GO2NFC141U NFC Reader

Mode: G141U

Overview

The GO2NFC141U NFC Reader is contactless smart card reader/writer developed based on the 13.56 MHz Contactless (RFID) Technology. Unlike other NFC readers/writers, GO2NFC141U is very compact and small.

GO2NFC141U is a plug-and-play USB device. It is compatible with different devices and applications. Compliant with the ISO/IEC18092 standard for Near Field Communication (NFC), it supports not only Mifare and ISO 14443 A and B cards but also all four types of NFC tags. It also could pushes or pulls data with Android NFC phone. Although in a small size, the proximity operating distance of GO2NFC141U is 5 cm. With an access speed of up to 424 kbps and a full USB speed of up to 12 Mbps, GO2NFC141U can also read and write quickly and efficiently, depending on the type of contactless tag in use.

Software for GO2NFC141U is available for both Windows OS and Linux. Most important, the software is open source, totally free for any changing and usage. The software contains operation function such as reading/writing card, pushing URL to Android phone, pulling message from Android phone.

The GO2NFC141U NFC Reader is ideal for access control, e-payment, e-ticketing for events and mass transit, toll road fare collection and network authentication.

Feature

GO2NFC141U Standard Features:

- USB 2.0 Full Speed Interface
- Smart Card Reader:

  Read/write speed up to 424 kbps

  Built-in antenna for contactless tag access, with card reading distance of around 50 mm (depending on tag type)

  Supports ISO 14443 Type A and B cards, Mifare, FeliCa, and all 4 types of NFC (ISO/IEC 18092) tags

  Supports pushing/pulling data with Android NFC phones
- Very small size
- User-controllable buzzer

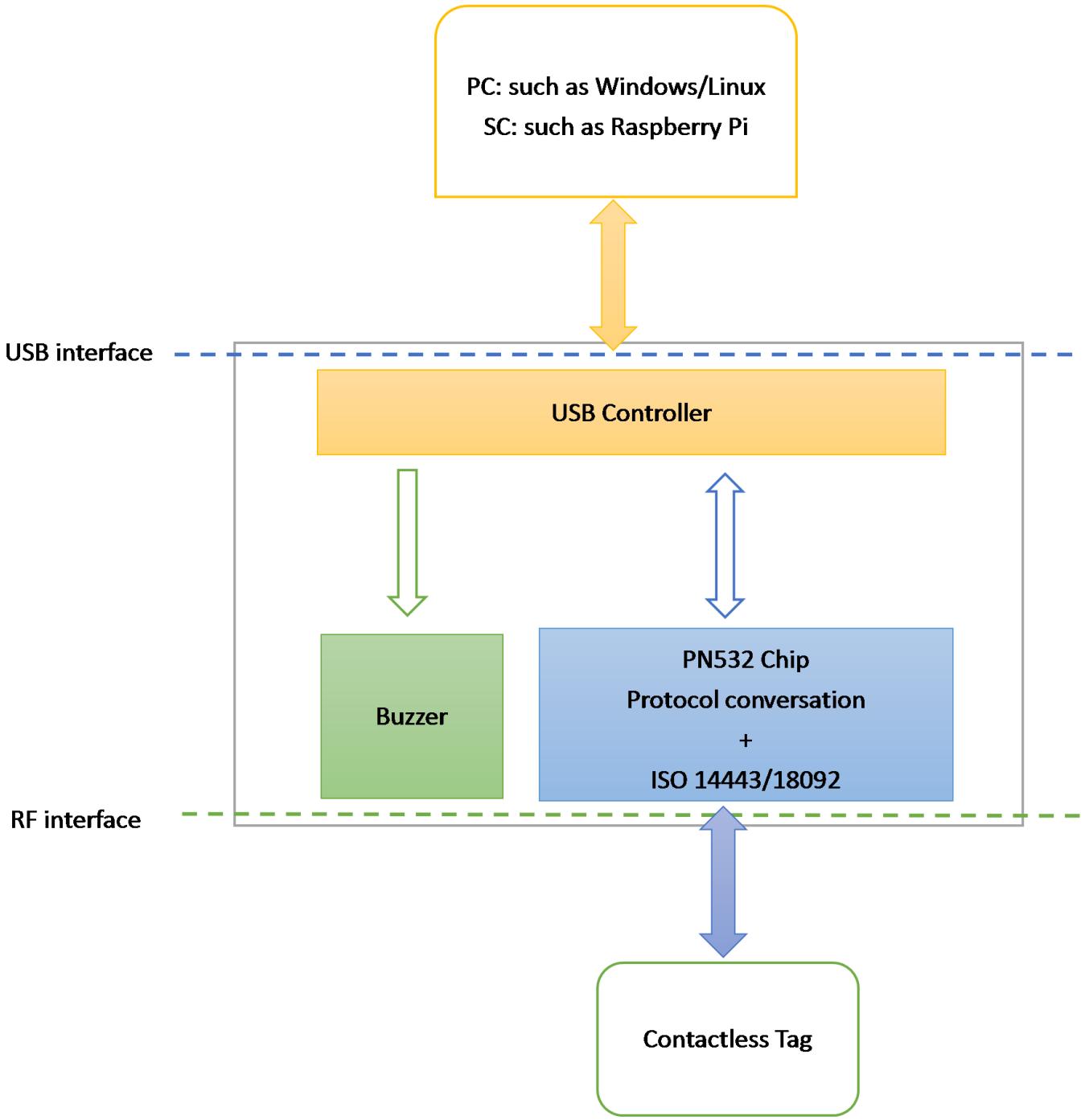
Typical Applications

- e-Government
- Banking & Payment
- Network Security
- Access Control
- e-Purse & Loyalty
- Transportation

Technical Specifications

Dimensions	46.0mm (L) * 48.5mm (W) * 10.5mm (H)
Weight	20 g (without USB cable)
Interface	Micro USB socket
Operating Distance	Around 50 mm (depends on the tag type)
Supply Voltage	Regulated 5V DC
Supply Current	100mA (Maximum); 15mA (standby); 40mA (normal)
Operating Temperature	0-50 °C
Operating Frequency	13.56 MHz
Smart Card Interface Support	• ISO14443 Type A & B • Mifare • FeliCa • 4 types of NFC (ISO/IEC18092) tags
Operating System Support	• Win XP, Win Vista, Win 7, Win 8, Win XP x64, Win Vista x64, Win 7 x64, Win 8 x 64 • Linux

