> 1 ID

packetIdCount = 0x01 -> 1 ID

TagInfo 1

Length of TagInfo = 0x0E -> 14 Bytes

StartByte-TagID = 0x01

ID Length = 0x0C -> 12 Bytes

TagID = 30-08-33-b2-33-33- 01-40-35-05-00-00

5.4.13 Inventory-Cyclic (50-02)

This function can be used to start and stop a cyclic inventory. With a cyclic inventory the reader does inventories autonomous in a specified cycle. The timeout between such cycles can be found in the manual of the reader.

This function can be activated and deactivated:

enum INVENTORY_MODE (unsigned char)	
INVENTORY_OFF	0x00
INVENTORY_ON	0x01

After an activation packet was sent from the PC to the reader, the reader sends an answer with the status of the operation. Afterwards interrupt messages are sent to the PC in a defined cycle.

Parameters: InventoryMode **mode**

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Start cyclic interrupted inventory (0x01).

PC -> Reader 52 46 45 01 5002 02 01 03 **01** 04 cs

(Echo) Reader -> PC 52 46 45 01 5002 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

(TagID) Reader -> PC 52 46 45 01 9002 02 0E 03 01 0C 000000000000000000000001 04 cs

dataLength = 0x0E

StartByte-TagID = 0x01

IDLength = 0x0C

TagID = “000000000000000000000001”

5.4.14 Read-From-Tag (50-03)

With this function data can be read from the memory of a tag. The meaning of the data, that can be written, can be found in the manual of the tag. On some tags a memory bank must be specified. The memory banks are specified in the manual of the reader dependent of the type of tag.

Parameters: unsigned char **tagIdCount**, unsigned char **tagId**[tagIdCount]**,** unsigned char **memoryBank,** unsigned short **startAddress**, unsigned long **accessPassword,** unsigned char **byteCount**

Return Values: RFE_RET_VALUE **status,** unsigned char **byteCount,** unsigned char **data[byteCount]**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Write 5 byte to the tag 30-00-30-08-33-b2-dd-d9-01-40-35-05-00-00 at the memory bank 1 and the start address 0x12:

PC -> Reader 52 46 45 01 5003 02 16 03 0C 300833b2ddd9014035050000 01 0000 00000000 05 04 cs

dataLength = 0x16

tagIdCount = 0x0C -> 12 Bytes

tagId = 30-08-33-b2-33-33-01-40-35-05-00-00

memoryBank = 0x01 -> second bank

startAddress = 0x0000

accessPassword = 0x00000000

bytesCount = 0x05 -> 5 Bytes

Reader -> PC 52 46 45 01 5003 02 01 03 00 05 0223A4884C 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

bytesCount = 0x05 -> 5 Bytes

data = 0x0223A4884C

5.4.15 Write-To-Tag (50-04)

With this function data can be written to the memory of a tag. The meaning of the data, that can be written, can be found in the manual of the tag. On some tags a memory bank must be specified. The memory banks are specified in the manual of the reader dependent of the type of tag.

Parameters: unsigned char **tagIdCount**, unsigned char **tagId**[tagIdCount]**,** unsigned char **memoryBank,** unsigned short **startAddress**, unsigned long **accessPassword,** unsigned char **byteCount**, unsigned char **data** [byteCount]

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Write 5 byte to the tag 30-00-30-08-33-b2-dd-d9-01-40-35-05-00-00 at the memory bank 1 and the start address 0x12:

PC -> Reader 52 46 45 01 5004 02 16 03 0C 300833b2ddd9014035050000 01 0000 00000000 05 0223A4884C 04 cs

dataLength = 0x16

tagIdCount = 0x0C -> 12 Bytes

tagId = 30-08-33-b2-33-33-01-40-35-05-00-00

memoryBank = 0x01 -> first bank

startAddress = 0x0000

accessPassword = 0x00000000

bytesCount = 0x05 -> 5 Bytes

data = 0x0223A4884C

Reader -> PC 52 46 45 01 5004 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

5.4.16 Lock-Tag (50-05)

With this function the tag can be locked. The meaning of the mode and the memory space can be found in the documentation of the reader.

