

# Introduction to NFC*Link*

**Texas Instruments  
Safety and Security MCU  
NFC/RFID Applications Team**

11/25/2014

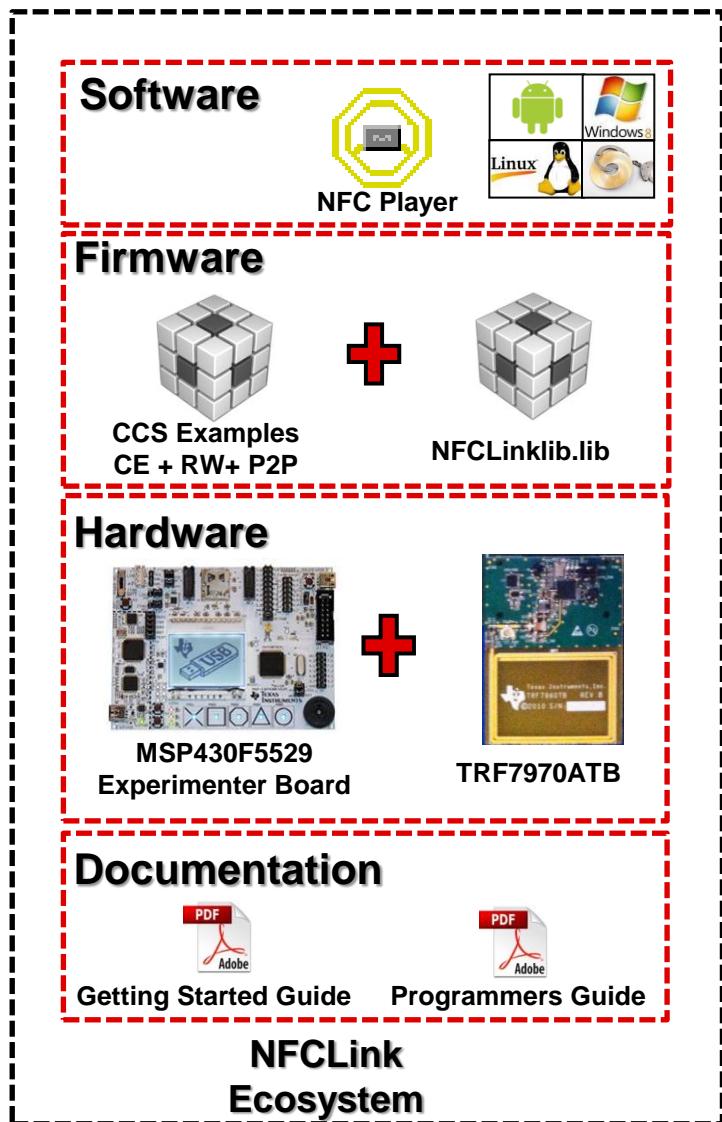
# Agenda

- NFCLink Overview
- NFCStack + Evaluation Overview
- NFCLink Beta Release components
- Hardware Changes
- NFCLink Demo – “Out of box experience”
  - Hardware + Software Configuration
  - Demonstration examples
    - Reader/Writer
    - Card Emulation
    - Peer to Peer
- NFCLink File Structure
- Porting to other MSP430s
- Memory Footprints
- Summary

# NFCLINK OVERVIEW

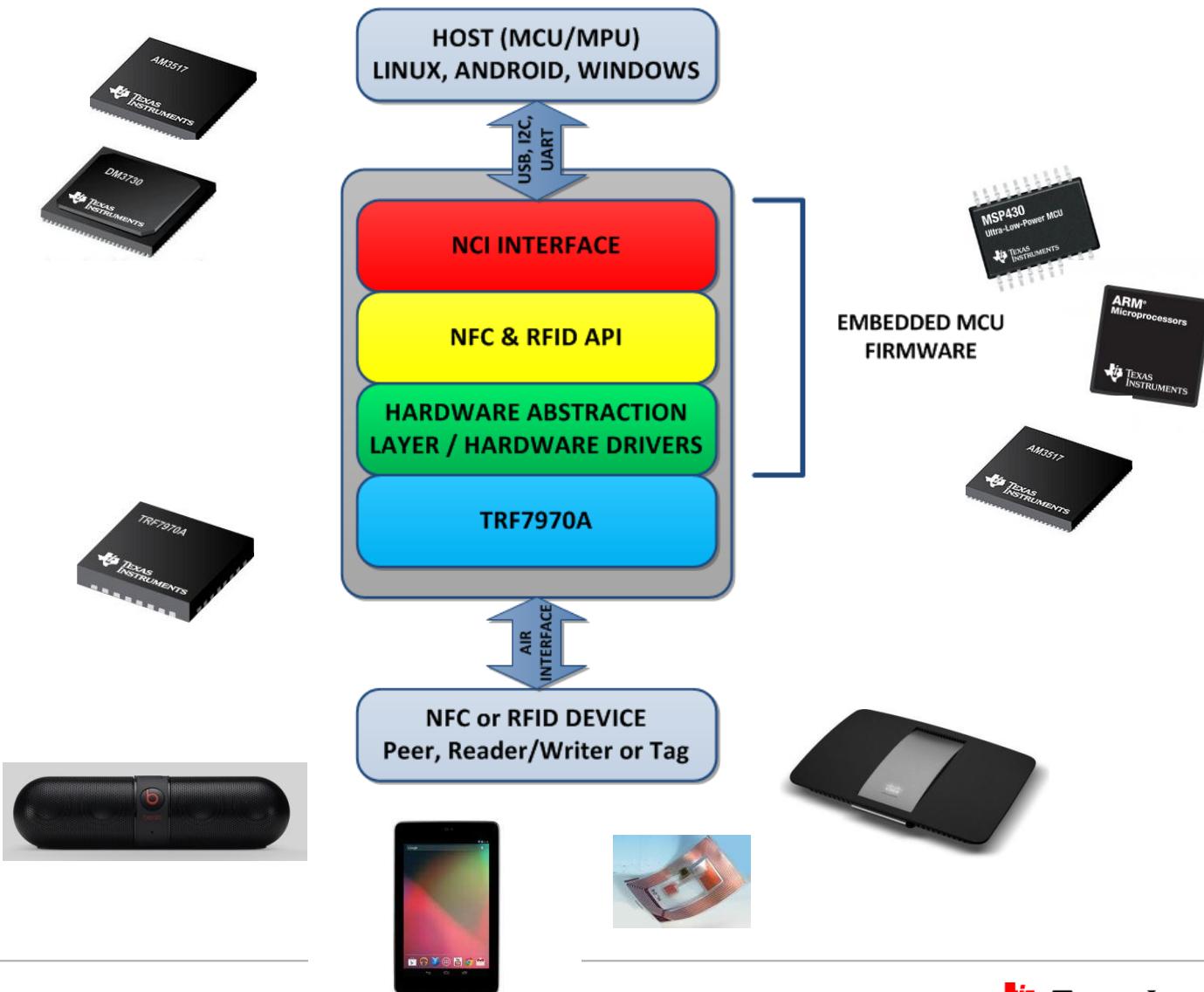
11/25/2014

# What is NFCLink and what is it used for?



11/25/2014

# NFCLink Architecture



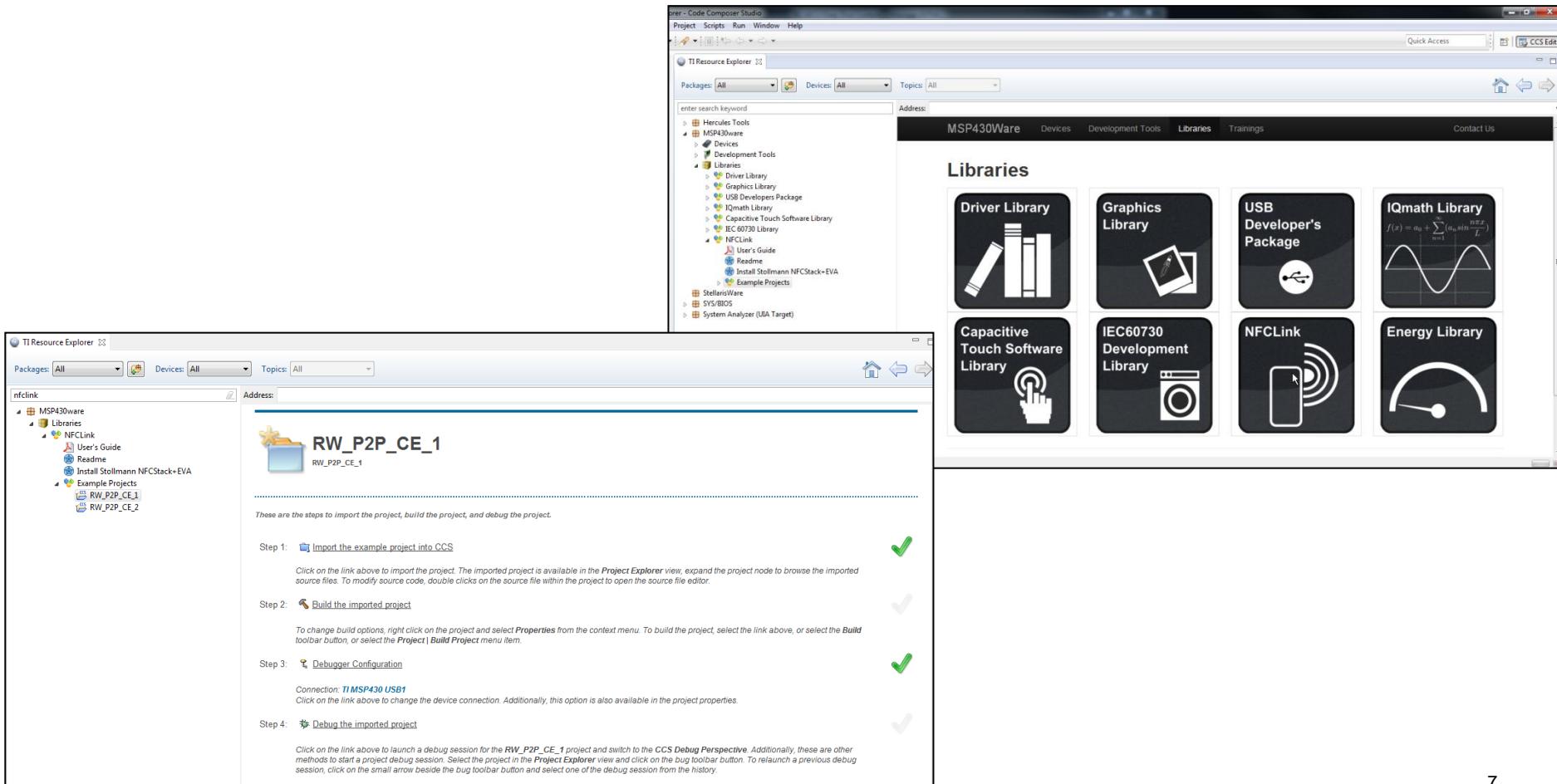
# NFCLink Tool Folder (Product Page)

- NFCLink has a dedicated URL: <http://www.ti.com/tool/nfclink>
- Download includes standalone installer for firmware folder, GUI and reference documentation.

The screenshot shows the Texas Instruments website for the NFCLink NFC/HF RFID Firmware. The page includes a navigation bar with links to Products, Applications & Designs, Tools & Software, Support & Community, Sample & Buy, and About TI. A search bar is also present. The main content area displays the product details for NFCLink, including its part number, availability from Texas Instruments or third parties, and an 'Alert Me' button. Below this is a 'Description & Features' section, which is currently expanded, showing a detailed list of features such as support for various NFC and HF tag types and platforms. To the right of the main content is a smaller window showing a screenshot of the NFCLink software interface, which is a graphical user interface for managing NFC and HF RFID devices. The software window shows various tabs and lists related to device management and configuration.

# MSP430Ware

- NFCLink is also included as a library inside MSP430Ware



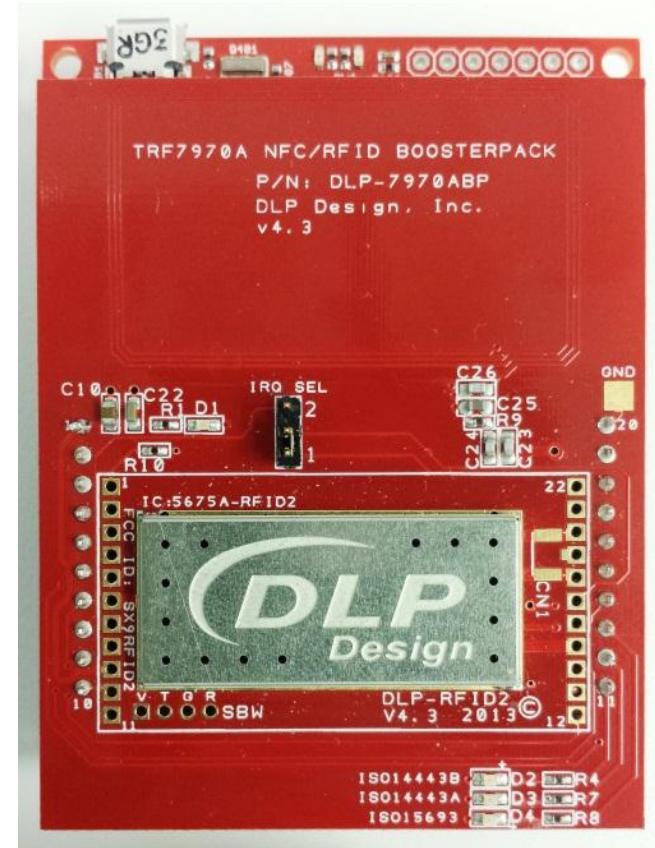
# TI Design Status

- The TRF7970ATB was added to TI Designs, with NFCLink as the firmware and software deliverable.
- It is available at the following URL → <http://www.ti.com/tool/TIDM-NFC-TRANSCEIVER>

The top screenshot shows the main product page for the TRF7970ATB. It features a large image of several NFC cards, a price of \$49.00(USD), and a prominent 'Order Now' button. The bottom screenshot provides more detailed resources: a Schematic/Block Diagram, a Design Guide (with a screenshot of the software interface), and Source Files (with a screenshot of the software interface). Each resource has a corresponding 'Download' button below it.

# Hardware changes since June 2013

- Hardware additions:
  - MSP-EXP430F5529 is not currently available
  - MSP-EXP430F5529LP + TRF7970A based BoosterPack has been created and made available.
  - DLP-7970A BoosterPack is from DLP Design
    - <http://www.ti.com/tool/dlp-7970abp>
  - The DLP-7970ABP is an add-on board designed to fit TI's MCU LaunchPads that incorporates DLP Design's DLP-RFID2 RFID module.
  - This NFC/RFID BoosterPack allows the software firmware and/or application developer to get familiar with the functionalities of the TRF7970A multi-protocol, fully-integrated, 13.56MHz RFID/NFC IC on their Texas Instruments embedded microcontroller platform of choice without having to worry about developing the RF section.

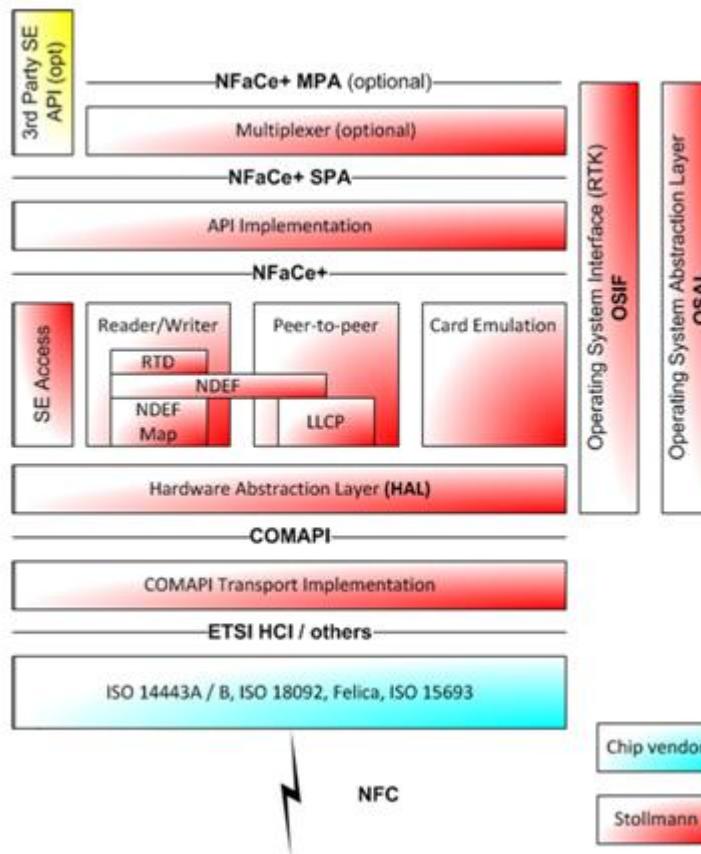


# NFCSTACK + EVALUATION OVERVIEW

11/25/2014

# What is NFCStack+Eva and what is its Architecture?

- High level overview of the Windows GUI implementation.  
In this case the TRF7970A + MSP430 would be the chip vendor.