Architecture



Software

Users could download the software through the following link:

http://www.go2nfc.com/download/

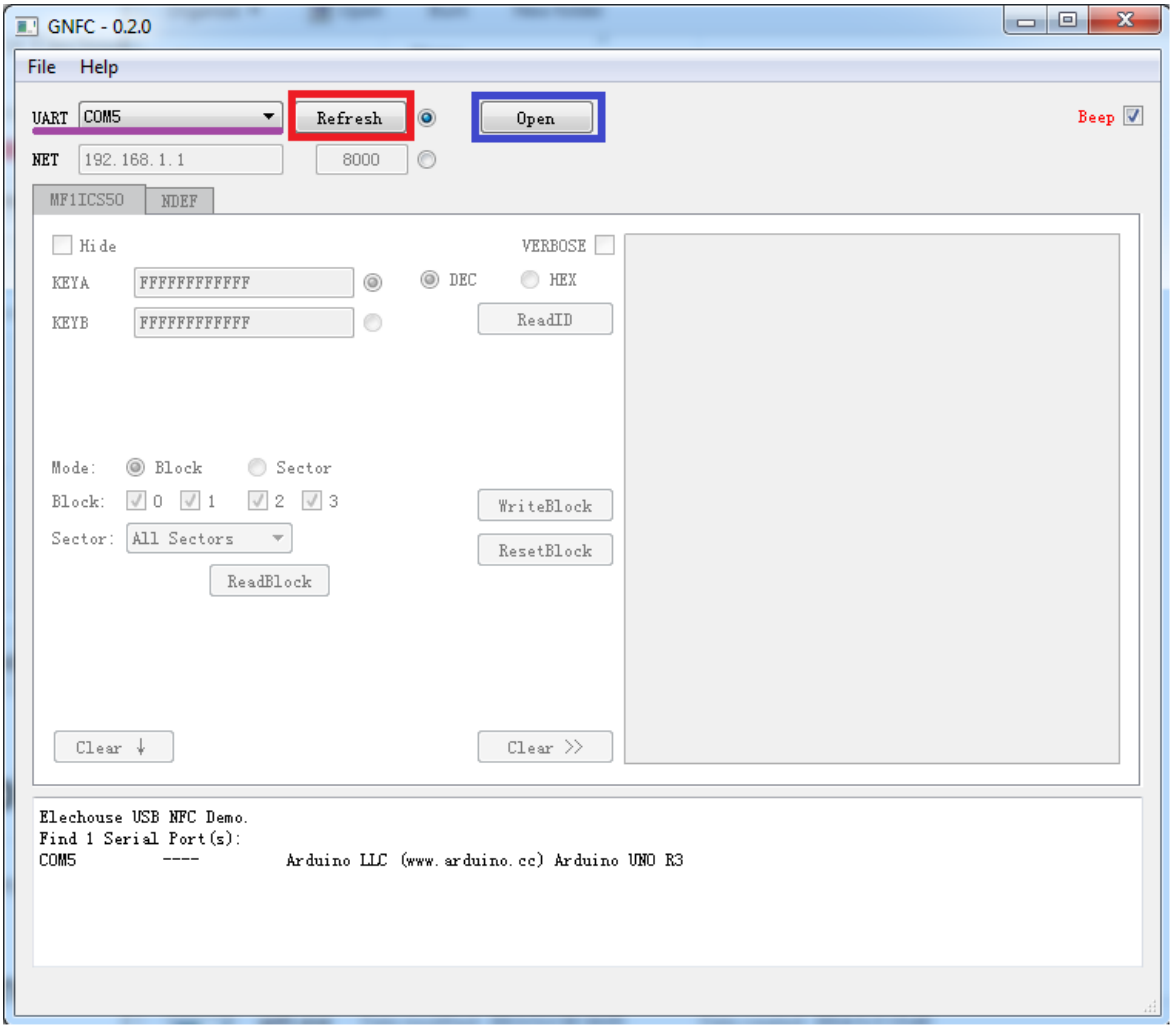
Mobile phone users could scan the following code.

Operation Guide

Windows

Connect GO2NFC device with your computer with micro USB cable. Install the driver from “Device manager”. You could find the “**DRIVER**” folder which contains the driver in the software files. If the driver was installed well, you could find a COM port for this device.