Parameters: unsigned char **tagIdCount**, unsigned char **tagId**[tagIdCount]**,** unsigned char **mode,** unsigned char **mem_space**, unsigned long **accessPassword**

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Lock the tag 30-00-30-08-33-b2-dd-d9-01-40-35-05-00-00 with the mode 1 at the memory space 1:

PC -> Reader 52 46 45 01 5005 02 13 03 0C 300833b2ddd9014035050000 01 01 00000000 04 cs

dataLength = 0x13

tagIdCount = 0x0C -> 12 Bytes

tagId = 30-08-33-b2-33-33-01-40-35-05-00-00 mode = 0x01

mem_space = 0x01

accessPassword = 0x00000000

Reader -> PC 52 46 45 01 5005 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

5.4.17 Special Customer Requests (70-01)

5.4.17.1 Read and Clear (payload mode 00 and 01)

With this function the tag can be read and immediately over-written by “00 00 00 00 00 00 00 00 00 00 00 00”.

Parameters: unsigned char **mode**, unsigned long **accessPasswordReturn**

Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Start inventory

PC -> Reader 52 46 45 01 7001 02 05 03 01 00000000 04 cs
dataLength = 0x05

inventoryMode = INVENTORY_ON

accessPassword = 0x00000000

Example: Stop inventory

PC -> Reader 52 46 45 01 7001 02 01 03 00 04 cs

dataLength = 0x01

inventoryMode = INVENTORY_OFF

Reader -> PC 52 46 45 01 7001 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

(TagID) Reader -> PC 52 46 45 01 9002 02 0E 03 01 0C 000000000000000000000001 04 cs

dataLength = 0x0E

StartByte-TagID = 0x01

IDLength = 0x0C

TagID = “000000000000000000000001”

5.4.17.2 Write 12 bytes EPC with Password (payload mode 00 and 01)

With this function the tag can be written 12bytes EPC number which was protected by 4 bytes Access Code.

Parameters: unsigned char **mode**, unsigned long **accessPassword**, unsigned char **tagId**[tagIdCount]

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Start writing

PC -> Reader 52 46 45 01 7001 02 10 03 01 11111111 112233445566778899 00AABB 04 cs

dataLength = 0x10

inventoryMode = INVENTORY_ON

accessPassword = 0x11111111

12bytes EPC = 0x112233445566778899 00AABB

Example: Stop writing

PC -> Reader 52 46 45 01 7001 02 01 03 00 04 cs

dataLength = 0x01

inventoryMode = INVENTORY_OFF

Reader -> PC 52 46 45 01 7001 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

(Success) Reader -> PC 52 46 45 01 9002 02 0E 03 01 0C 112233445566778899 00AABB 04 cs

dataLength = 0x0E

StartByte-TagID = 0x01

IDLength = 0x0C

TagID = “112233445566778899 00AABB”

5.4.17.2 Set Buzzer Status (payload mode 02)

With this function the reader can enable and disable the buzzer.

Parameters: unsigned char **mode**, unsigned char **buzzer status**,

Return Values: RFE_RET_VALUE **status**

Status Values: RFE_SUCCESS, RFE_ERR_ON_EXEC_OP, RFE_ERR_COULD_NOT_WRITE, RFE_ERR_WRONG_PARAM_COUNT, RFE_ERR_WRONG_PARAM

Example: Enable Buzzer

PC -> Reader 52 46 45 01 7001 02 02 03 02 01 04 cs

dataLength = 0x02

inventoryMode = Buzzer

Status = 0x01 Enable

Example: Disable Buzzer

PC -> Reader 52 46 45 01 7001 02 02 03 02 00 04 cs

dataLength = 0x02

inventoryMode = Buzzer

Status = 0x00 Disable

Reader -> PC 52 46 45 01 7001 02 01 03 00 04 cs

dataLength = 0x01

status = 0x00 -> RFE_SUCCESS

5.5 Data Structures

The used data structures are collected in this chapter and shown in C syntax:

```
enum RFE_RET_VALUE
{
RFE_SUCCESS = 0x00,
RFE_ERR_OP_NOT_SUPPORTED = 0x50,
RFE_ERR_UNKOWN_ERR = 0x51,
RFE_ERR_ON_EXEC_OP = 0x52,
RFE_ERR_COULD_NOT_WRITE = 0x53,
RFE_ERR_WRONG_PARAM_COUNT = 0x54,
RFE_ERR_WRONG_PARAM = 0x55
};
enum HEARTBEAT_SIGNAL
{
HEARTBEAT_ON = 0x00,
HEARTBEAT_OFF = 0x01
};
enum ANTENNA_POWER
{
ANTENNA_ON = 0x00,
ANTENNA_OFF = 0x01
};
enum INVENTORY_MODE
{
INVENTORY_OFF = 0x00,
INVENTORY_ON = 0x01
};
```

6 Using ST110 Reader Tool

Below is a general overview of ST110 Reader Tool with instructions for installing and operating the ST110 Reader Tool. This tool is provided with no implied warranties.