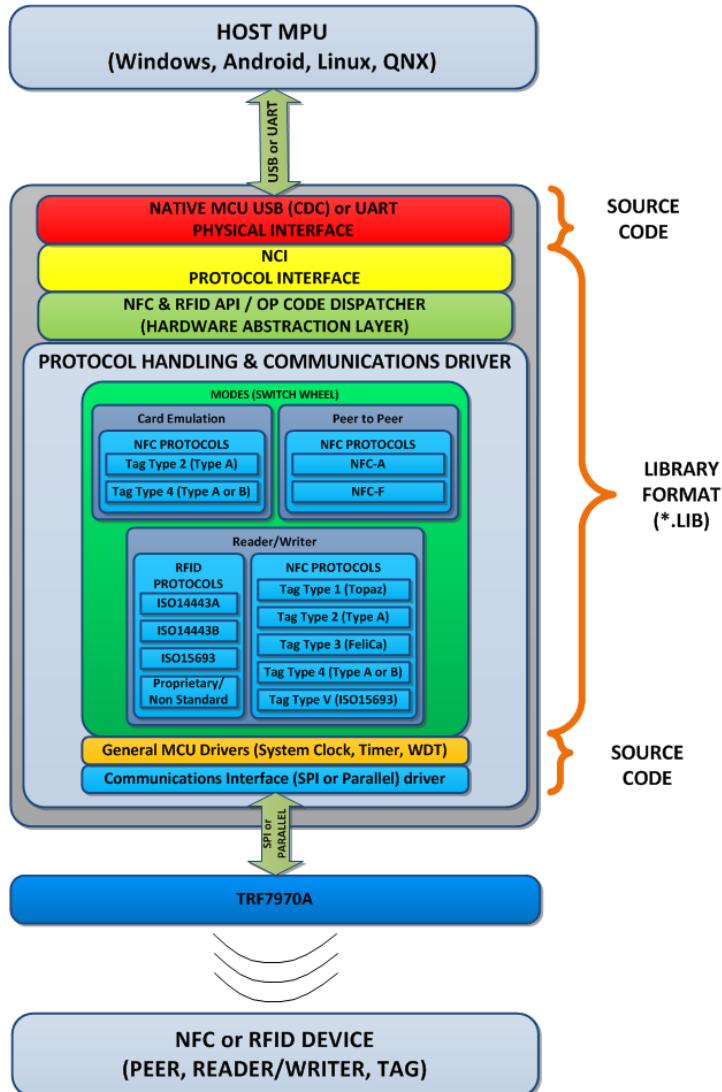
# Where to Integrate the NFCStack

- In the first implementation which is releasing on June 18<sup>th</sup>, 2013 – the NFCStack+ is compiled as a Windows OS GUI with an installer, for demonstration purposes.
- In an actual application, the NFCStack+ would be compiled as:
  - a driver component into the OS using applicable compiler and the supplied source code. (host integration / integrated host)
  - NFC module – run full or parts of the protocol stack on a dedicated CPU w/memory (commonly used for high volume/lower cost, automotive or POS applications)
  - OS to NFC Controller (embedded MCU + TRF7970A) – same as first example, but running the stack on the OS.
    - This is basically what the NFCLink solution is currently.

# NFC BETA RELEASE COMPONENTS

11/25/2014

# What is in June 18, 2013 release?



**Other components of the release will be:**

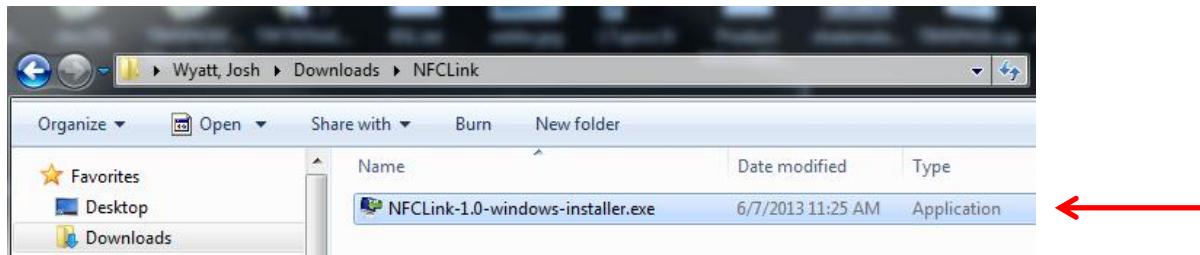
- NFCStack+Eva\_r6.0.47.5\_RC.exe (Windows GUI installer)
- NFCLink compiled library (nfclinklib.lib)
- CCS Projects
  - RW\_P2P\_CE1\_Example – for USB CDC applications
  - RW\_P2P\_CE2\_Example – for UART applications
- Getting Started Guide (for the demo)

# NFCLINK DEMO “OUT OF BOX EXPERIENCE”

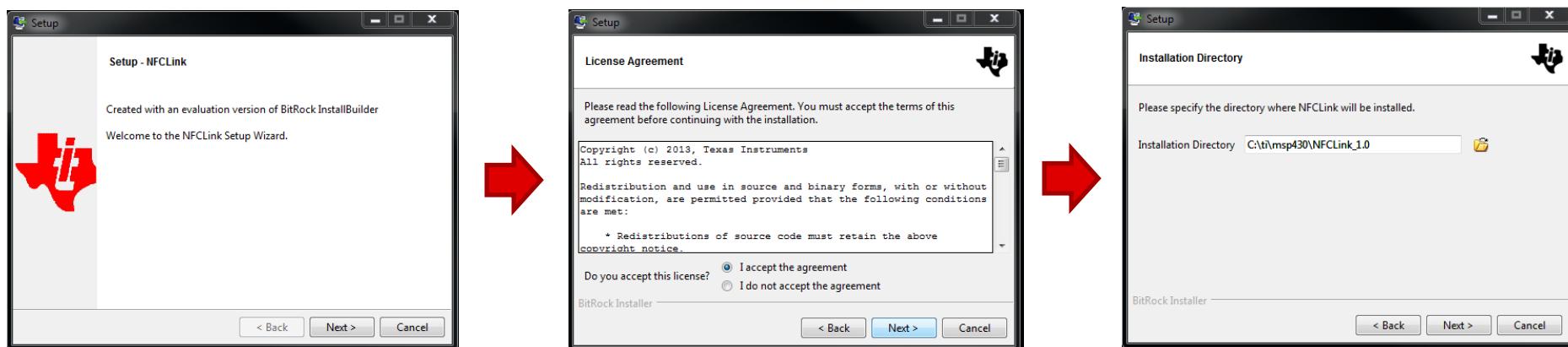
11/25/2014

# Install NFCLink-1.0

- Use the NFCLink Installer executable to get the firmware code project and the NFCPlayer GUI loaded onto PC.

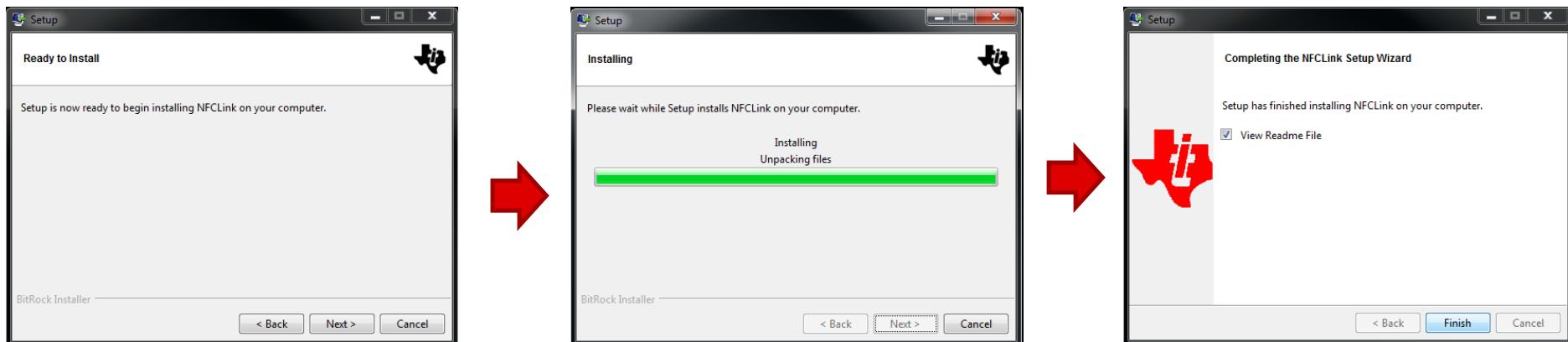


- Double click installer and follow instructions

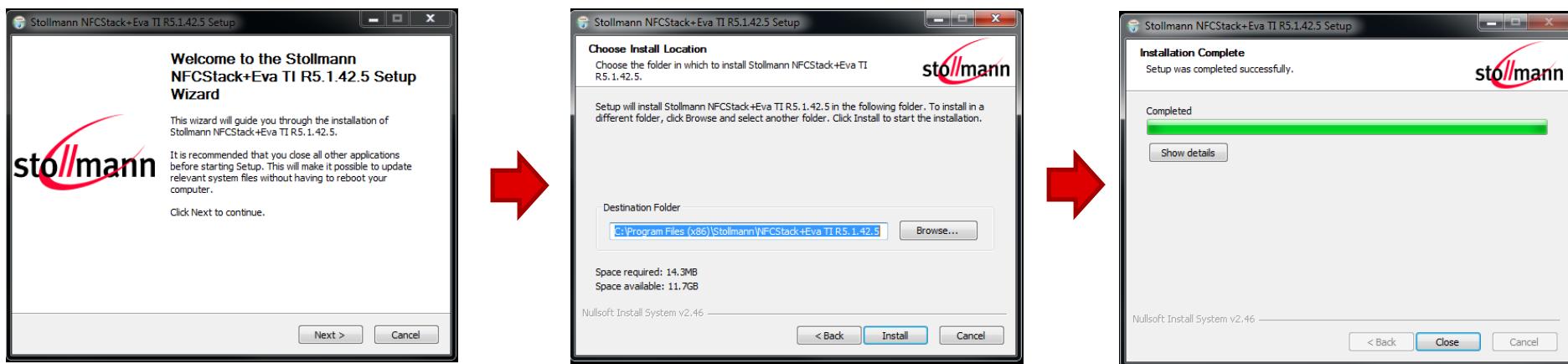


- Continues on next slide

# Install NFCLink-1.0 (cont.)

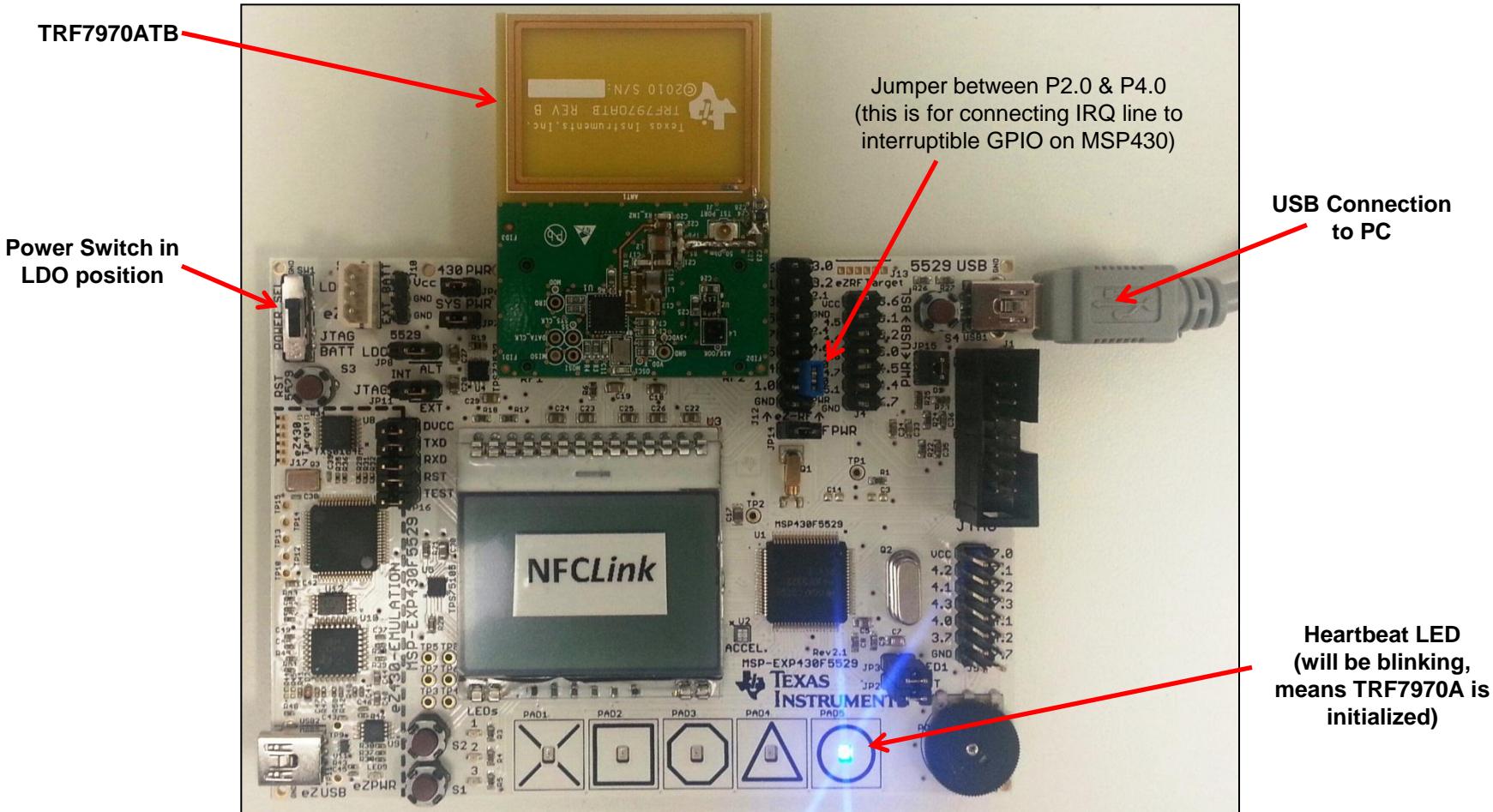


- Then the NFCPlayer GUI will begin installation automatically



- When complete, just press close button and plug the hardware into USB port on PC. (see picture on next slide)

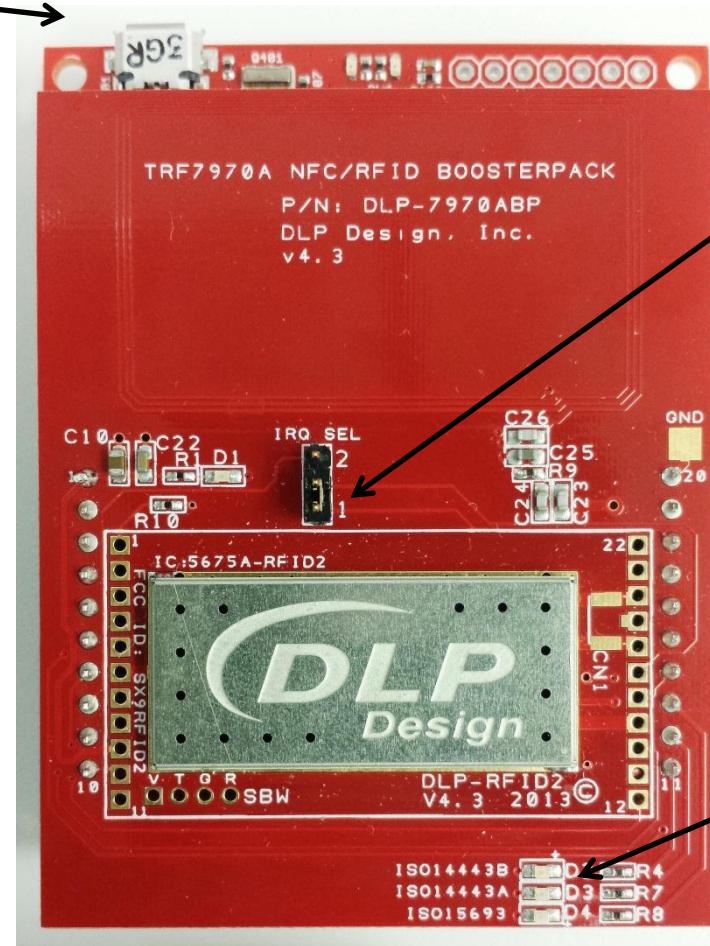
# Hardware Configuration Image (EXP Board)



11/25/2014

# Hardware Configuration Image (EXP LP Board)

USB Connection  
to PC /  
Emulation  
Connection

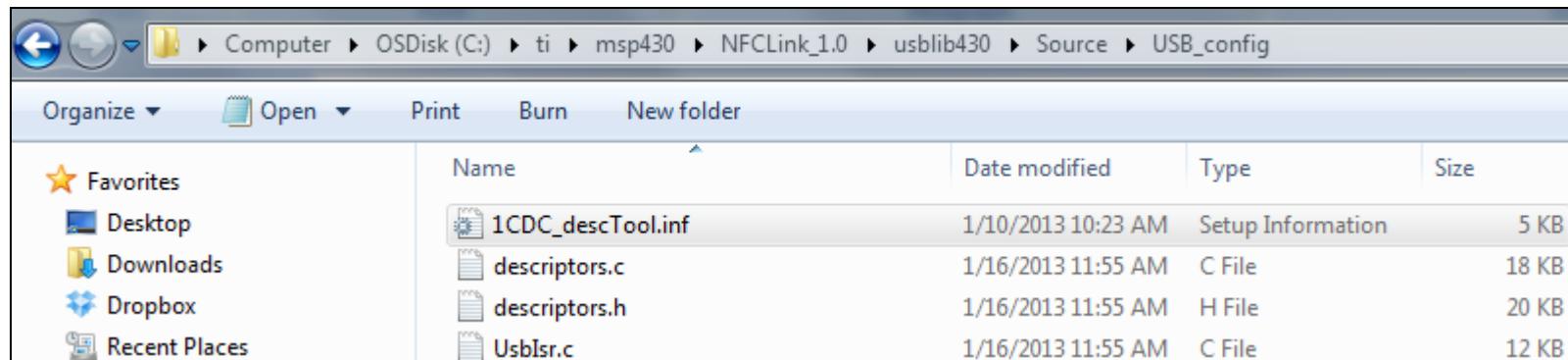


Heartbeat LED  
(on LP board, below the BP)  
(blinking means TRF7970A is  
initialized)

# Install USB Driver

- The first time the hardware is plugged into PC, the USB driver will need to be installed by pointing the wizard to the correct location/path.
- The file path for device manager wizard is:

C:\ti\msp430\NFCLink\_1.0\usblib430\Source\USB\_config



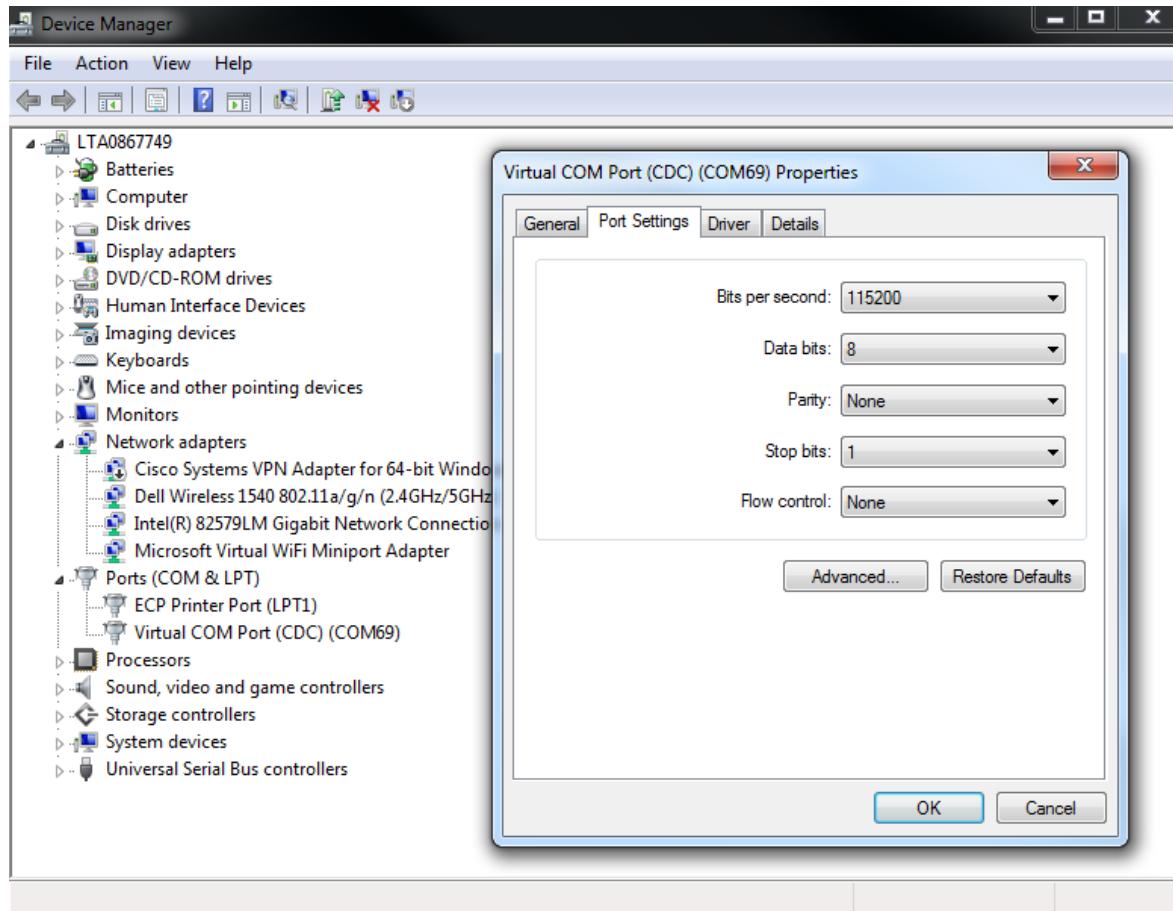
A screenshot of a Windows File Explorer window. The address bar shows the path: Computer > OSDisk (C:) > ti > msp430 > NFCLink\_1.0 > usblib430 > Source > USB\_config. The left sidebar has 'Favorites' with icons for Desktop, Downloads, Dropbox, and Recent Places. The main area shows a list of files in the USB\_config folder:

	Name	Date modified	Type	Size
	1CDC_descTool.inf	1/10/2013 10:23 AM	Setup Information	5 KB
	descriptors.c	1/16/2013 11:55 AM	C File	18 KB
	descriptors.h	1/16/2013 11:55 AM	H File	20 KB
	UsbIsr.c	1/16/2013 11:55 AM	C File	12 KB

# COM Port Settings

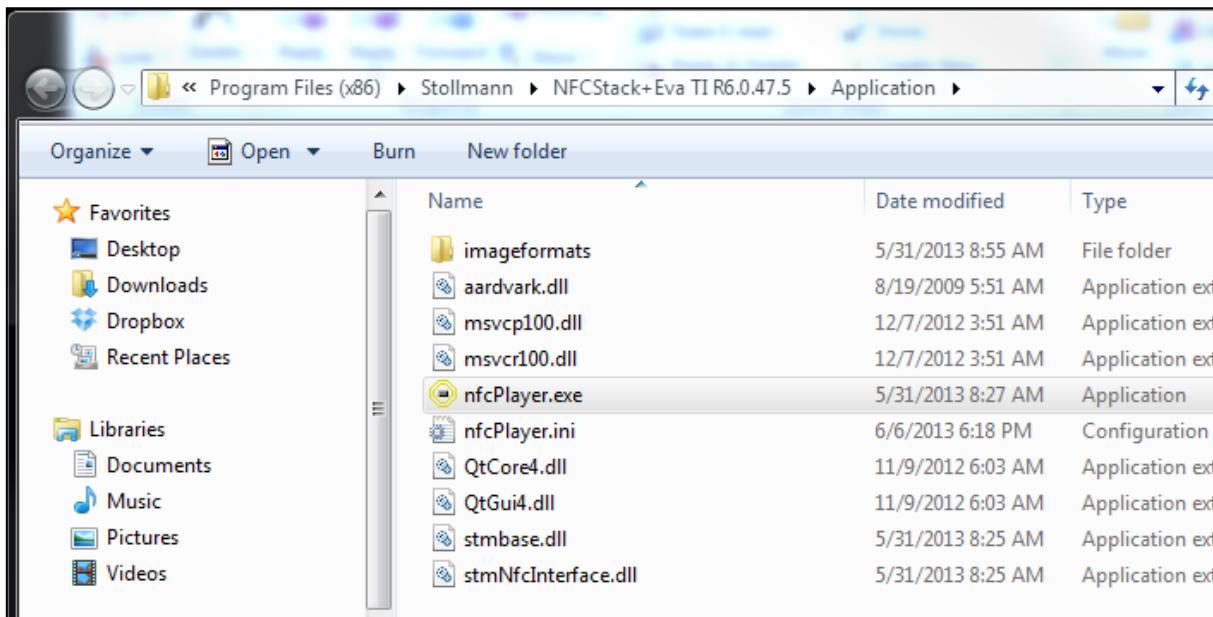
**NOTE: your COM port # most likely will be different**

- Adjust Virtual COM Port settings for 115200, 8N1



# Execute NFCPLayer (PC GUI)

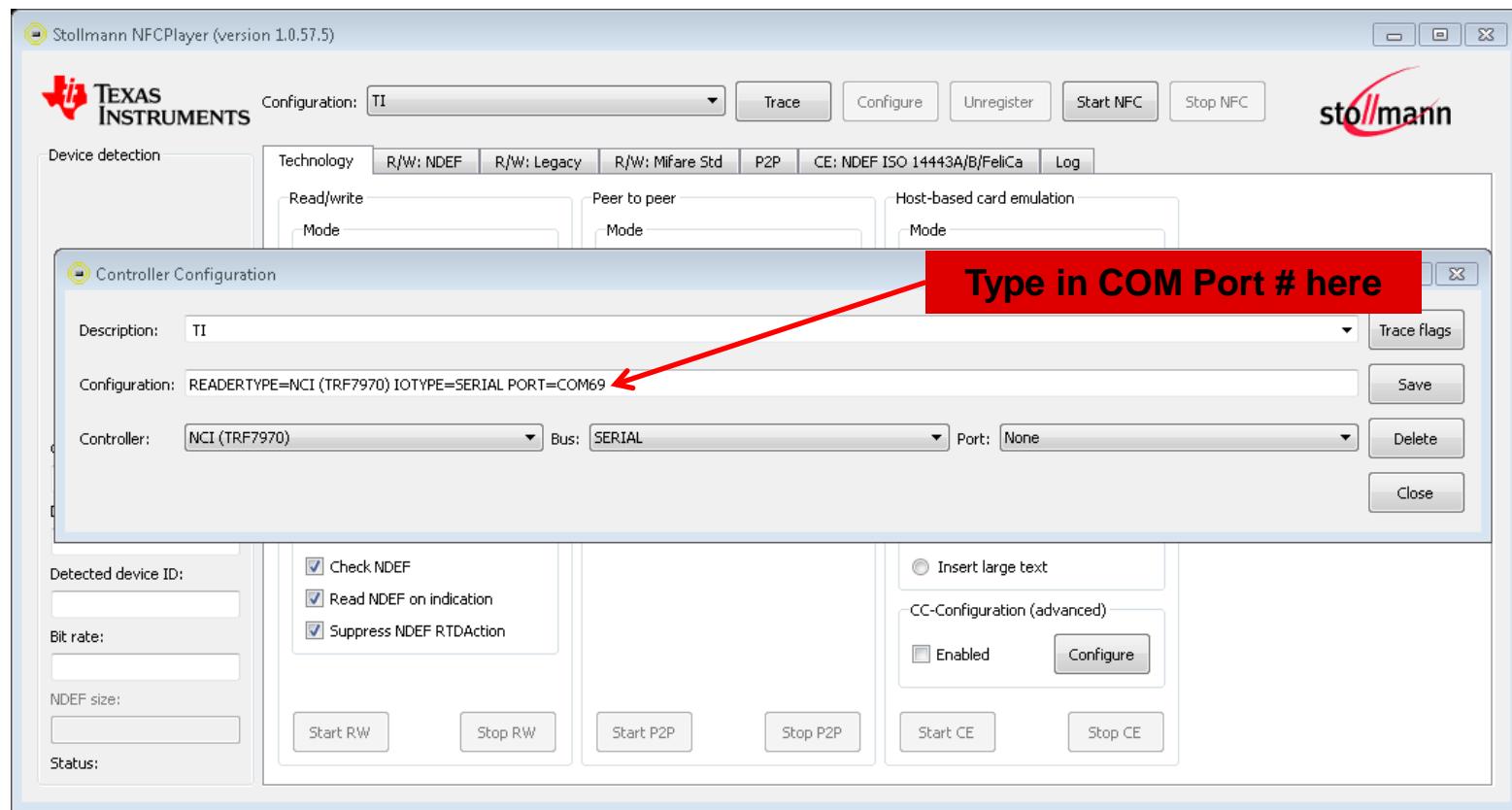
- Choose to execute NFCPLayer from Programs in PC start menu under Stollmann NFCStack+Eva TI R6.0.47.5 Folder by double clicking the nfcPlayer.exe file shown below.



- The GUI will open, see next slide.

# PC GUI (configuration)

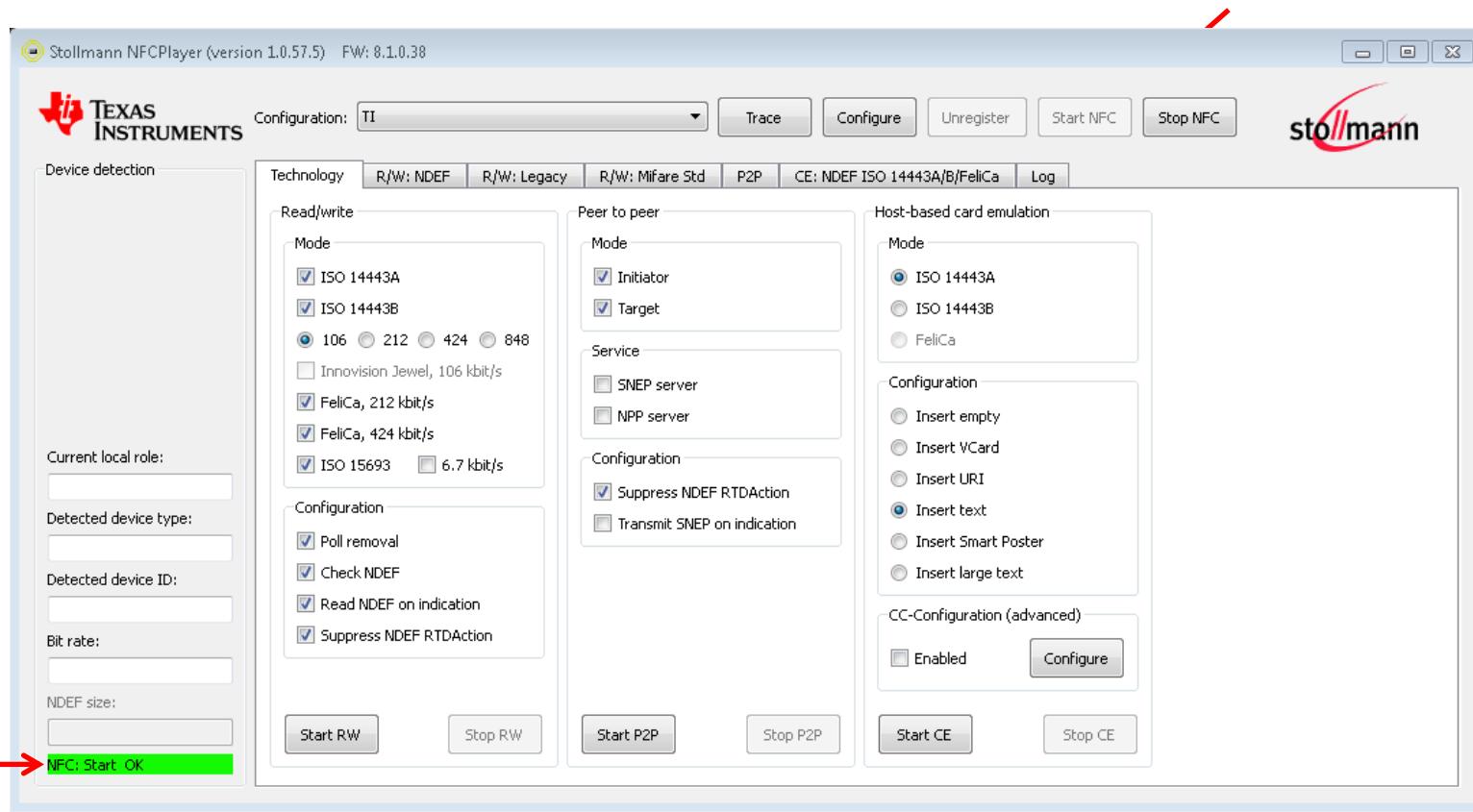
- After GUI opens, press the Configure button to pop up sub-screen and type in the number that the COM port enumerated to.
- Save this sub window, which will close it.



11/25/2014

# PC GUI (First Step after Config of the COM port)

- Press Start NFC Button, if correctly setup, then NFC: Start OK will appear in lower left hand window of the GUI.