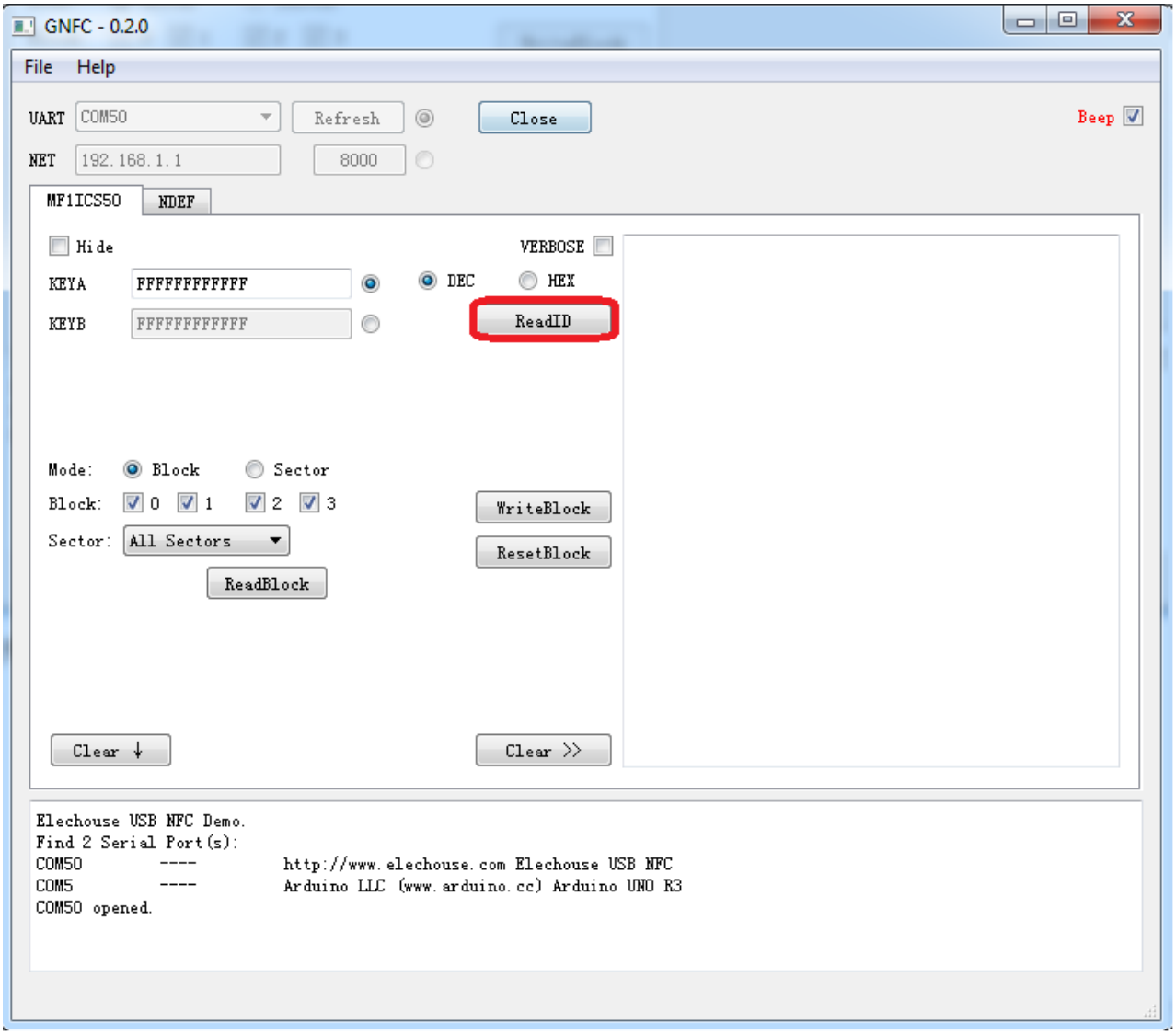
Start the software **“gnfc.exe”**



Click the button “**Refresh**”. The software will find the COM ports available. Choose the right COM port and then click “**Open**”. Information will be shown at the bottom.