6.1 Driver Installation

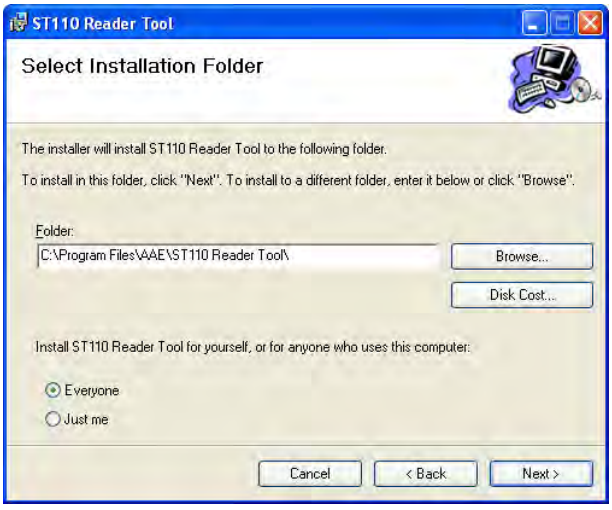
The ST110 UHF RFID READER uses USB port which provides a virtual COM port to communicate with a PC. You can install USB driver provided in the ST110 support CD.

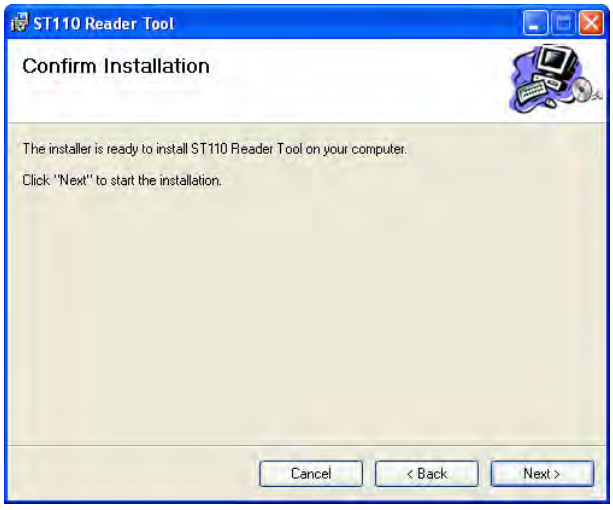
6.2 Microsoft Dot Net Framework Installation

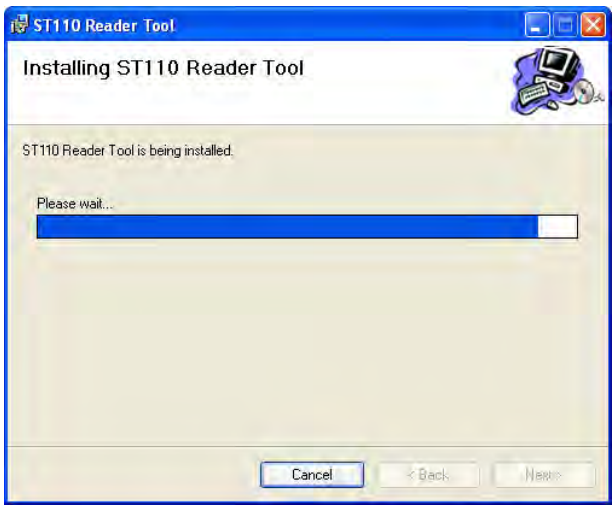
ST110 Reader Tool runs under Microsoft Dot Net Framework 3.5 environment. Microsoft Dot Net Framework 3.5 provided in the support CD You can install from the “X:\ ST110 ST510\PC SOFTWARE\DotNetFX3.5” path in the support CD.