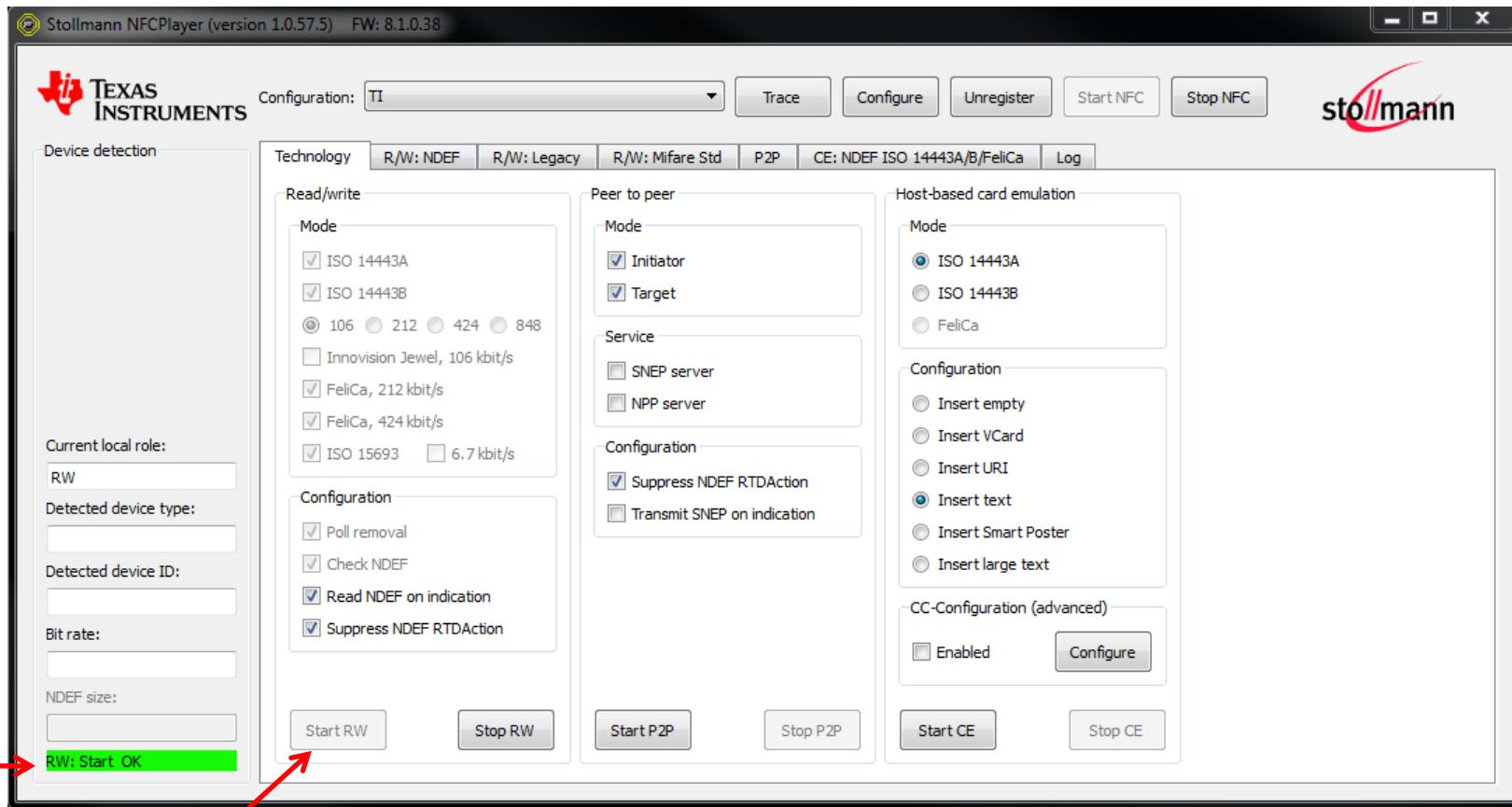
11/25/2014

# NFC/RFID READER/WRITER MODE

11/25/2014

# Reader/Writer Mode

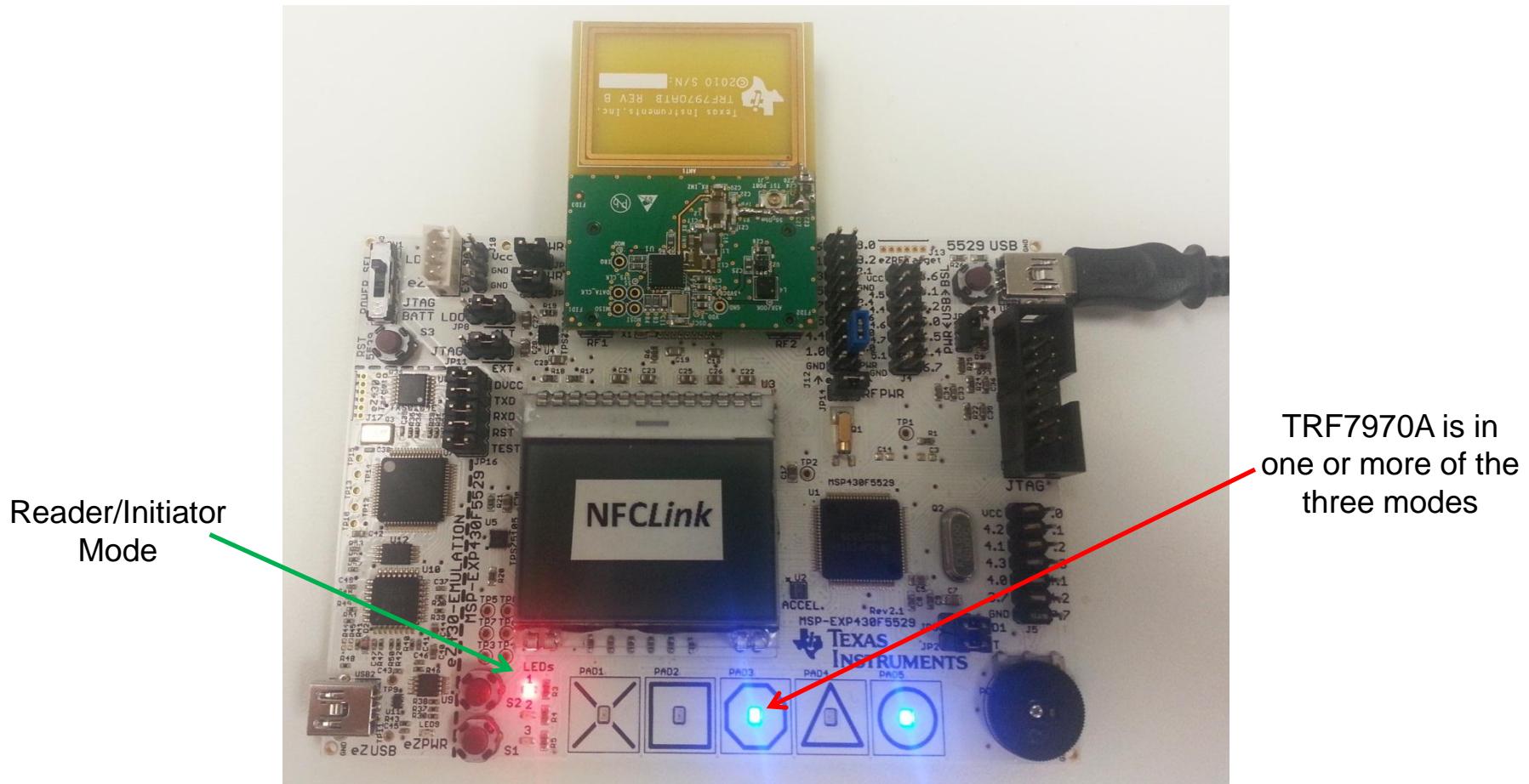
- For those wishing to use Reader/Writer Mode, press the Start RW button.
  - (RW: Start OK will appear in lower left hand window of the GUI)



11/25/2014

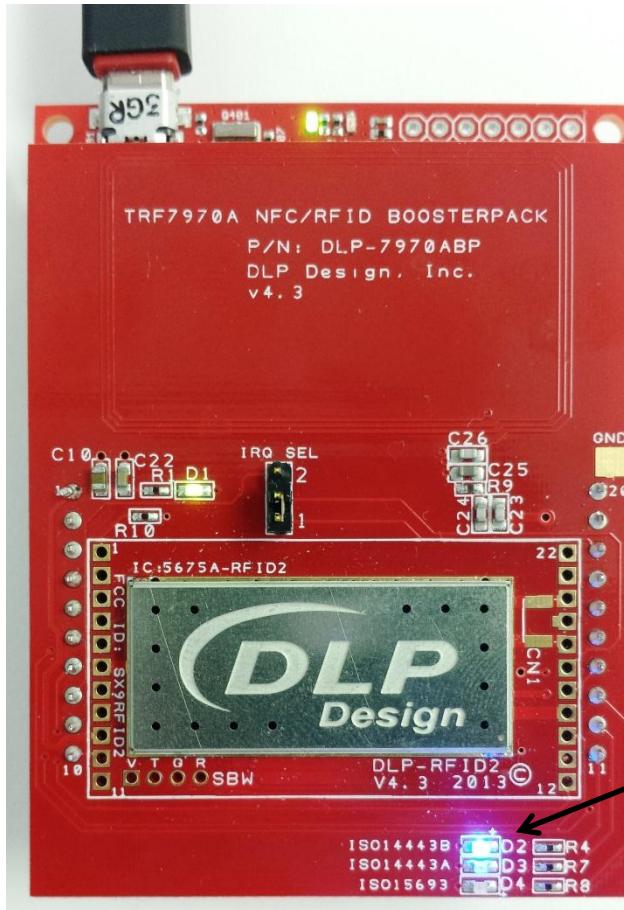
# Hardware LED status indicators (Reader/Writer Mode, EXP board)

- When hardware is in reader/writer mode, LED1 will be flashing and then go solid when an NFC/RFID tag is presented.



# Hardware LED status indicators (Reader/Writer Mode, EXP LP w/BP board)

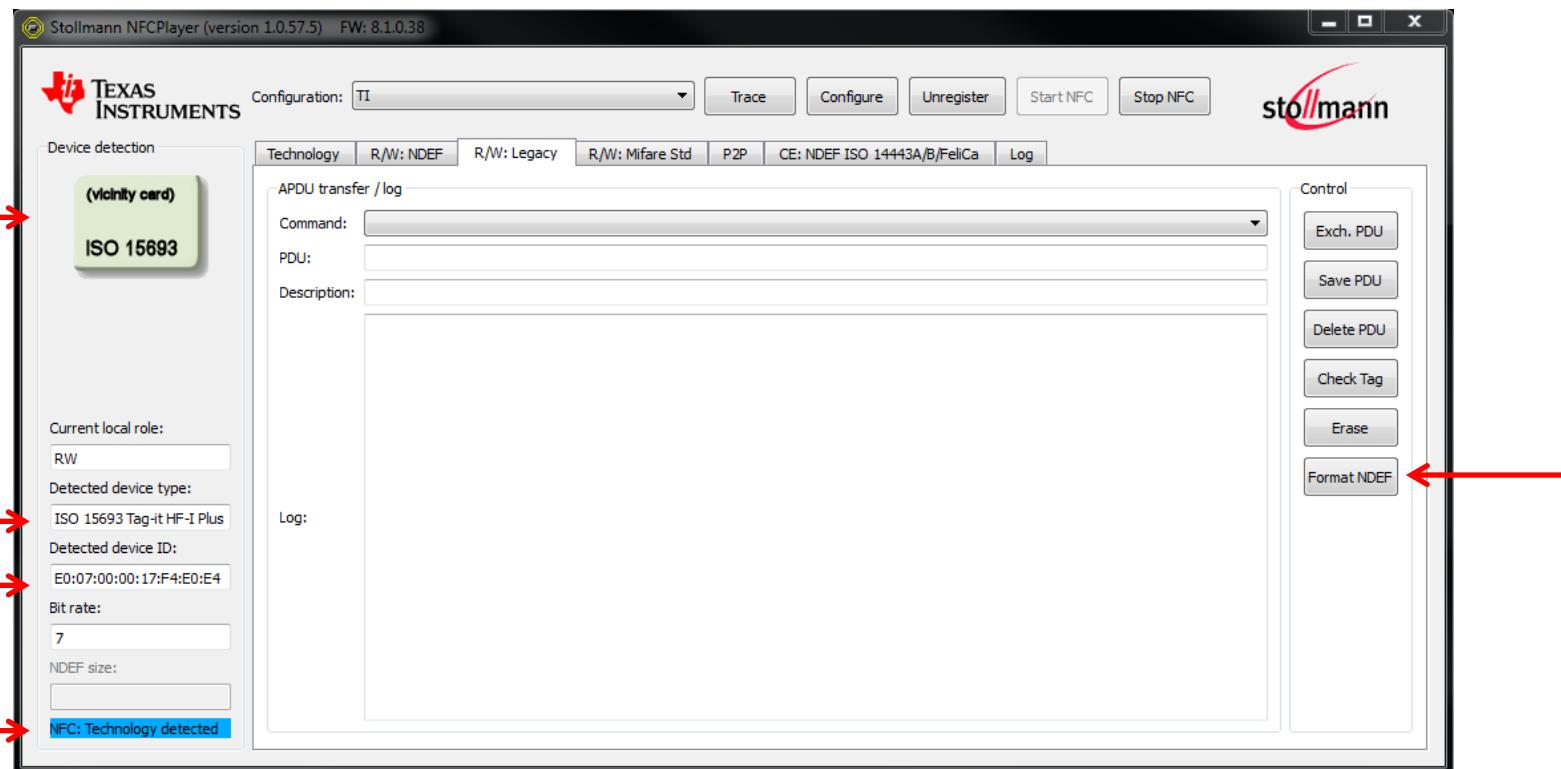
- When hardware is in reader/writer mode, LED labeled ISO14443B will be flashing and then go solid when an NFC/RFID tag is presented.



Reader/Initiator Mode  
(ISO14443B LED will be flashing)

# Reader/Writer Mode NFC-V tag

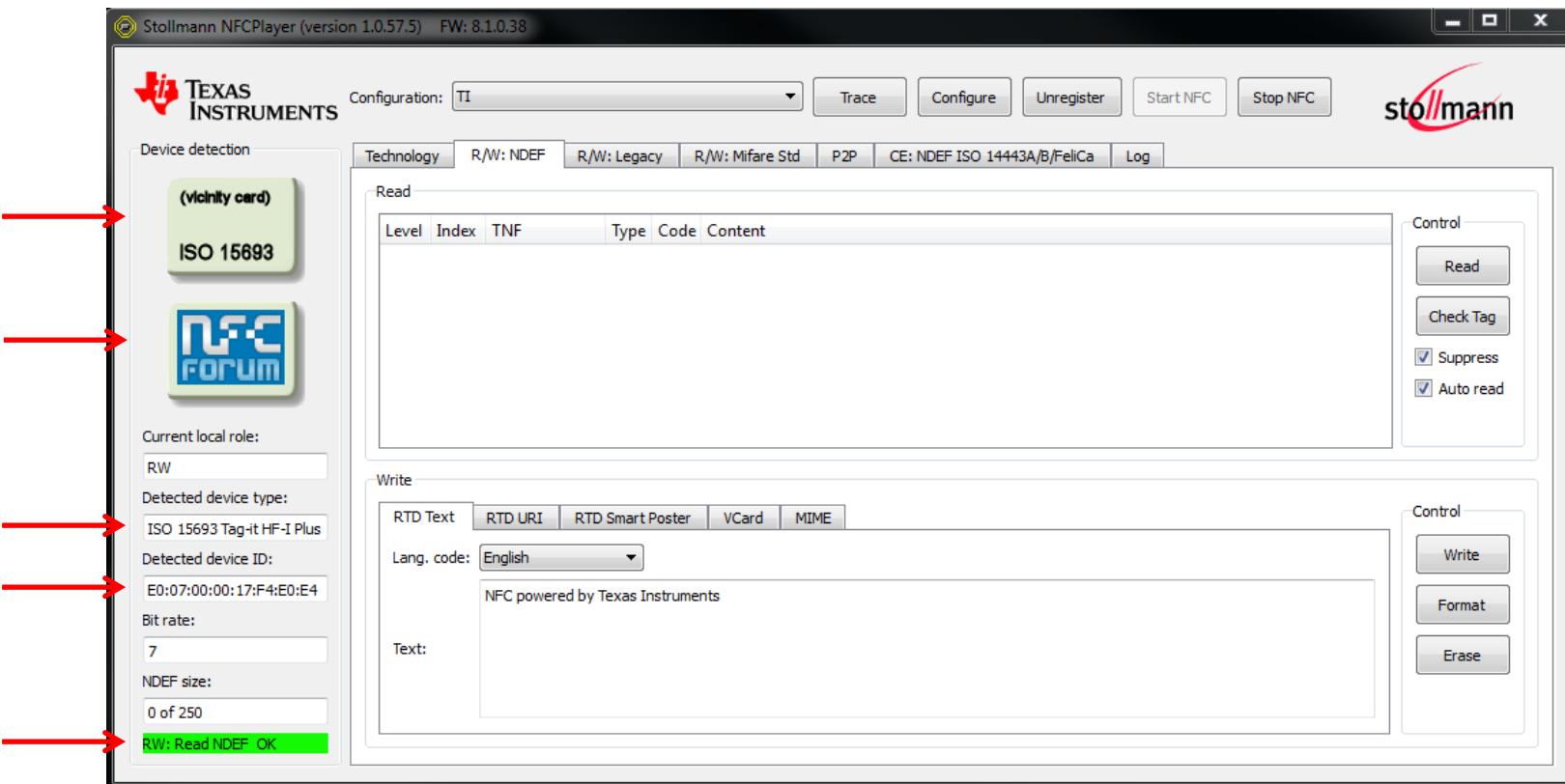
- After pressing the R/W button, present an NFC tag (formatted or not) – in this example we are presenting an unformatted NFC-V (ISO15693, TI HF-I) tag.
- Note the tab automatically flips to R/W: Legacy and the type of card is displayed graphically along with the Unique ID (in this case: E007000017F4E0E4)
- User can press the Format NDEF button to make the tag NFC-V type. (this puts Capability Container with empty NDEF in the user memory.)



11/25/2014

# Reader/Writer Mode NFC-V tag (cont.)

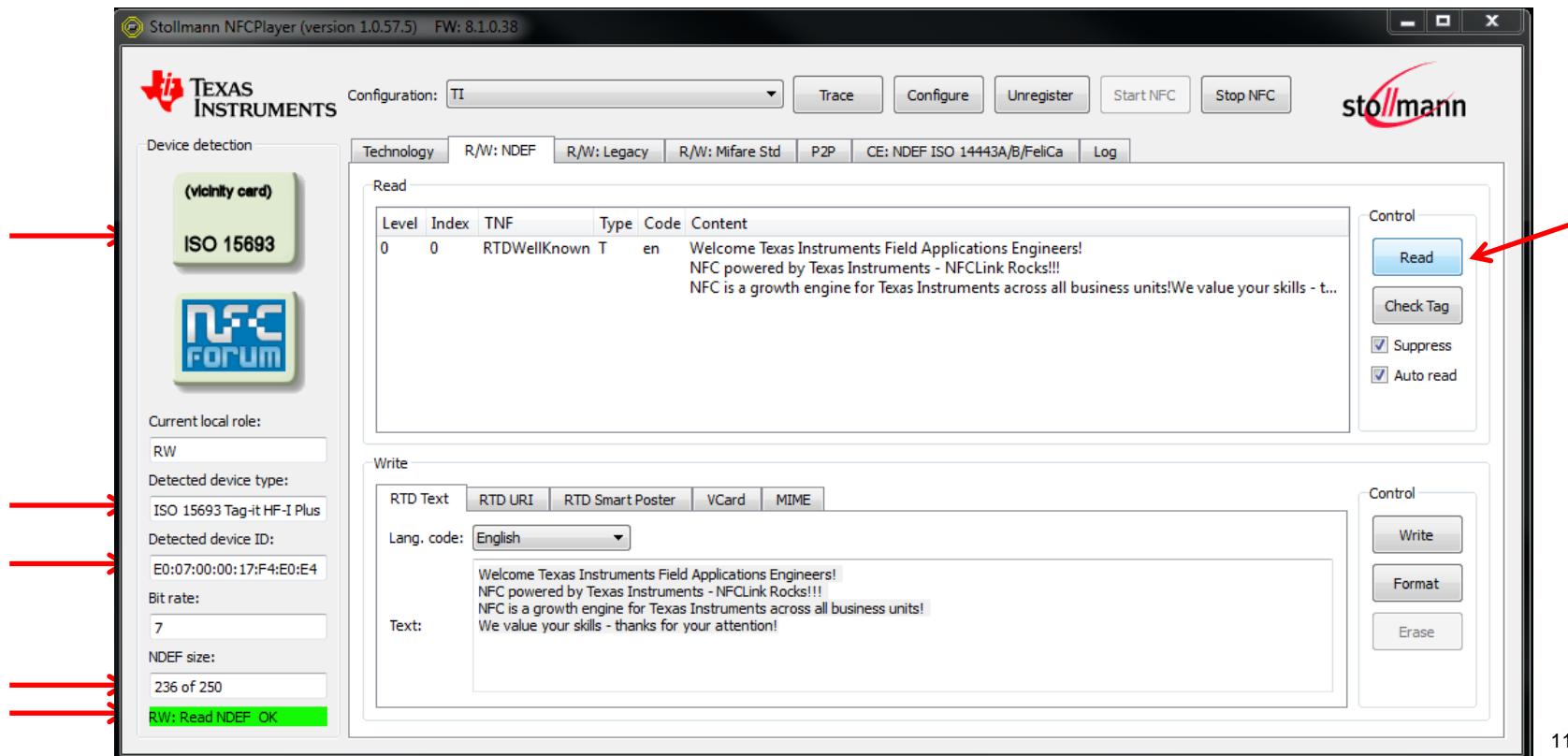
- After the tag is formatted, the GUI status window will quickly flip to Format NDEF OK, then present the R/W NDEF, with the status changing to RW: Read NDEF OK, below can be seen the results (formatted but empty tag)