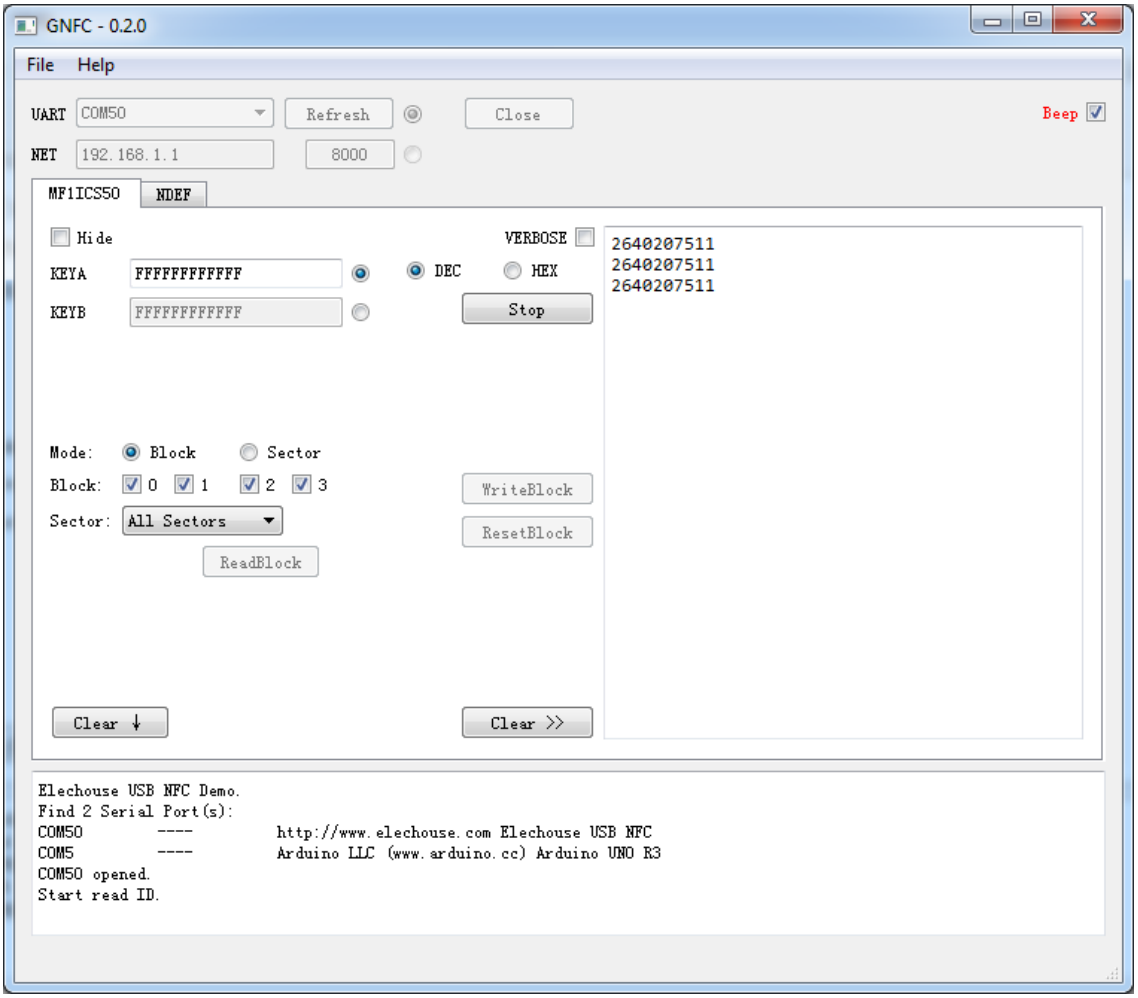
Read card information

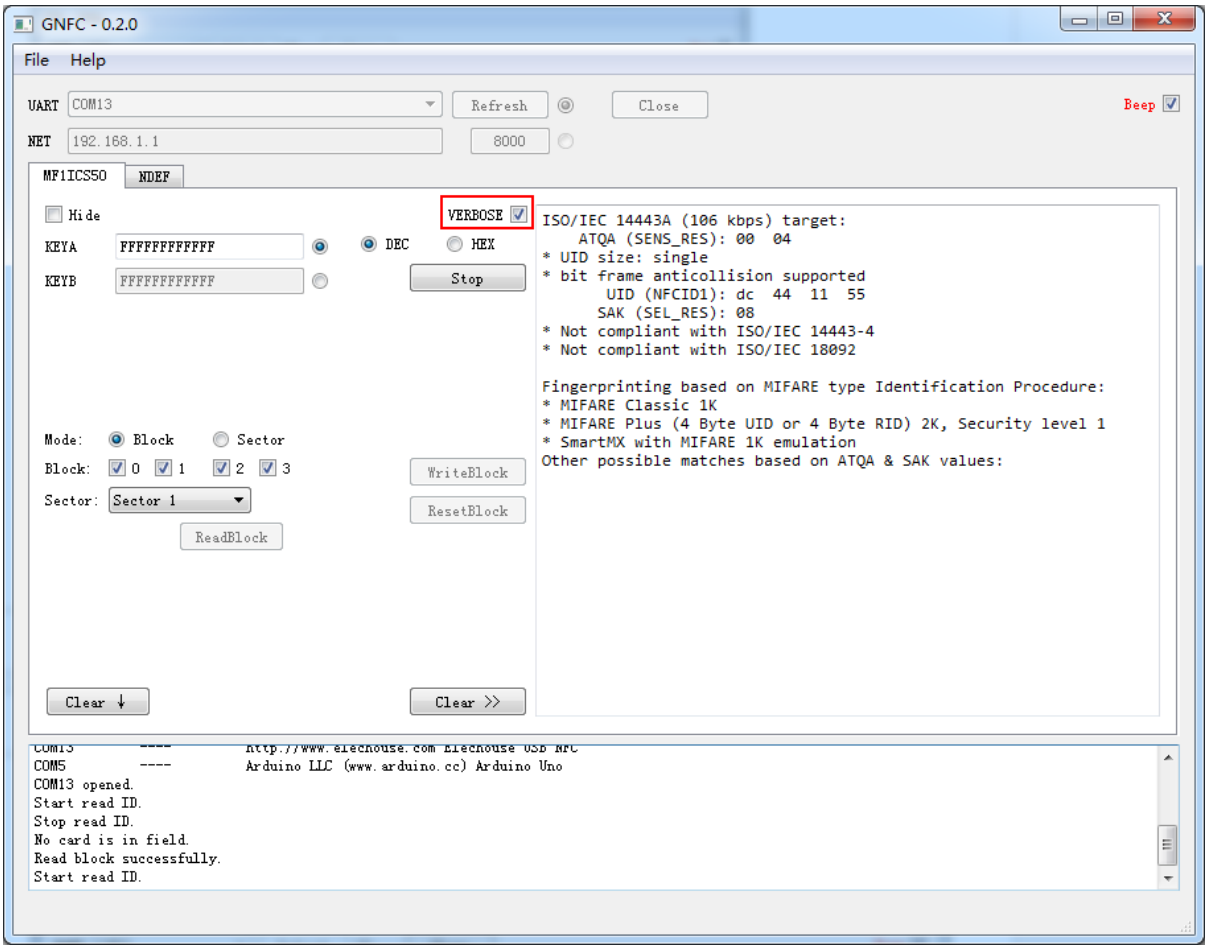
Click the button “**ReadID**”.



Put a valid card on the device. You could get the card ID.

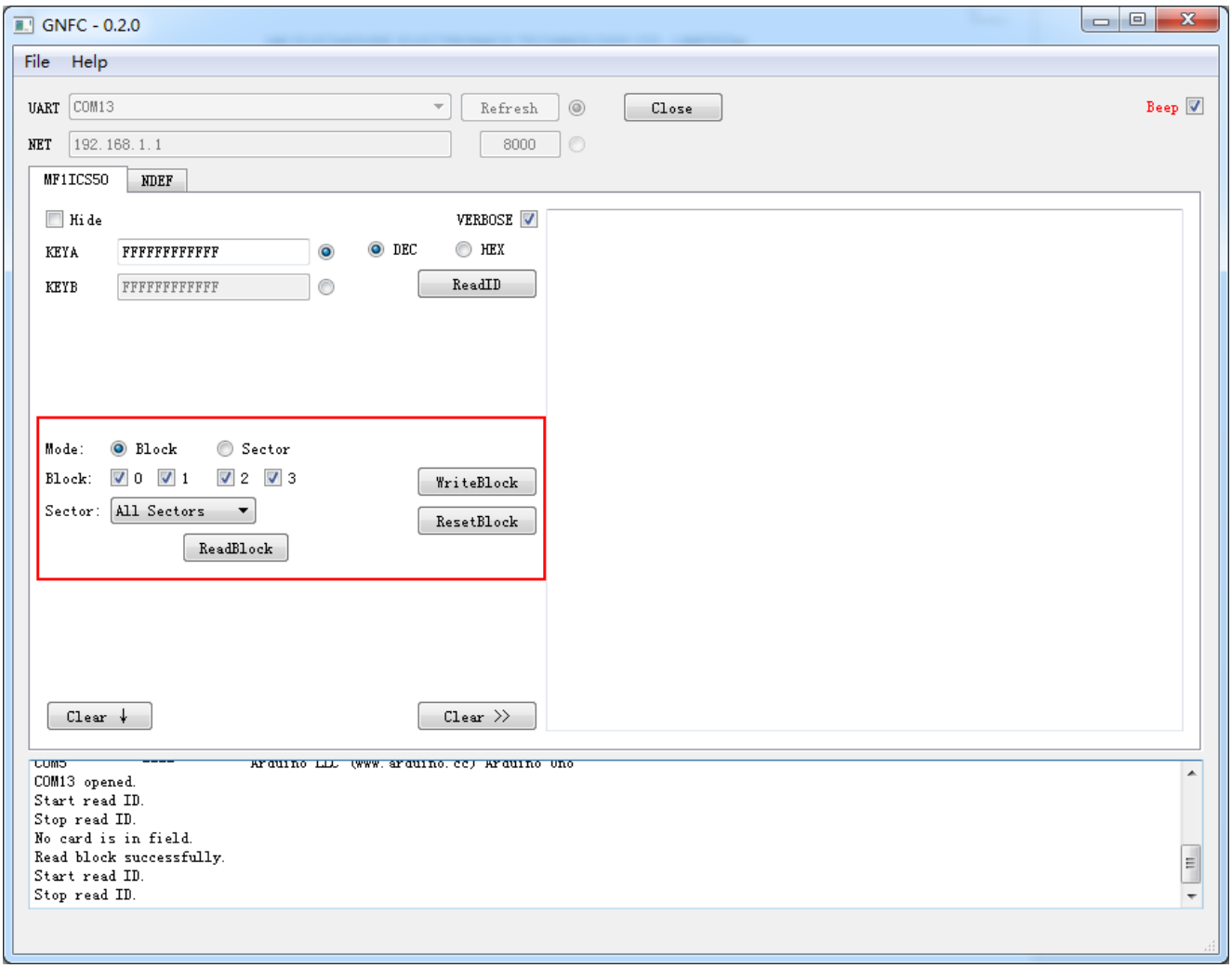
Check the “**VERBOSE”.** It could read back more information.