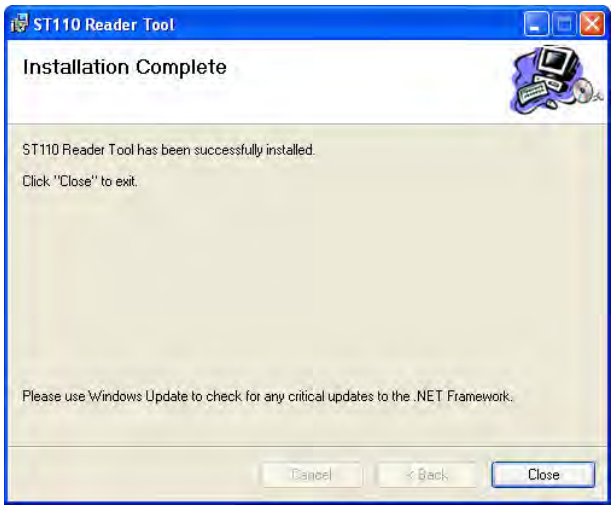
6.3 ST110 Reader Tool Installation

Now you can install ST110 Reader Tool from “X:\ST110 ST510\PC SOFTWARE\ST110 Reader Tool\ST110\Reader Tool.msi” path in the support CD.







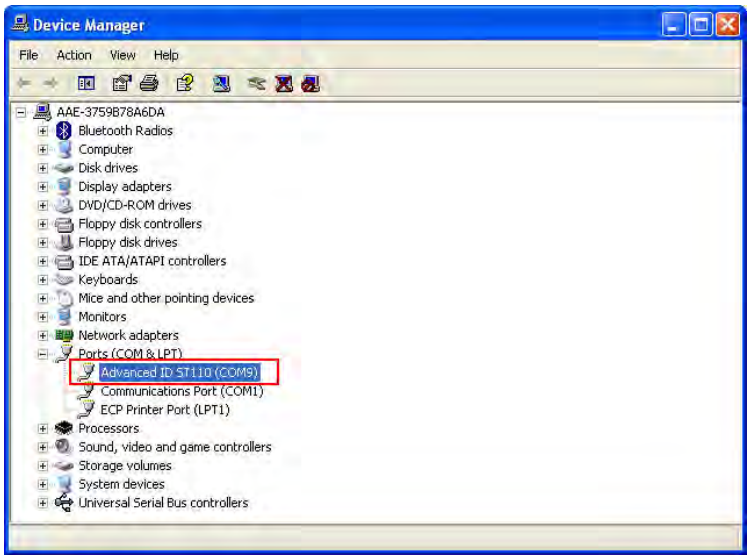


Now you can start ST110 Reader Tool by double click a shortcut icon on Desktop.

6.4 Software Instruction

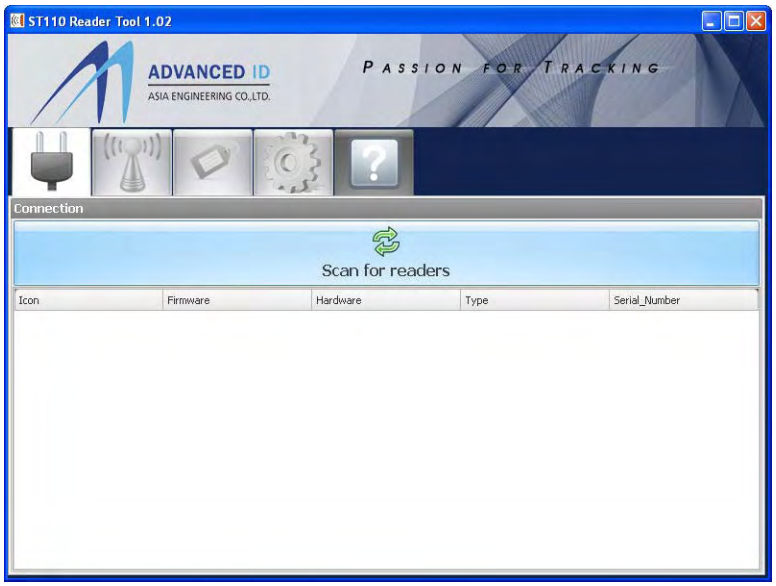
6.4.1 Connect the ST110 READER to the PC

If you install the USB driver correctly you will find the hardware in Device Manager as below:



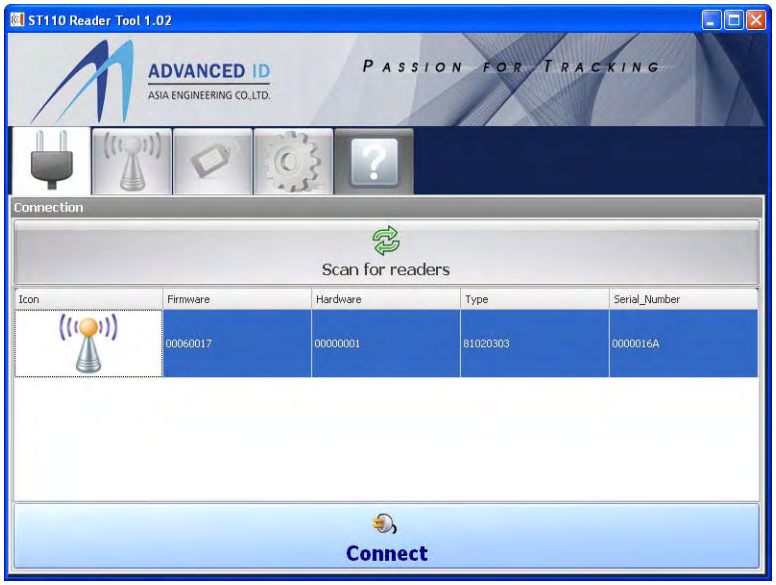
6.4.2 Detect RFID Readers

To detect RFID Readers, after the reader is plugged to PC, click Scan for readers



Picture 6.4.1 Scan reader

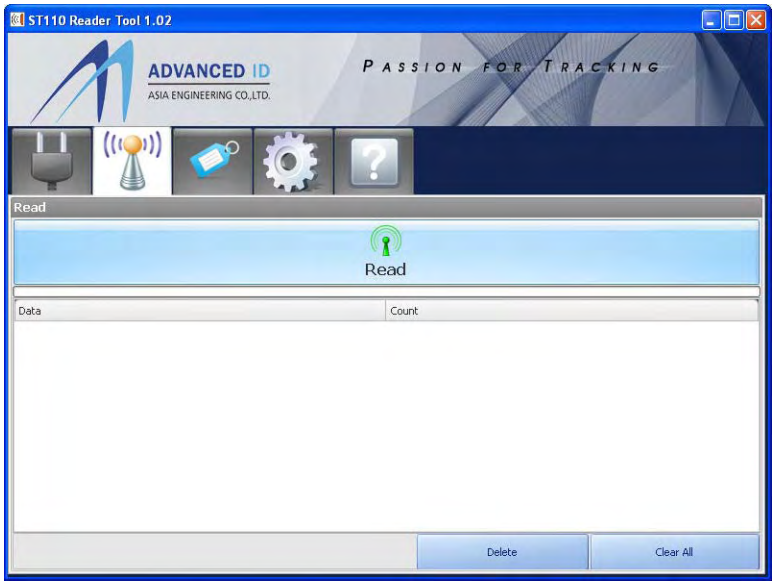
Then click 'Connect' button to start communicate with the reader.



Picture 6.4.2 Reader detected and connect a reader

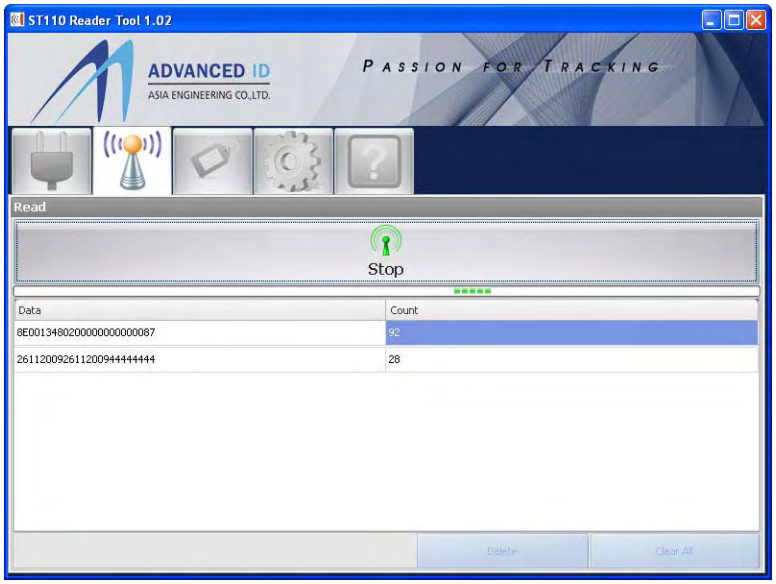
6.4.3 Start/Stop reading data in EPC bank from RFID tags

After the reader is connected, the tab will be automatically switched to ‘Read’ tab. To start inventory-cyclic reading tags, click ‘Read’ button then put RFID tags on the reader range.