11/25/2014

# Reader/Writer Mode NFC-V tag (cont.)

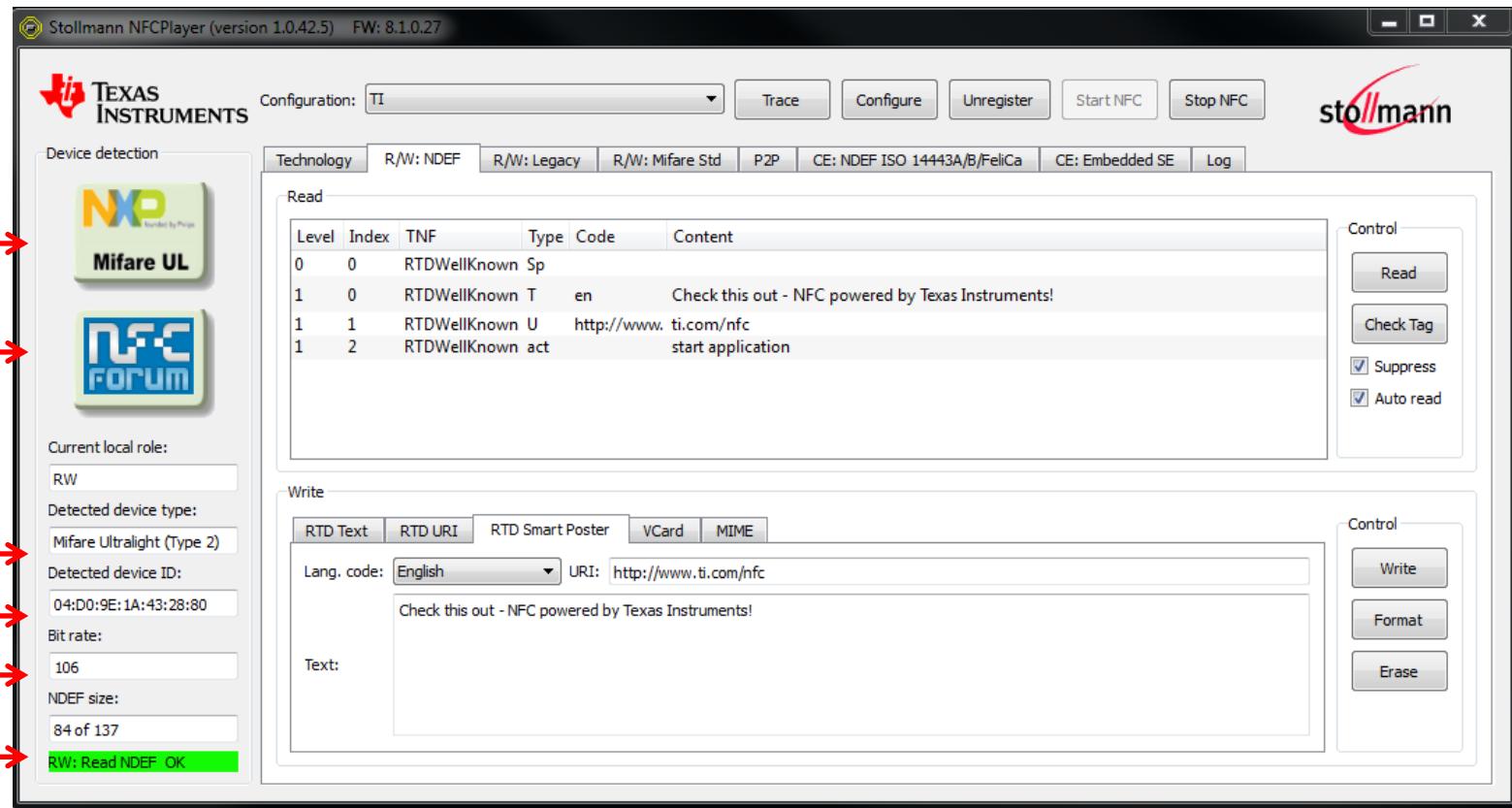
- User can now enter data into the bottom 'Write' window and press the Write button to write NDEF message into the tag.
- When this occurs, screen will quickly flip to Write NDEF , then the user can either press the Read button or remove the tag from the reader/writer antenna and re-present the tag to get the NDEF message read back



11/25/2014

# Reader/Writer Mode NFC Type 2 tag

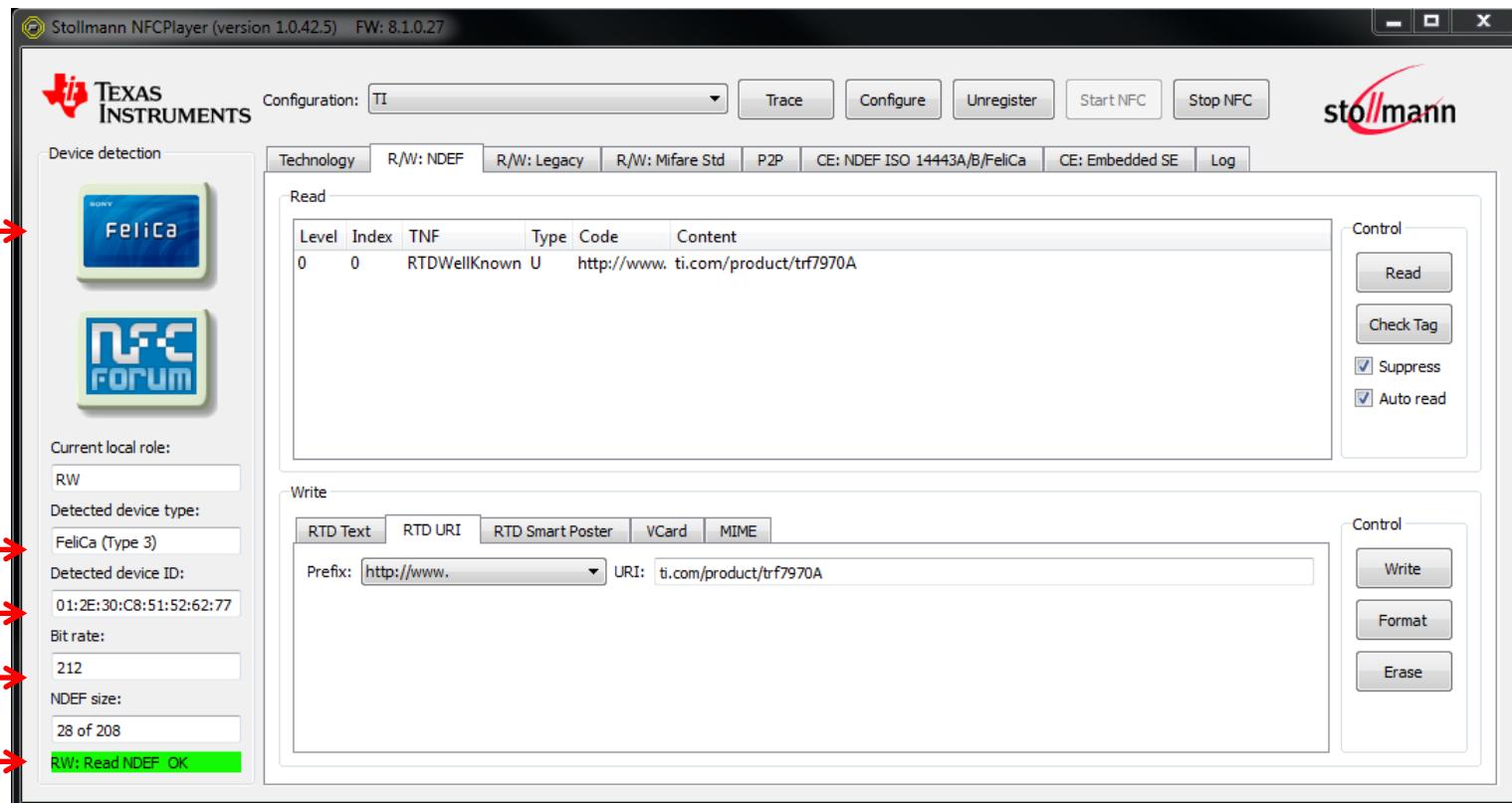
- Here is Tag Type 2 which has been formatted with the tool and then programmed as RTD Smart Poster.
- If Suppress NDEF RTD Action (in front panel) box is unchecked, presenting this tag will open browser up and will be routed to [www.ti.com/nfc](http://www.ti.com/nfc) web page.



11/25/2014

# Reader/Writer Mode NFC Type 3 tag

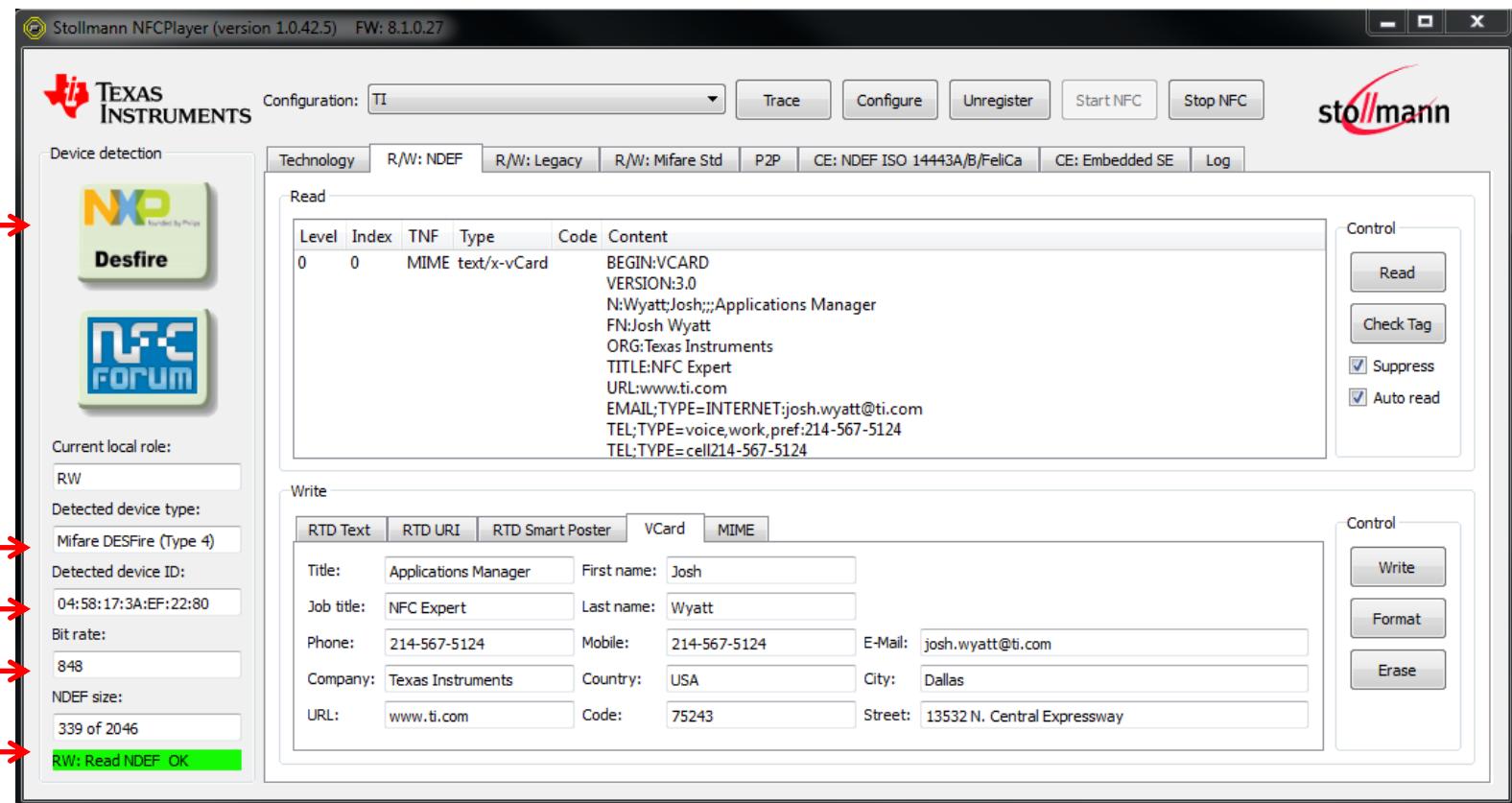
- Here is Tag Type 3 which has been formatted with the tool and then programmed as RTD URI.



11/25/2014

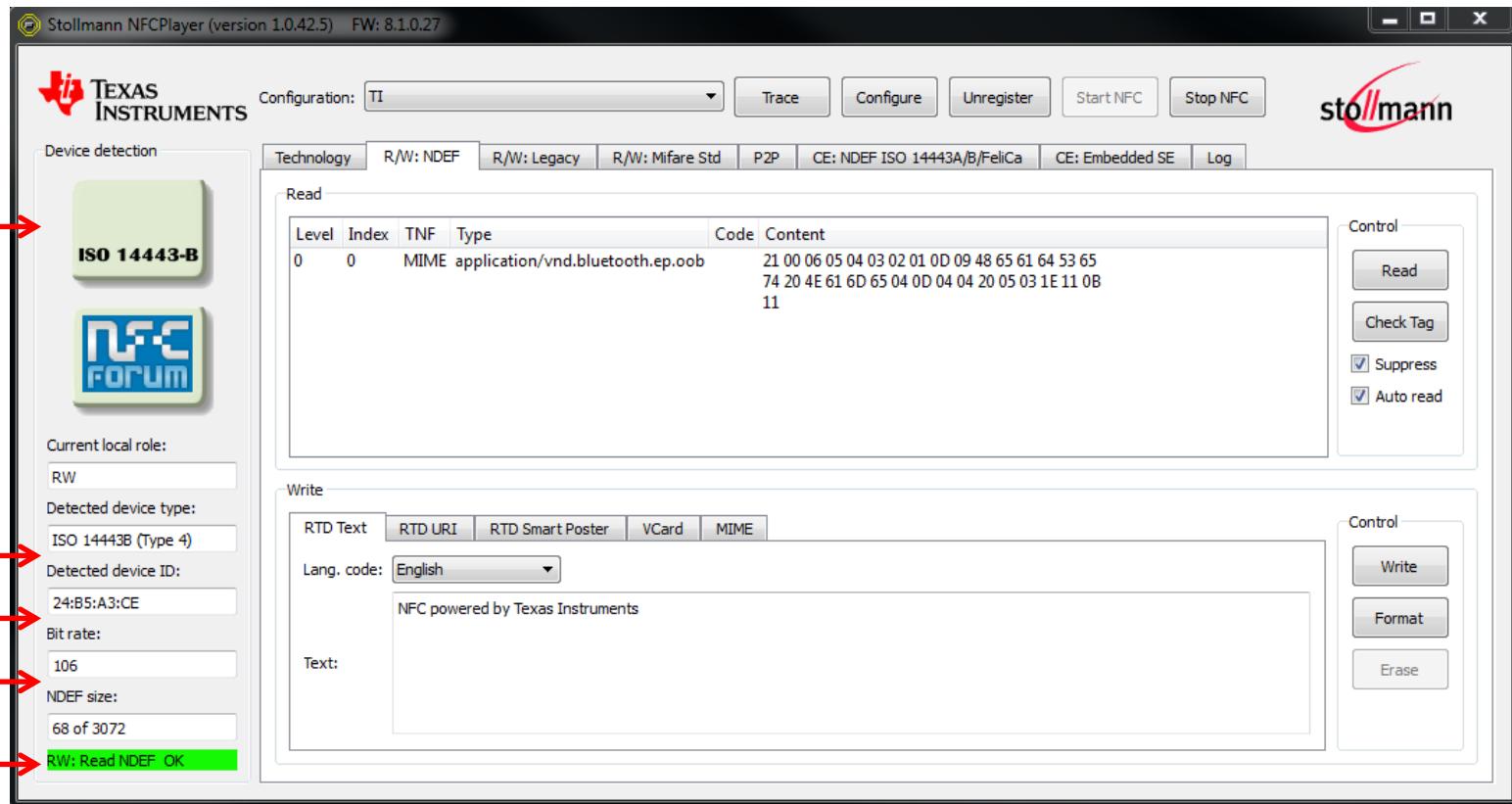
# Reader/Writer Mode NFC Type 4A tag

- Here is Tag Type 4A which has been formatted with the tool and then programmed as VCard.