GNFC - 0.2.0
File Help
UART COM50 Refresh Close Beep
NET 192.168.1.1 8000
MF1ICS50 NDEF
Hide VERBOSE
KEYA FFFFFFFFFFFF DEC HEX
KEYB FFFFFFFFFFFF Stop
2640207511
2640207511
2640207511
Mode: Block Sector
Block: 0 1 2 3 WriteBlock
Sector: All Sectors ResetBlock
ReadBlock
Clear Clear >>
Elechouse USB NFC Demo.
Find 2 Serial Port(s):
COM50 ---- http://www.elechouse.com Elechouse USB NFC
COM5 ---- Arduino LLC (www.arduino.cc) Arduino UNO R3
COM50 opened.
Start read ID.


GNFC - 0.2.0
File Help
UART COM13 Refresh Close Beep
NET 192.168.1.1 8000
MF1ICS50 NDEF
Hide VERBOSE
KEYA FFFFFFFFFFFF DEC HEX
KEYB FFFFFFFFFFFF Stop
ISO/IEC 14443A (106 kbps) target:
ATQA (SENS_RES): 00 04
* UID size: single
* bit frame anticollision supported
UID (NFCID1): dc 44 11 55
SAK (SEL_RES): 08
* Not compliant with ISO/IEC 14443-4
* Not compliant with ISO/IEC 18092
Fingerprinting based on MIFARE type Identification Procedure:
* MIFARE Classic 1K
* MIFARE Plus (4 Byte UID or 4 Byte RID) 2K, Security level 1
* SmartMX with MIFARE 1K emulation
Other possible matches based on ATQA & SAK values:
Mode: Block Sector
Block: 0 1 2 3 WriteBlock
Sector: Sector 1 ResetBlock
ReadBlock
Clear Clear >>
COM5 ---- Arduino LLC (www.arduino.cc) Arduino Uno
COM13 opened.
Start read ID.
Stop read ID.
No card is in field.
Read block successfully.
Start read ID.

Read/Write the data stored in the card

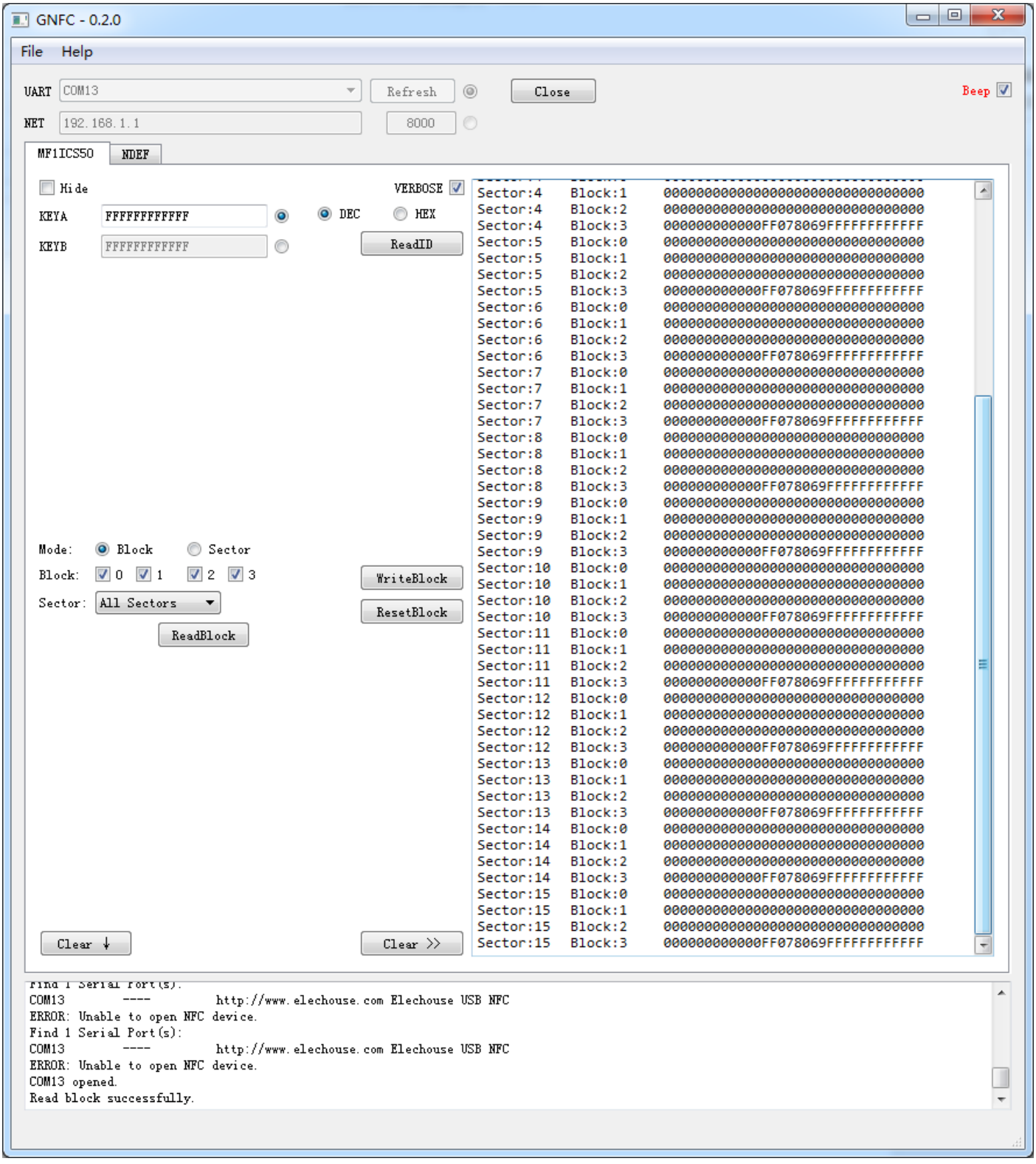
The following buttons controls data reading or writing