Picture 6.4.3 Read tab

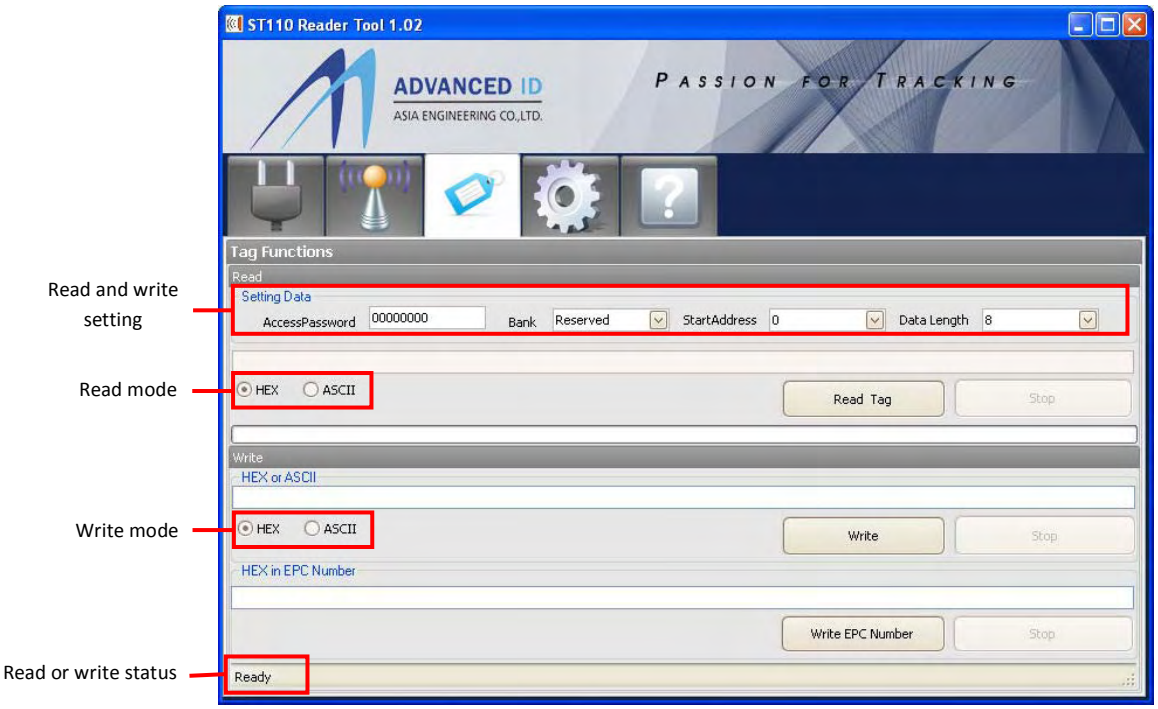
To stop inventory-cyclic reading data, click ‘Stop’ button.



Picture 6.4.4 Reader is reading tags from EPC bank

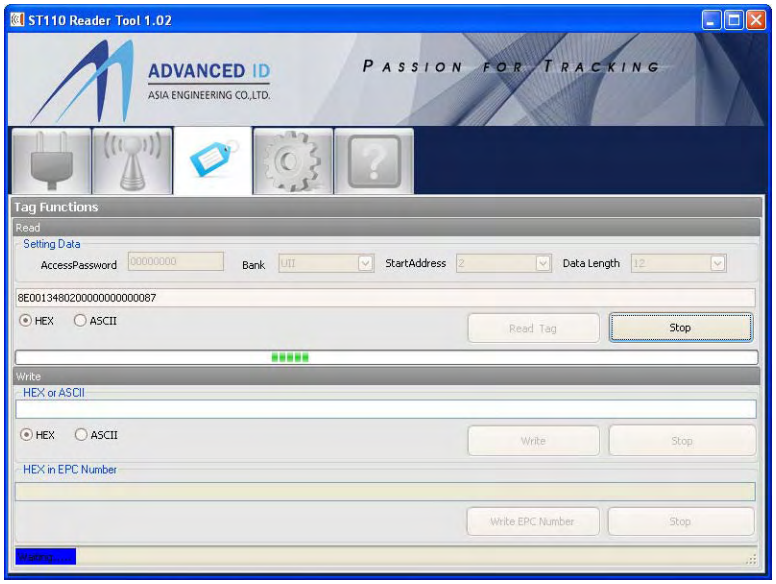
6.4.4 Read/Write data to a tag form Tag functions tab