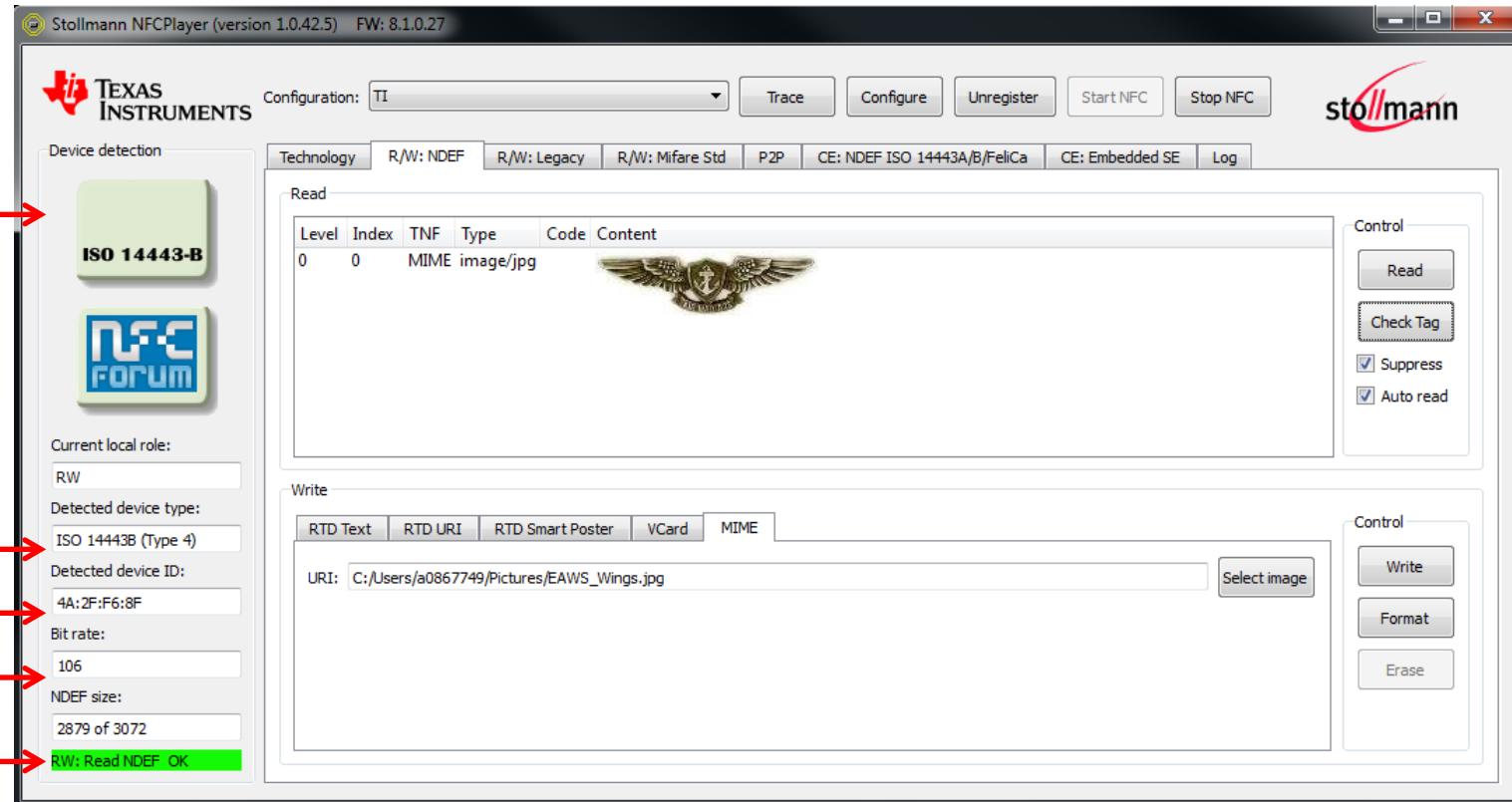
# Reader/Writer Mode NFC Type 4B tag

- Here is Tag Type 4B (RF430CL330H) which has been formatted with the tool and then programmed for NFC Forum Bluetooth Connection Handover.



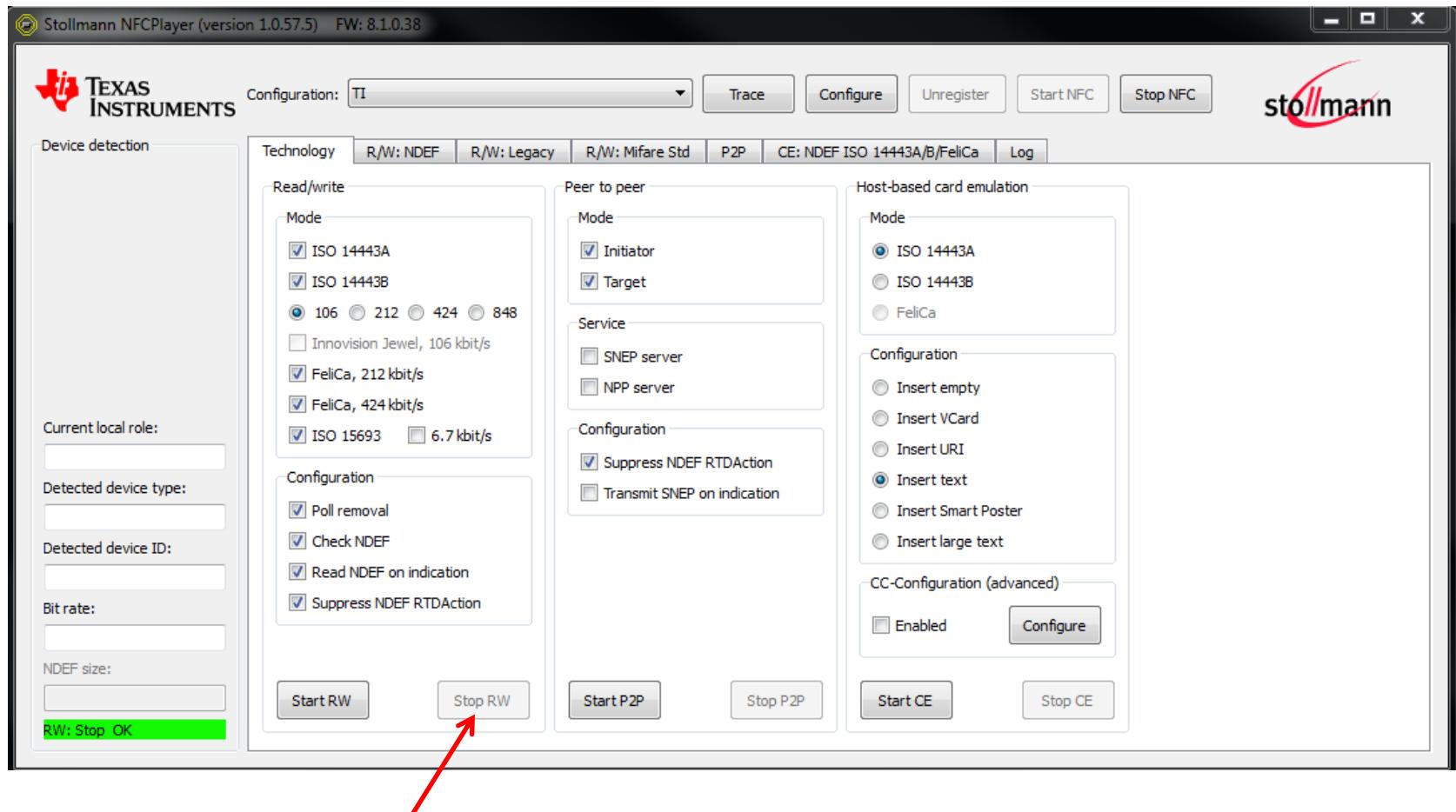
# Reader/Writer Mode NFC Type 4B tag (cont.)

- Here is Tag Type 4B (RF430CL330H) which has been formatted with the tool and then programmed with an image (MIME).



# Exiting or Stopping the Reader/Writer Mode

- To stop the R/W mode, remove the tag in the field and press the Stop RW button.



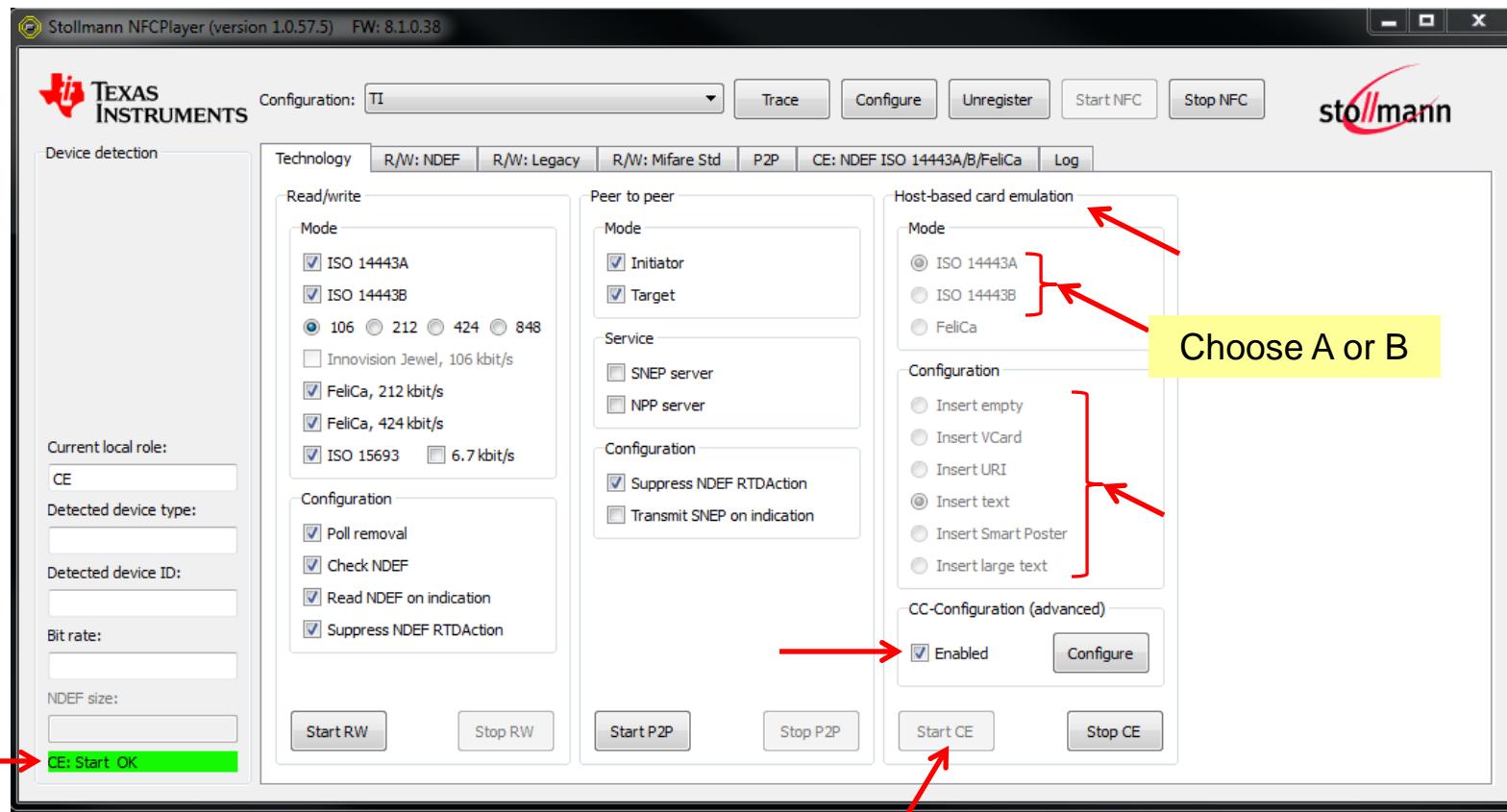
11/25/2014

# NFC/RFID CARD EMULATION MODE

11/25/2014

# Card Emulation Mode

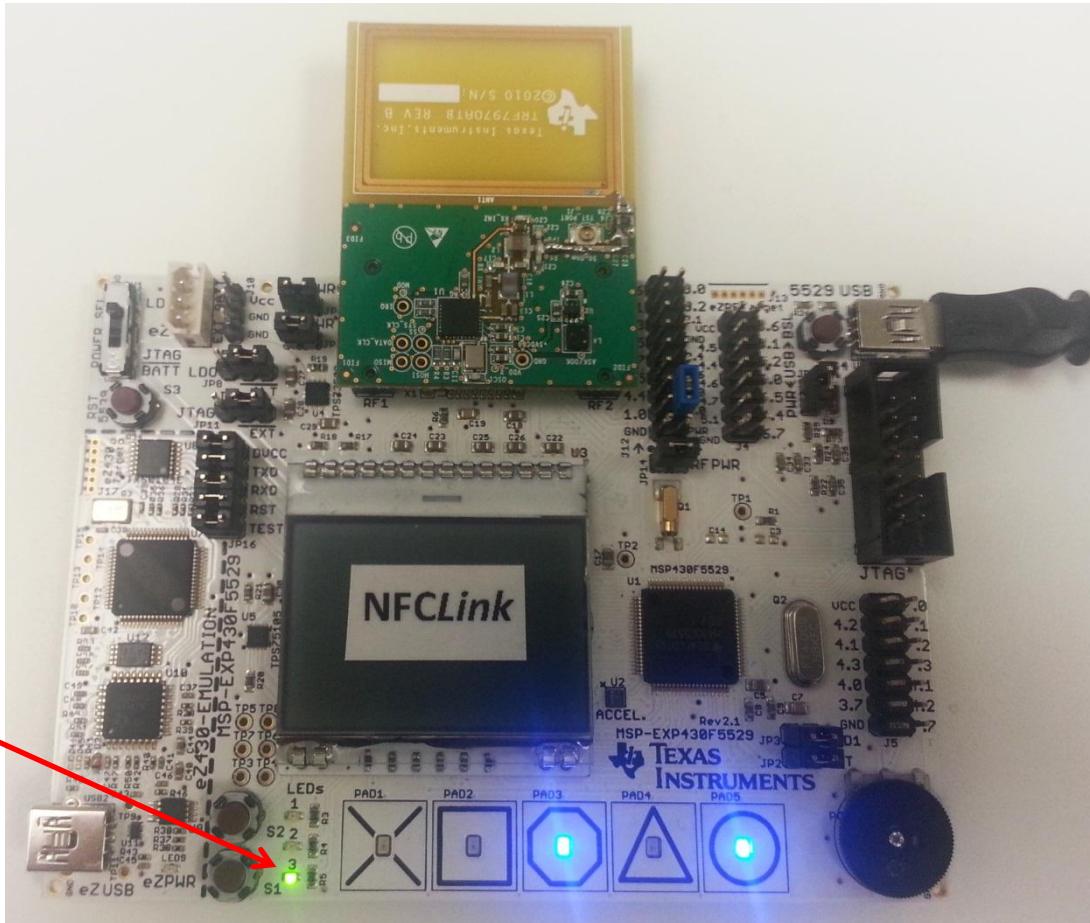
- For those wishing to use Card Emulation Mode, under the Host-based card emulation, choose the Mode and the Configuration, then press the Start CE button.
  - (CE: Start OK will appear in lower left hand window of the GUI)



11/25/2014

# Hardware LED status indicators (Card Emulation Mode, EXP board)

- When hardware is in card emulation mode, LED3 will be solid.



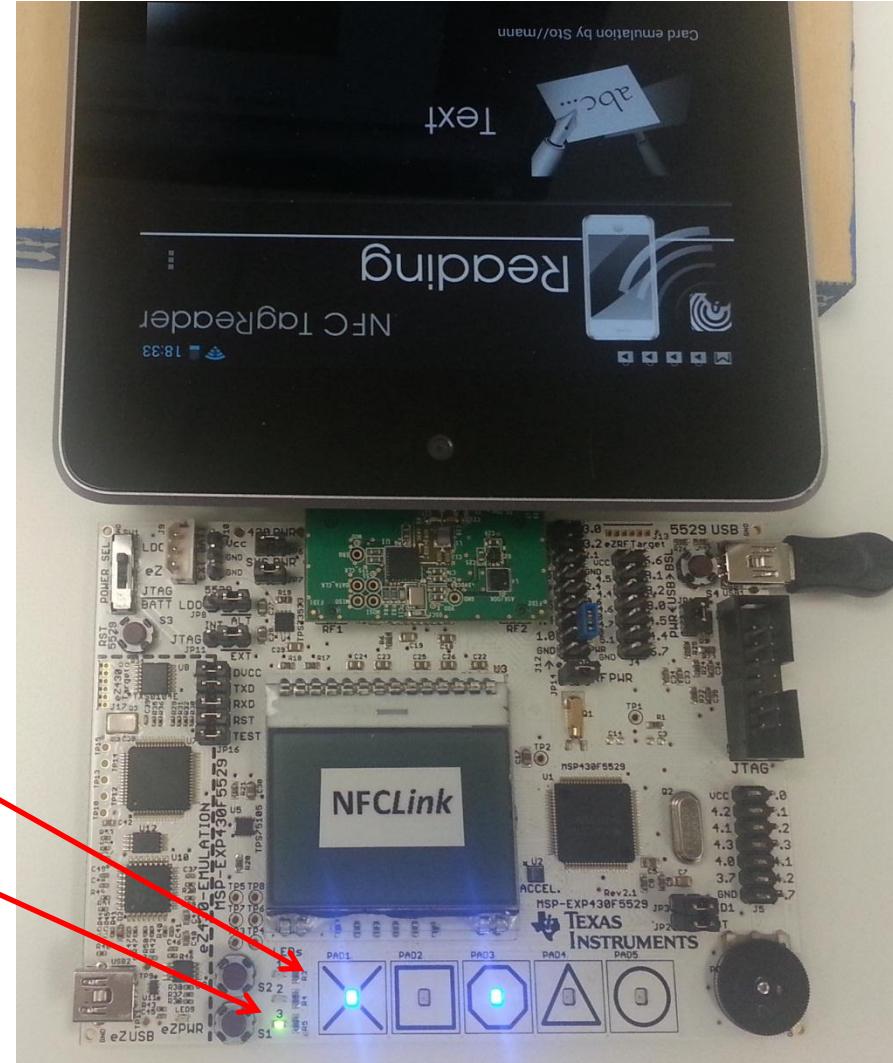
Card Emulation  
Mode

# Hardware LED status indicators (Card Emulation Mode, EXP board, cont.)

- When hardware is in card emulation mode, and NFC enabled device acting as reader/writer is presented, LED in center of PAD1 will illuminate.
- The PAD1 LED will flash as the device approaches, then when field strength is sufficient it will stay on solid.