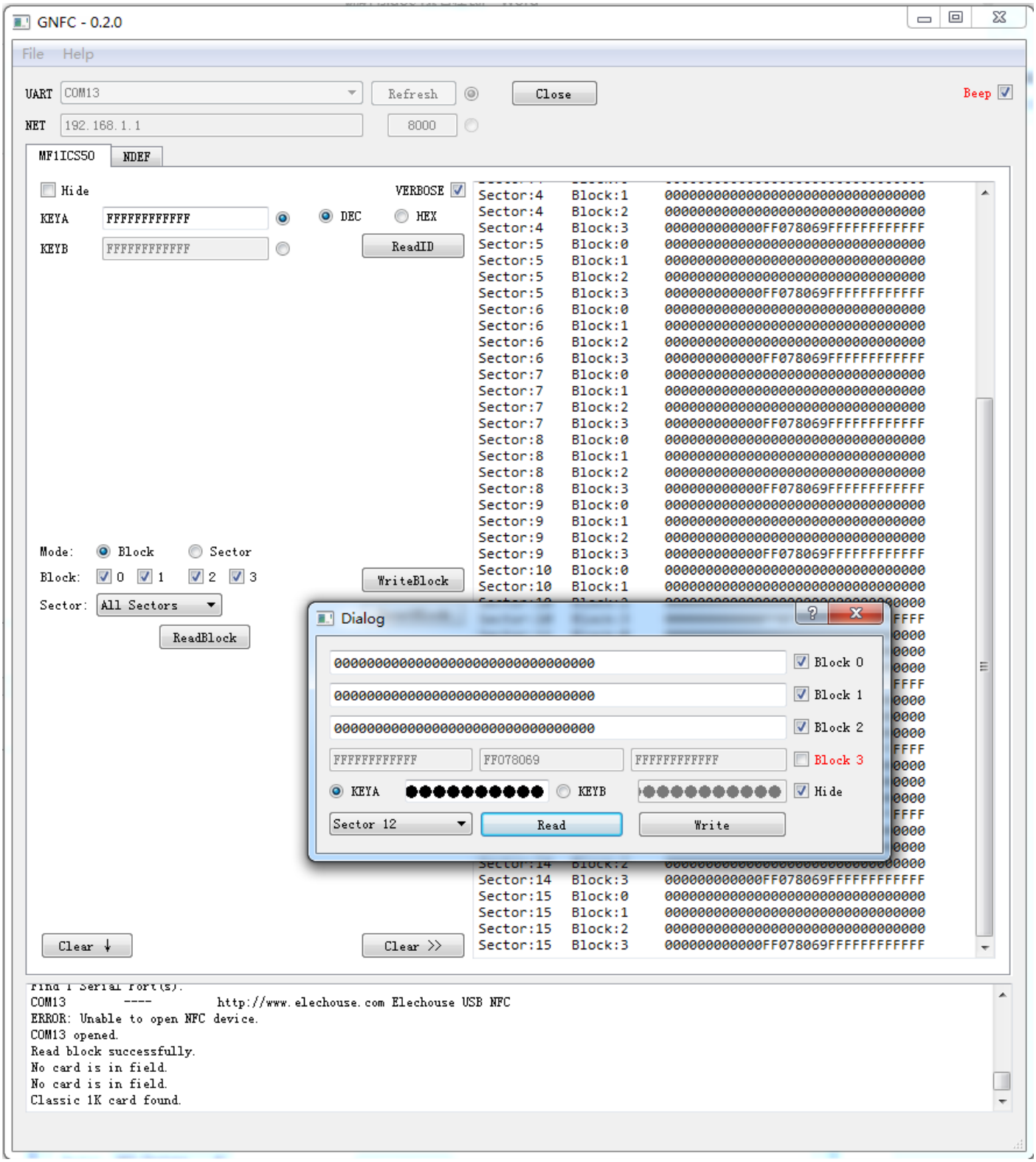
To write/read data into the card, you need to read the data structure of the card.

Choose the Block/Sector you want to operate.

Put a valid card on the device. Then click the button “**ReadBlock**”.