In the 'Tag Functions' tab, there are functions to read and write a single tag data. User is able to read or write data in each bank of the RFID tag (Reserved, UII/EPC, TID and User)



To read HEX from a single tag

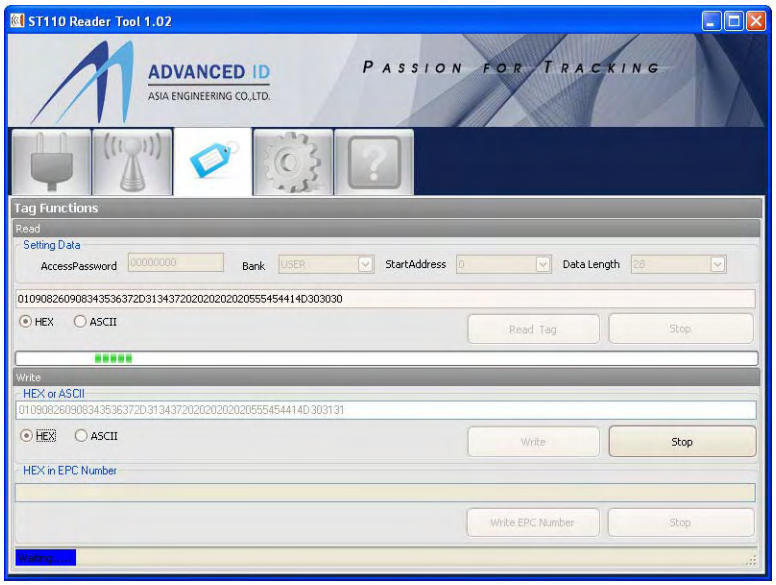
Click 'Read Tag' button, then put a tag in reader range. Read function will be stop after finish reading a tag or you can click 'Stop' button to stop reading.



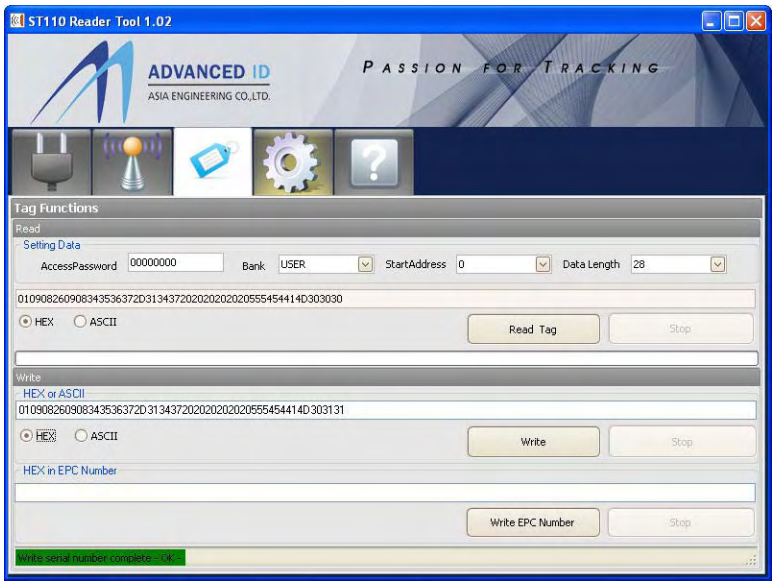
Picture 6.4.5 Reader try to read a tag.

To write HEX to a single tag

Enter heximal data in text box. Click 'Write' button, then put a tag in reader range. Write function will be stop after finish writing a tag or you can click 'Stop' button to stop writing.