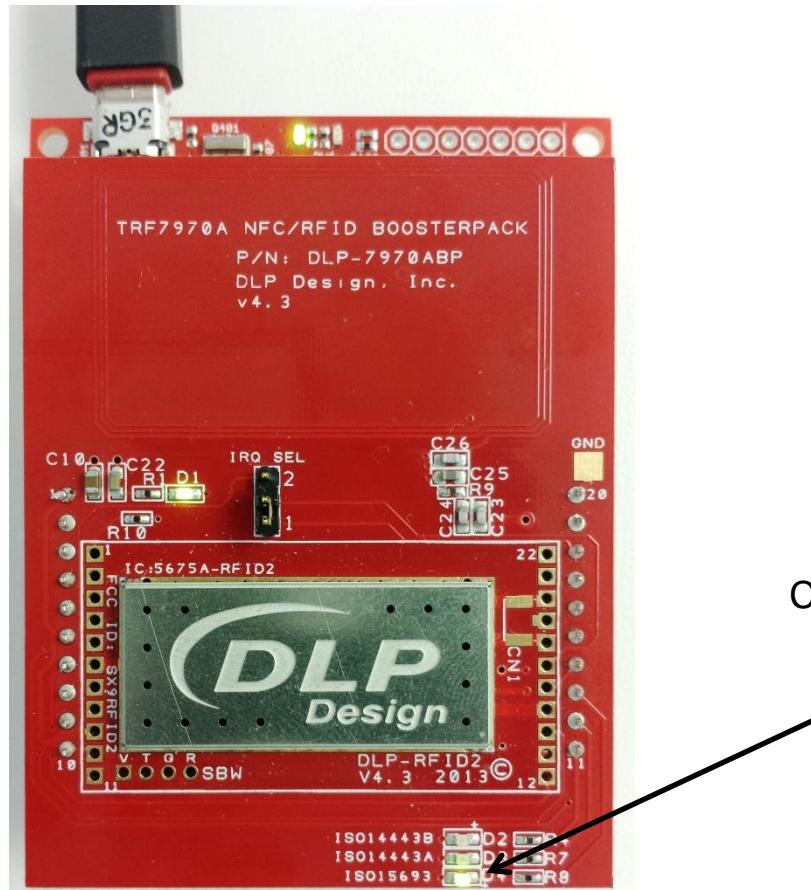
HF Field Present

Card Emulation Mode



# Hardware LED status indicators (Card Emulation Mode, EXP LP board)

- When hardware is in card emulation mode, ISO15693 LED will be solid.
- LED2 on EXP LP will come on when HF field is presented. (not shown)



Card Emulation  
Mode

# Card Emulation Mode (cont.)

- When an NFC handset (or other NFC enabled reader/writer) is presented, the hardware will be read out just like a passive NFC/RFID tag.
- Here we show various screen captures from handset application called NFC TagReader (from KDDI)



VCard



URI



text



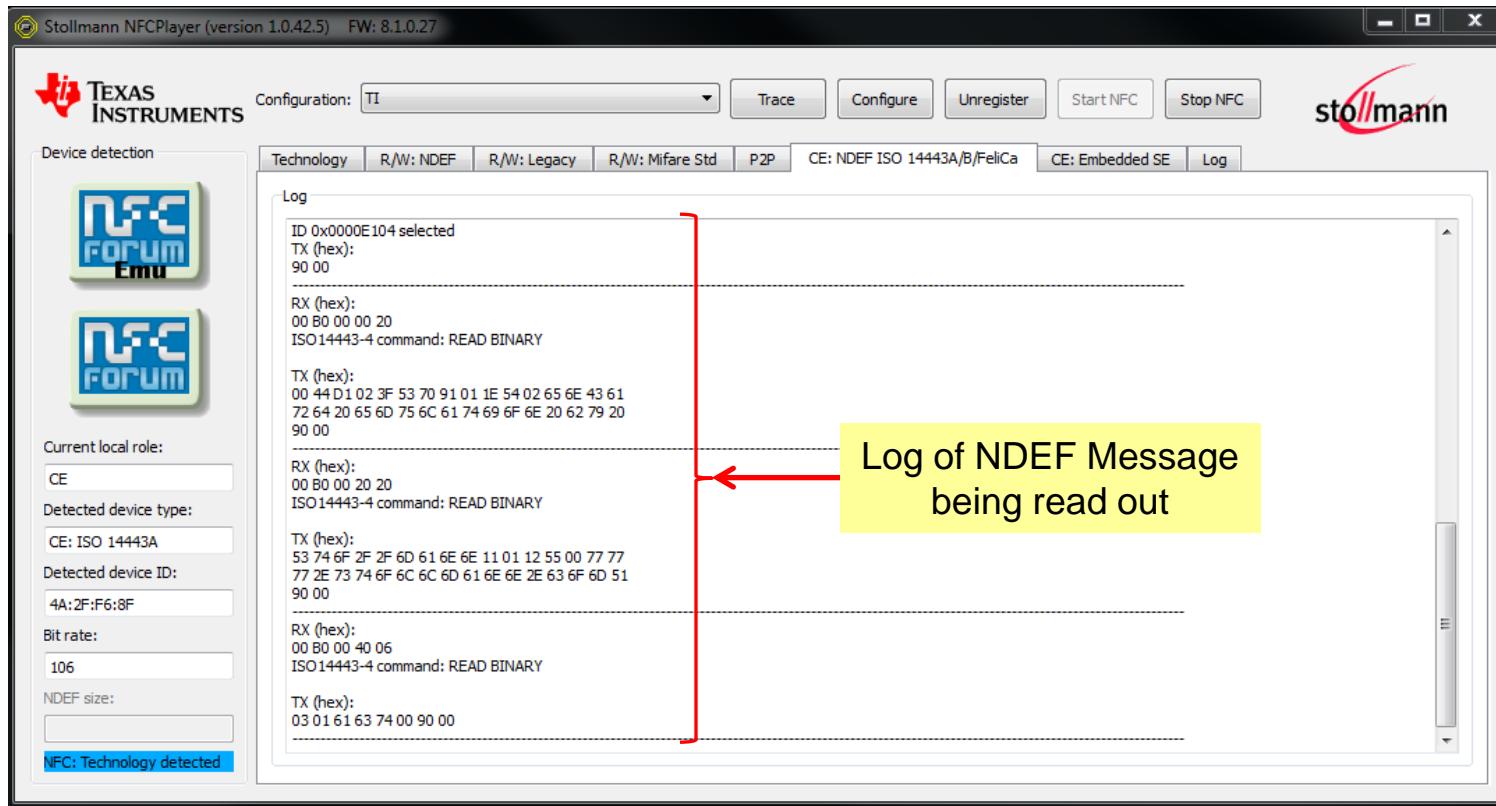
Smart Poster



Large text

# Card Emulation Mode (cont.)

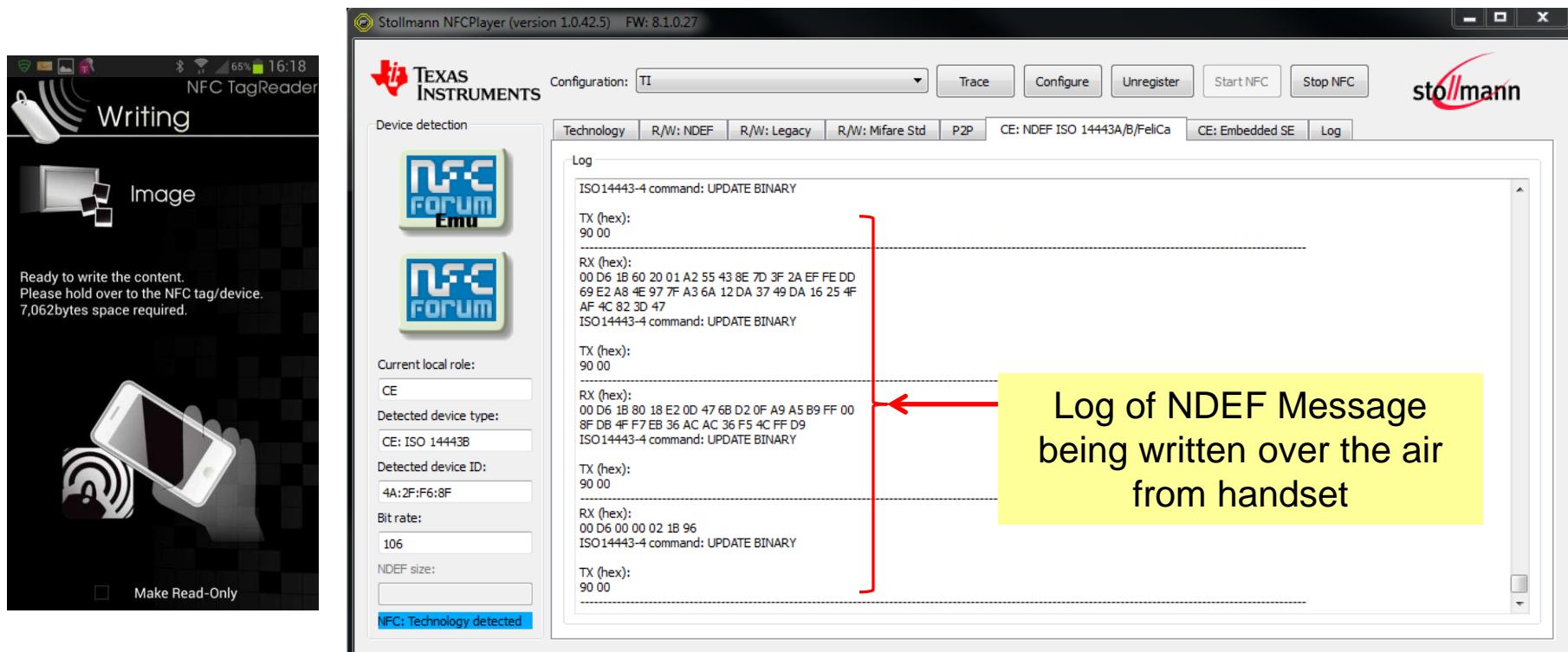
- When an NFC handset (or other NFC enabled reader/writer) is presented, the hardware will be read out just like a passive NFC/RFID tag.
- Here we show the GUI screen, as it is showing the packet activity between the NFC handset (or other reader/writer) and the hardware.



11/25/2014

# Card Emulation Mode (cont.)

- You can also use a handset application (like the KDDI one) to write an image (for example) to the hardware.



# Card Emulation Mode (cont.)

- Then of course the user can read back the image (or other data) from the hardware, using the application on the handset as well. (Master Yoda shown below in ~7kB form)



11/25/2014

# R/W & Card Emulation Modes used in conjunction with one another

- User can also set up two of the hardware sets and run two instances of the GUI on the same PC (with one as reader/writer and the other in card emulation mode) then read/write data without using an NFC handset. (Here we have written ~5.7kB image and have read it back)

The image displays two side-by-side screenshots of the Stollmann NFCPlayer software interface, version 1.0.42.5, running on a Windows operating system. Both screenshots show the 'Read' tab selected.

**Left Screenshot (R/W Mode):**

- Configuration:** TI
- Technology:** R/W: NDEF, R/W: Legacy, R/W: Mifare Std, P2P, CE: NDEF ISO 14443A/B/FeliCa, CE: Embedded SE, Log
- Device detection:** ISO 14443-B, NFC Forum
- Current local role:** RW
- Detected device type:** ISO 14443B (Type 4)
- Detected device ID:** 01:02:03:04
- Bit rate:** 106
- NDEF size:** 5758 of 8190
- R/W: Read NDEF OK**

**Right Screenshot (CE Mode):**

- Configuration:** TI
- Technology:** R/W: NDEF, R/W: Legacy, R/W: Mifare Std, P2P, CE: NDEF ISO 14443A/B/FeliCa, CE: Embedded SE, Log
- Device detection:** NFC Forum Emu
- Current local role:** CE
- Detected device type:** CE: ISO 14443B
- Detected device ID:** [redacted]
- Bit rate:** 106
- NDEF size:** [redacted]
- Log:** Shows a sequence of ISO 14443-4 commands for reading binary data, with RX and TX hex values and command details.

**Photograph of Hardware:**

A photograph shows two hardware boards connected via USB. The board on the left is a white breadboard with various components and a red LED. The board on the right is a blue breadboard labeled 'NFClink' with several blue LEDs. A red arrow points from the left screenshot to the left hardware board, and another red arrow points from the right screenshot to the right hardware board.

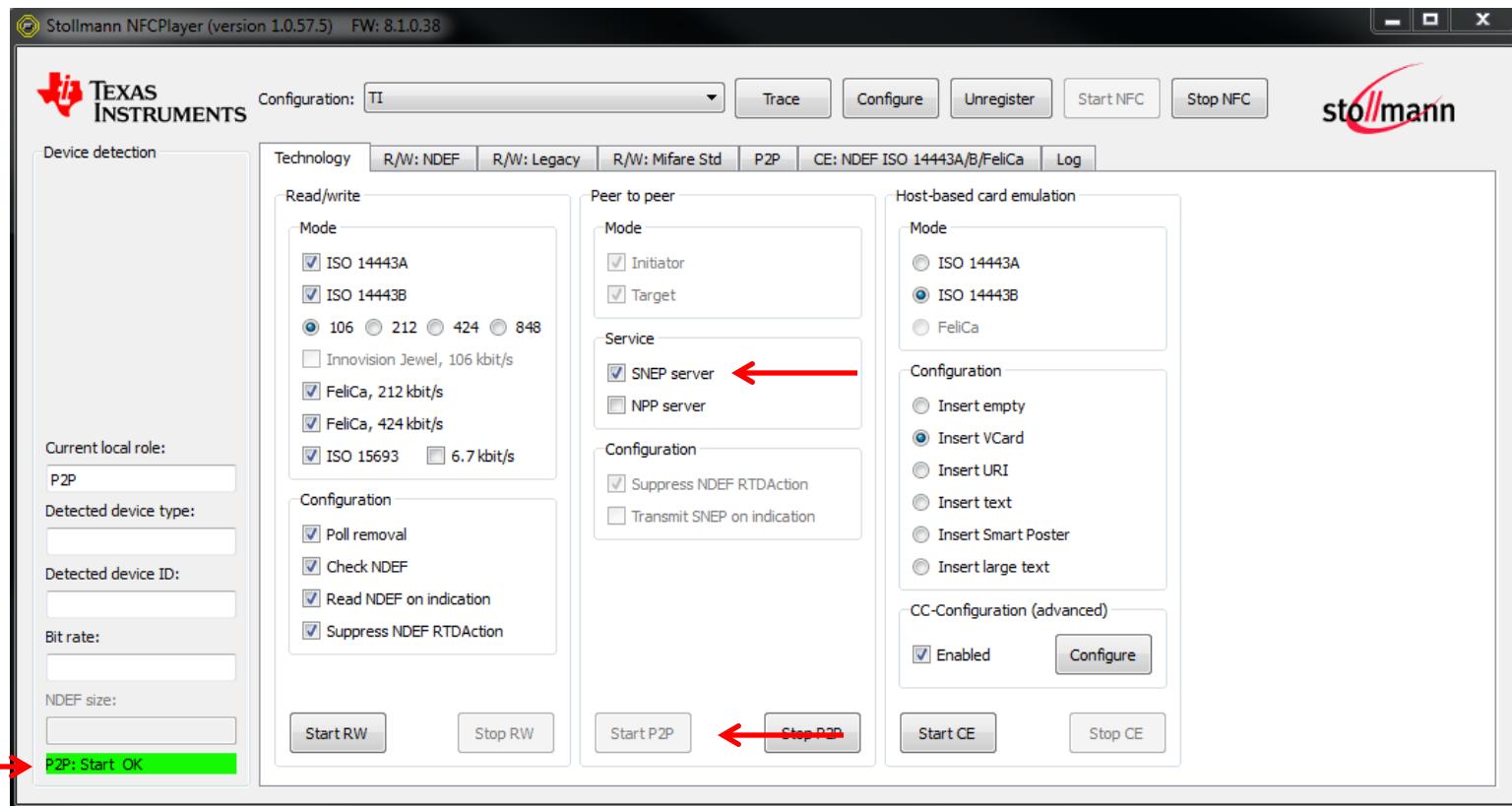
11/25/2014

# NFC/RFID P2P MODE

11/25/2014

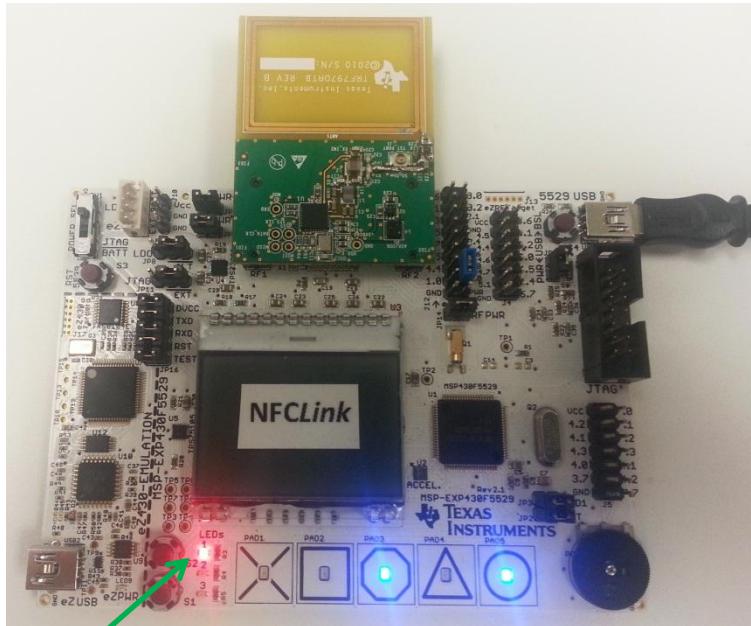
# Peer to Peer Mode (P2P)

- For those wishing to use P2P Mode, under the Peer to Peer column, choose the SNEP box, then press the Start P2P button.
  - (P2P: Start OK will appear in lower left hand window of the GUI)



# Hardware LED status indicators (P2P Mode, EXP board)

- When hardware is in P2P mode, LED1 & LED 2 will alternately be flashing.
- When NFC device is presented, the mode which is being used by the hardware (initiator or target) will go solid.
- After transfer is complete, the LEDs will go back to alternately flashing