Click the button “**WriteBlock**” and pop up new dialog



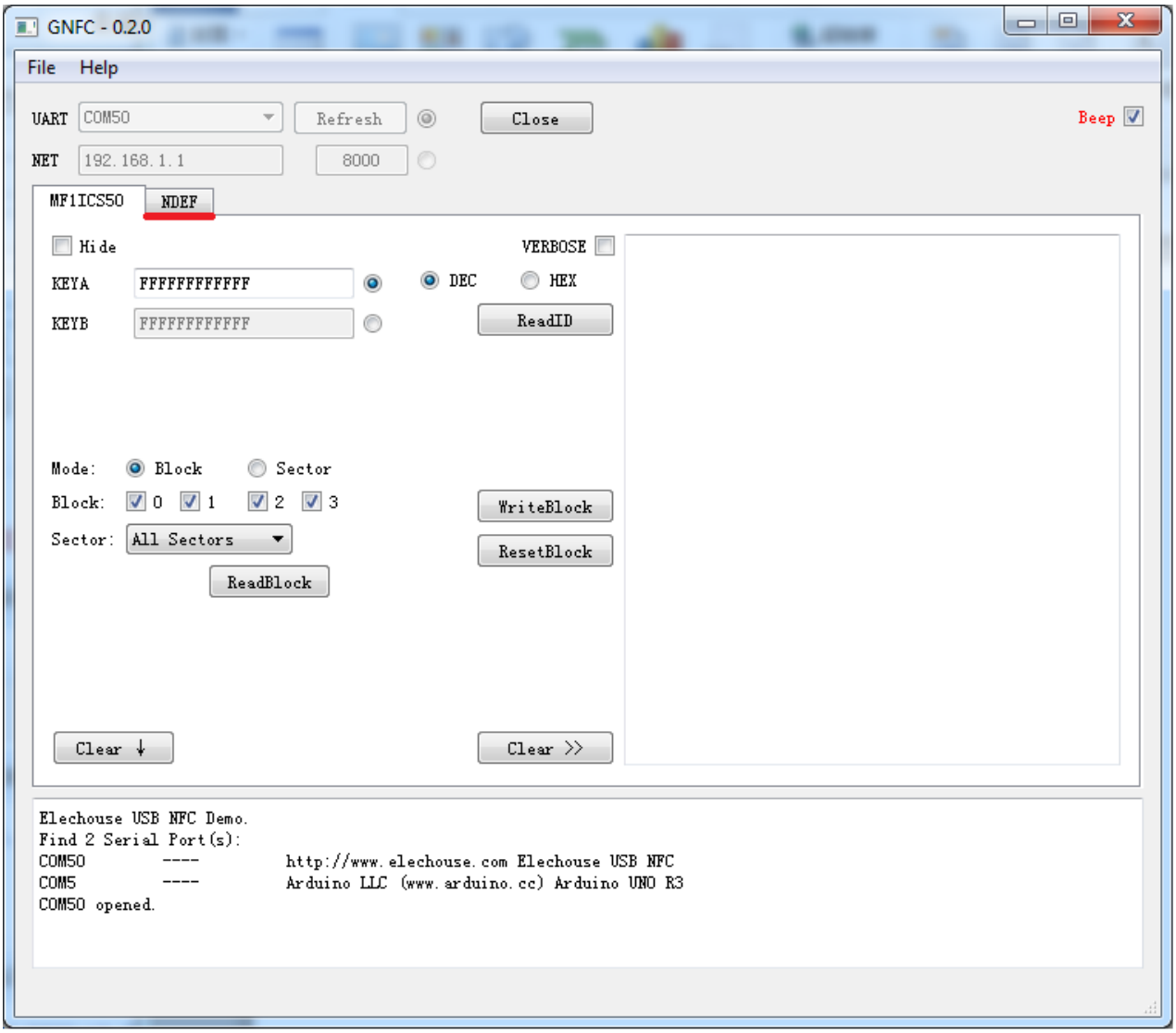
In the new dialog, users could read or write data.

Block 3 is usually used to store password. Becareful if you want to write it.

NFC with Phone

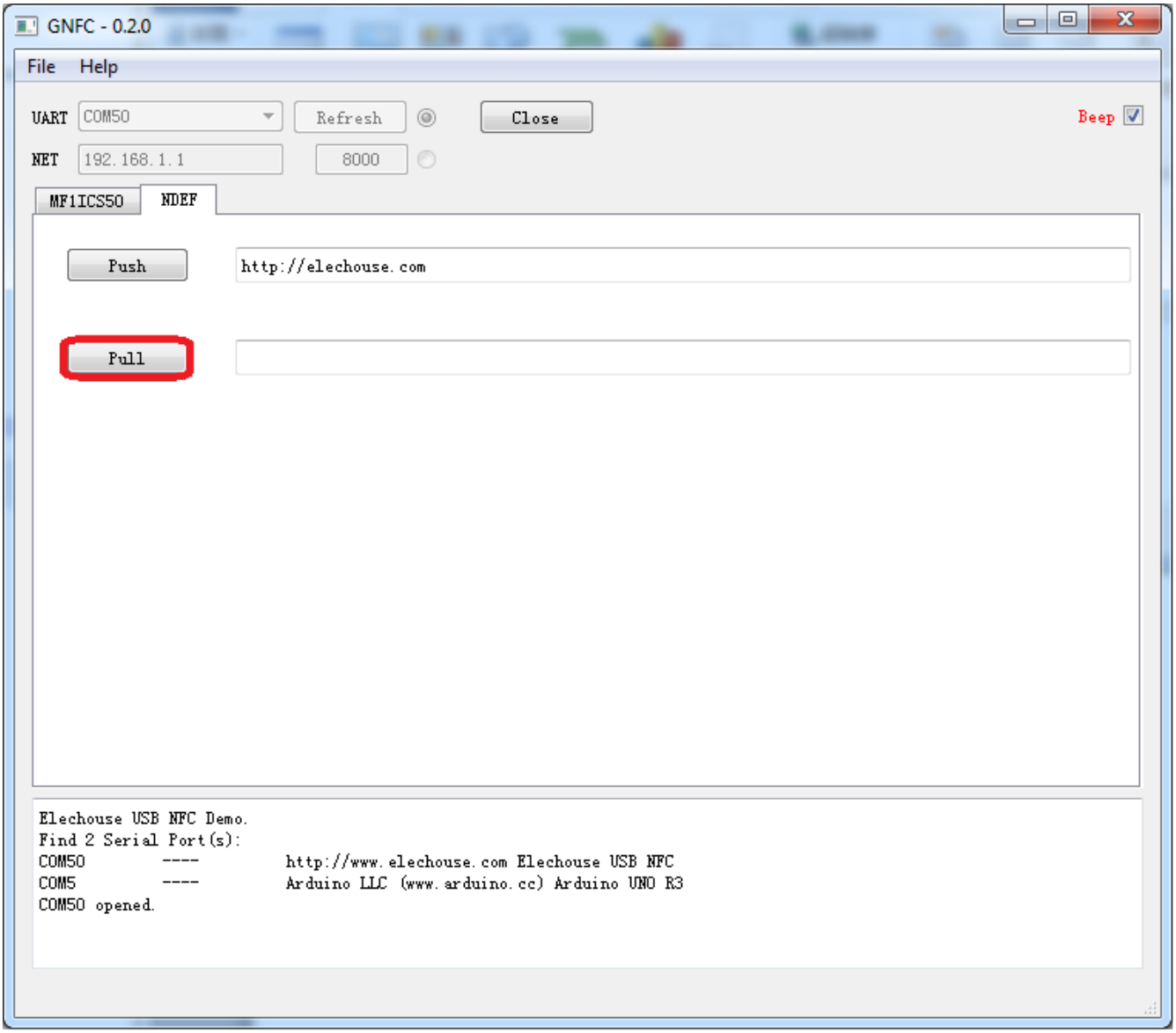
The software also contains function to push/pull URL with phone.