Picture 6.4.6 Reader try to write data to a tag



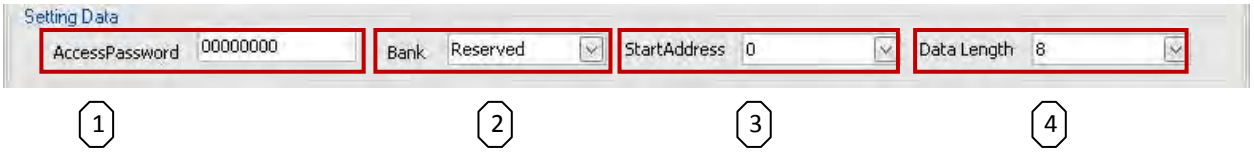
Picture 6.4.7 Finish writing a tag

To write HEX data in EPC bank

Enter heximal data 12 bytes in the HEX in EPC Number text box. Click 'Write EPC Number' button then put a tag in reader range. Write EPC Number function will be stop after finish writing a tag or you can click 'Stop' button to stop writing.

Note: Read and write function is set to HEX mode by default. To read or write in ASCII mode, select 'ASCII' radio button before click Read or Write tag

Read and write functions Setting



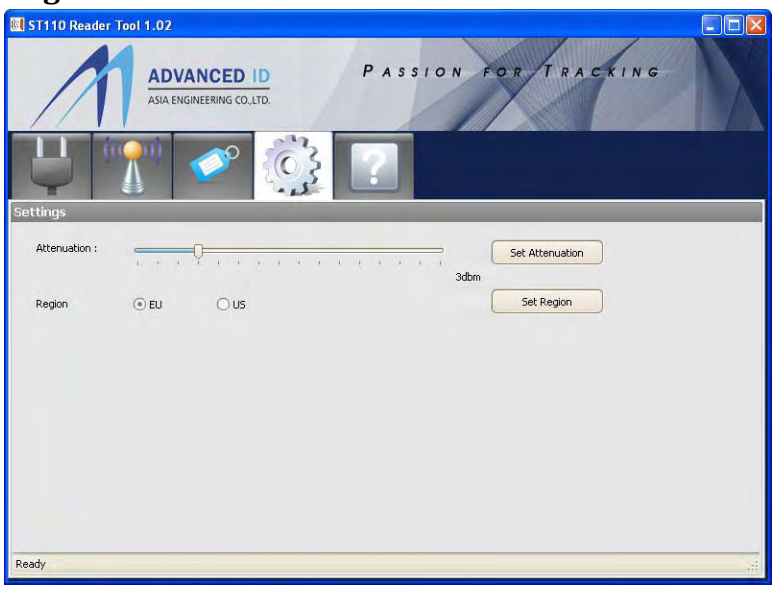
1. AccessPassword : Set read/write access password in 4 bytes

2. Bank : Set read/write from Reserved, UII, TID or User bank in RFID tag (Gen2)

3. StartAddress : Set read/write start address

4. Data Length: Set read/write data lengths

6.4.5 Reader Setting



Reader Attenuation Setting

This function uses for setting the attenuation power of the reader. To set to long read range change setting to minimum attenuation, 0 dbm, and to set to short read range change setting to maximum attenuation, 15 dbm.

Frequency Region Setting

There are 2 regions setting for ST110, frequencies for EU and US. For US, Frequencies range is 902.5 MHz plus by 0.5 MHz until 927 MHz (50 frequencies) and frequencies for EU are 867.5 MHz and 868.5 MHz.

FCC Statement

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 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.

To assure continued compliance, any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment. (Example - use only shielded interface cables when connecting to computer or peripheral devices).

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and
(2) This device must accept any interference received, including interference that may cause undesired operation.

FCC Radiation Exposure Statement:

(if mobile device (generally 20 cm distance) is applicable, MPE calculation)

“This equipment complies with FCC RF radiation exposure limits set forth for an uncontrolled environment. The antenna(s) used for this transmitter must be installed to provide a separation distance of at least 20 cm from all persons and must not be co-located or operating in conjunction with any other antenna or transmitter”.

Declaration of Conformity for USA

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

(1) This device may not cause harmful interference, and
(2) This device must accept any interference received, including interference that may cause undesired operation.

Usually this is followed by the following FCC caution:

Any changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate this equipment.

For **Canada,** Operation is subject to the following two conditions:

1) This device may not cause interference and
2) This device must accept any interference, including interference that may cause undesired operation of the device

If the antenna is detachable the following sentences according to RSS-210 clause 5.5/5.11 need to be included:

“This device has been designed to operate with an antenna having a maximum gain of .. dB. Antenna having a higher gain is strictly prohibited per regulations of Industry Canada. The required antenna impedance is 50 ohms”.

“To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the EIRP is not more than required for successful communication”.