Click the NDEF card



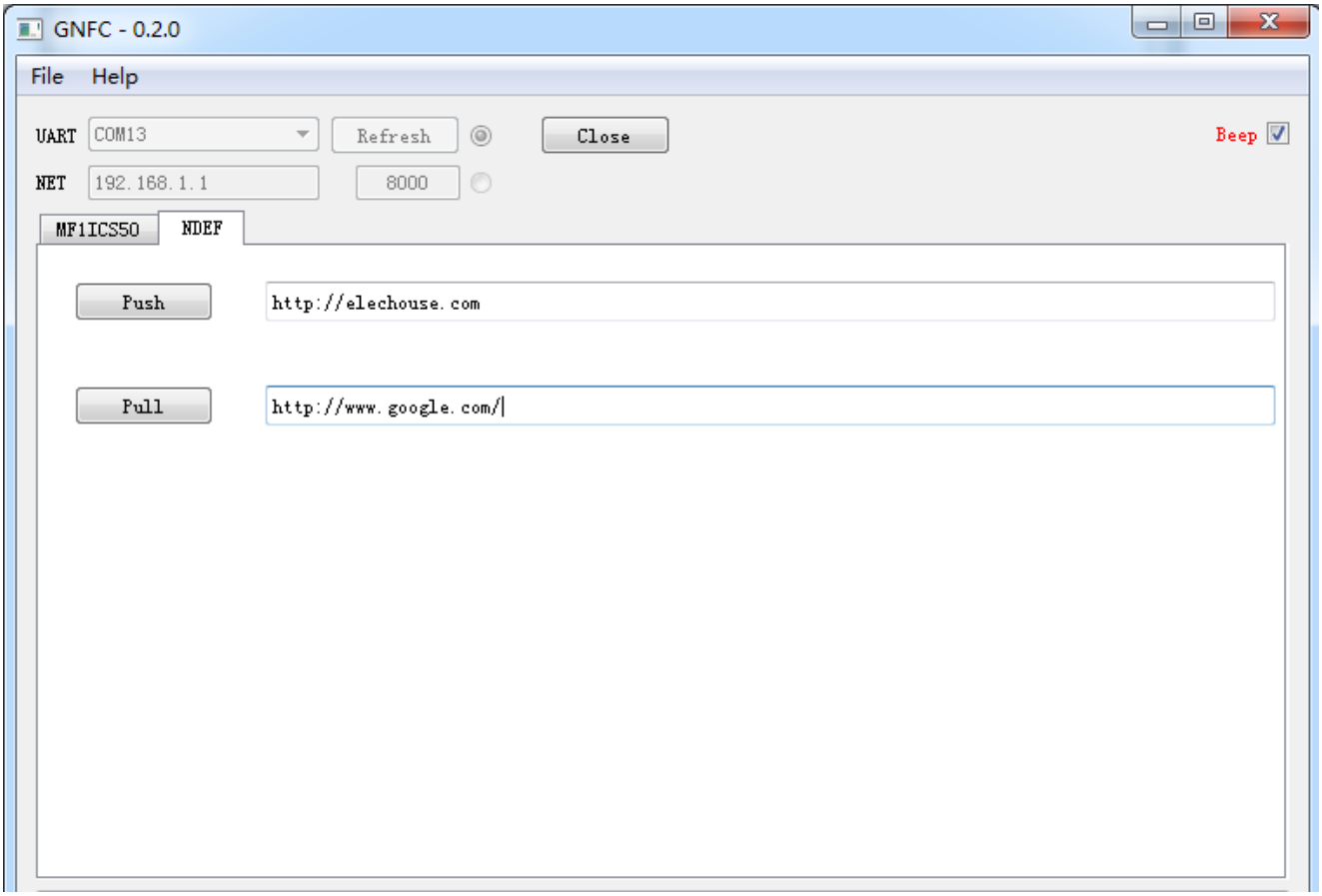
On your smart phone, set the NFC function in the right way. Start a browser and visit a website such as www.google.com

Click “**Pull**” button.



Within 5 seconds, put your phone on the device.

On the phone, the browser would room in. Click it. The URL in the browser would be pushed to the software on PC.



Input an URL (such as www.google.com) in the text box behind the button “**Push**”.

Click the button “**Push**”. Put a phone on the device. The browser would pop up and open the website.

Linux

QT5

1. Download qt-linux-opensource-5.x.x-x86-offline.run
2. sudo chmod +x qt-linux-opensource-5.1.1-x86-offline.run
3. ./qt-linux-opensource-5.1.1-x86-offline.run
4. next...next...next...

libnfc

so far, pn532_net driver is not included by libnfc, checkout JiapengLi/libnfc instead, by which net driver is included. To use net driver specify a device name like "pn532_net:192.168.1.1:8000".

```
autoreconf -vis

./configure --with-drivers=pn532_uart,pn532_net --sysconfdir=/etc --prefix=/usr

make

sudo make install all
```

libfreefare

```
autoreconf -vis

./configure --prefix=/usr

make

sudo make install
```

libllcp

```
autoreconf -vis

./configure --prefix=/usr

make

sudo make install
```

libndef

```
qmake

mingw32-make
```

lib files locate at *libndef/libndef/release/*, named **libndef1.a** and **ndef1.dll**, libndef/include/*.h is also needed.

FCC Compliance statement

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

(1) This device may not cause harmful interference

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

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

Caution:

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation.

This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications.

However, there is no guarantee that interference will not occur in a particular installation.

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

• Reorient or relocate the receiving antenna.

• Increase the separation between the equipment and receiver.

• Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.

• Consult the dealer or an experienced radio/TV technician for help.