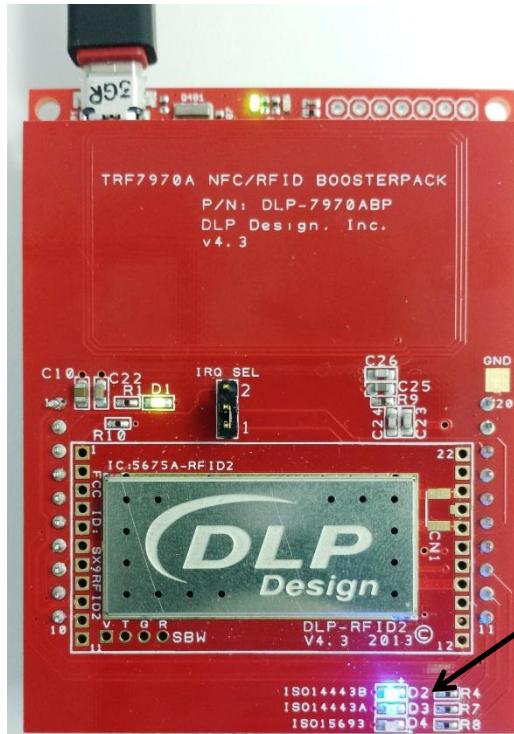
Reader/Initiator  
Mode



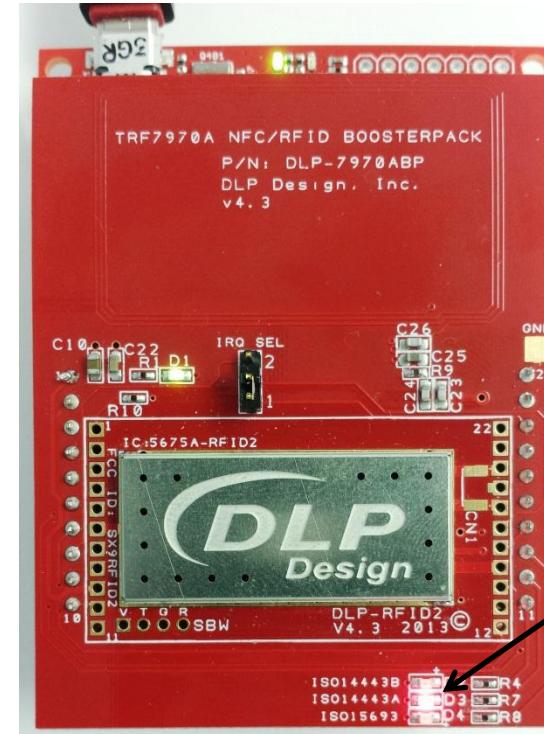
Target  
Mode

# Hardware LED status indicators (P2P Mode, EXP LP board)

- When hardware is in P2P mode, ISO14443B and ISO14443A LEDs will alternately be flashing.
- When NFC device is presented, the mode which is being used by the hardware (initiator or target) will go solid.
- After transfer is complete, the LEDs will go back to alternately flashing



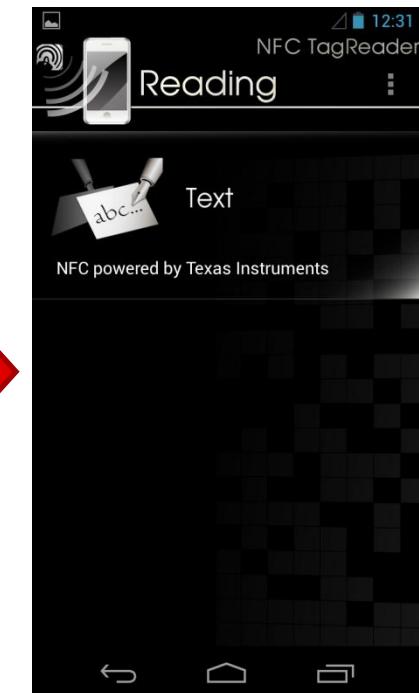
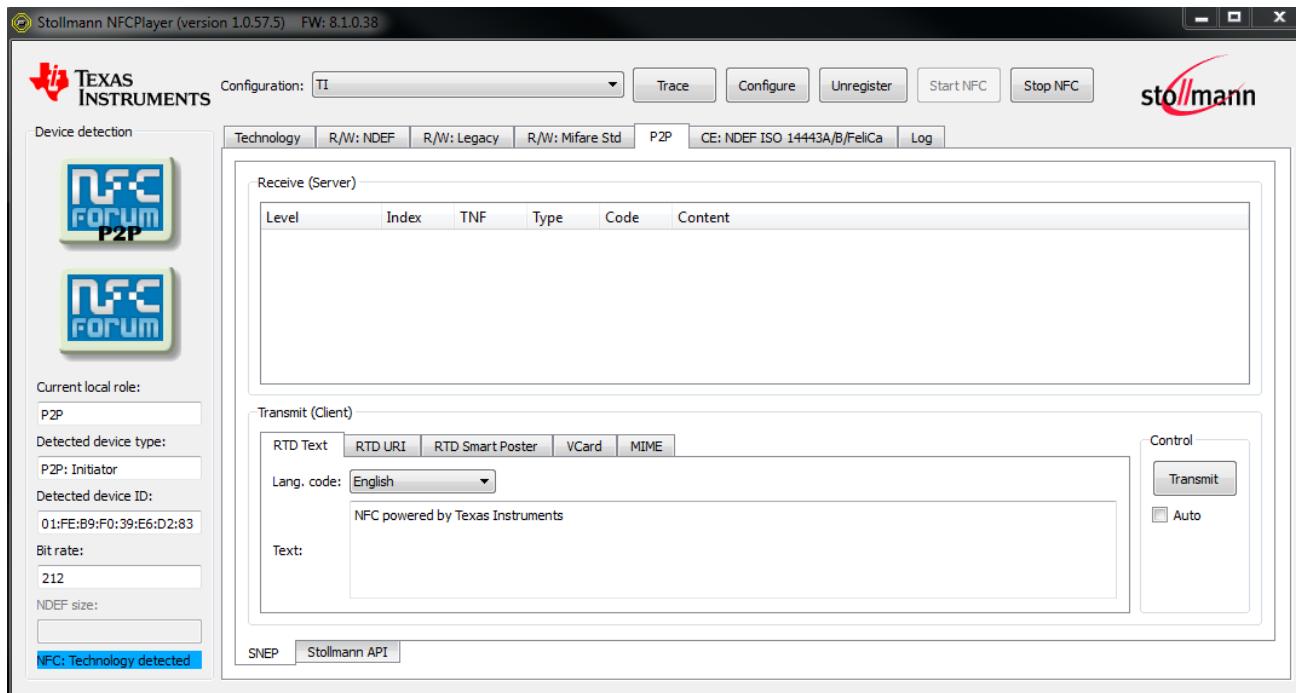
Reader/Initiator  
Mode



Target  
Mode

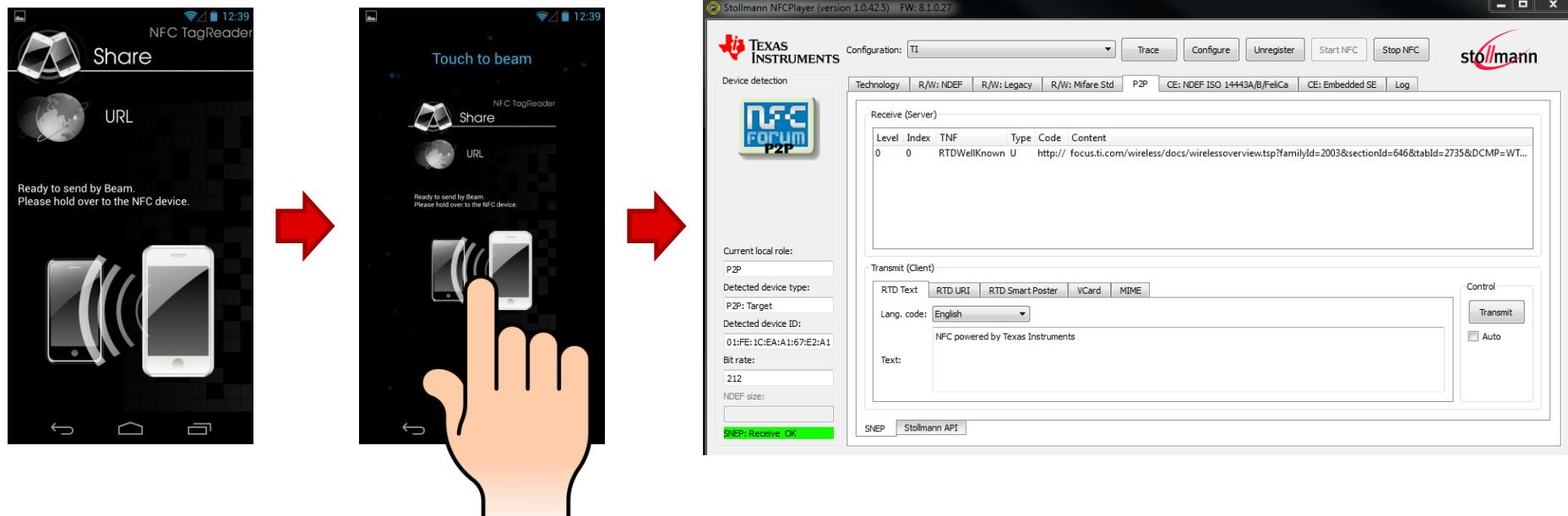
# Peer to Peer Mode (P2P) (cont.)

- When NFC Enabled device is presented (no app open in this case), the GUI screen flips over to P2P mode.
- Message can now be sent from GUI to the NFC enabled device using the Transmit button. This will open either the native app on the phone or a default one (NFC TagReader (from KDDI shown here))



# Peer to Peer Mode (P2P) (cont.)

- When NFC Enabled device is with an application open for doing P2P, the GUI screen flips over to P2P mode as before.
- Message can now be sent from the NFC enabled device to the GUI.
- Here we are “Beaming” a URL to a TI website, other message types can also be sent over too.
  - Alternative Radio Handovers, text content, SmartPoster, Phone Numbers, Applications, Images, etc.



# Peer to Peer between two NFCLink hardware sets

- User can also set up two of the hardware sets and run two instances of the GUI on the same PC, then demonstrate P2P functionality without using an NFC handset. (Here we have sent an RTD MIME message (in the form of an image) from one hardware set to another, using SNEP)

The image shows two side-by-side windows of the Stollmann NFCPlayer software. Both windows have a header bar with buttons for Configuration, Trace, Configure, Unregister, Start NFC, Stop NFC, and Log. The left window is titled "Stollmann NFCPlayer (version 1.057.5) FW: 8.1.0.38". It shows a "Receive (Server)" section with a table for Level, Index, TNF, Type, Code, and Content. Below it is a "Transmit (Client)" section with tabs for RTD Text, RTD URI, RTD Smart Poster, VCard, and MIME. The "MIME" tab is selected, and the URI field contains "C:/Users/a0867749/Pictures/NFC Images for transfer/130108162227.jpg". A "Select image" button and a "Control" panel with "Transmit" and "Auto" options are visible. The right window is also titled "Stollmann NFCPlayer (version 1.057.5) FW: 8.1.0.38". It shows a similar "Receive (Server)" section with a table for Level, Index, TNF, Type, Code, and Content. The "Content" column for the first row shows a thumbnail image of Earth. A red arrow points to this image with the text "Message received". Below it is a "Transmit (Client)" section with tabs for RTD Text, RTD URI, RTD Smart Poster, VCard, and MIME. The "MIME" tab is selected, and the Prefix field contains "http://www.". The URI field contains "stollmann.com". A "Control" panel with "Transmit" and "Auto" options is also present.

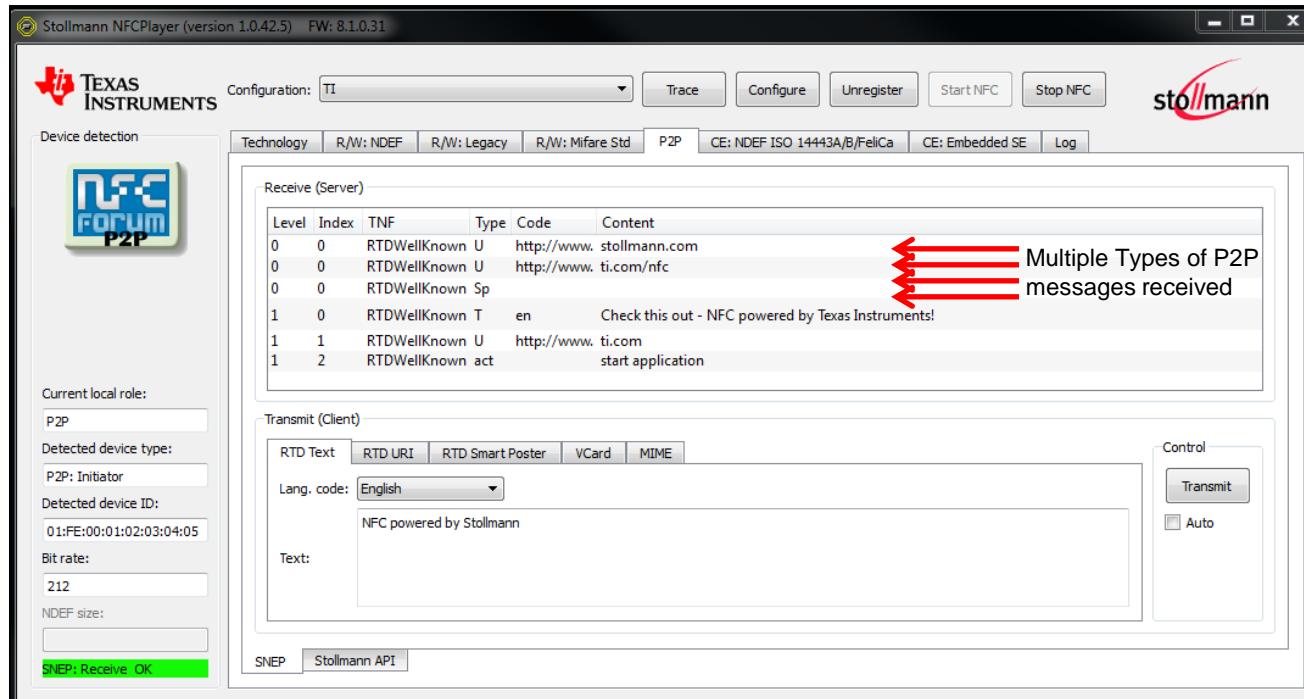
Hardware set in P2P mode



Hardware set in P2P mode

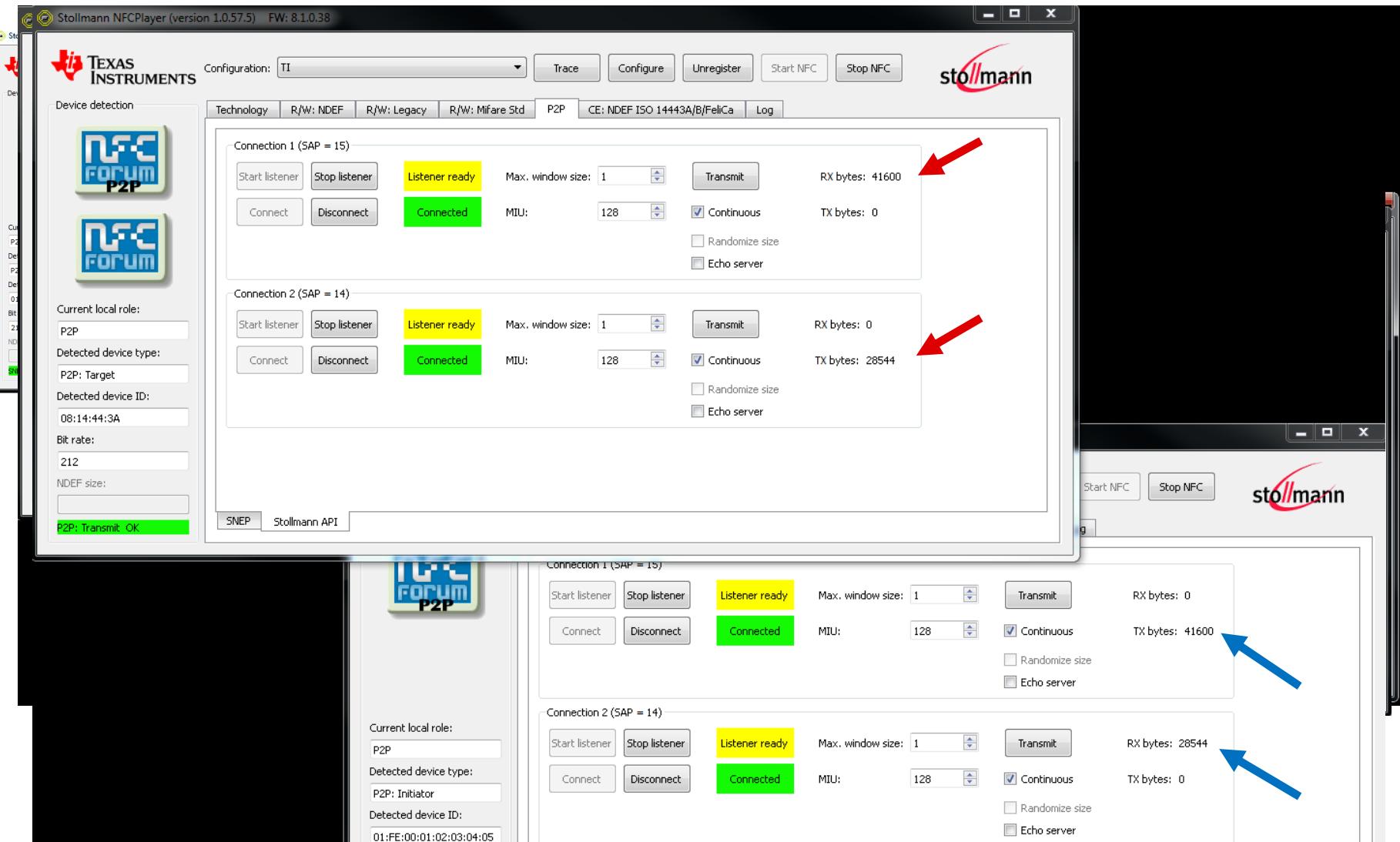
11/25/2014

# Peer to Peer between two NFCLink hardware sets (cont.)



11/25/2014

# Advanced Features with P2P



# NFCLINK FILE STRUCTURE

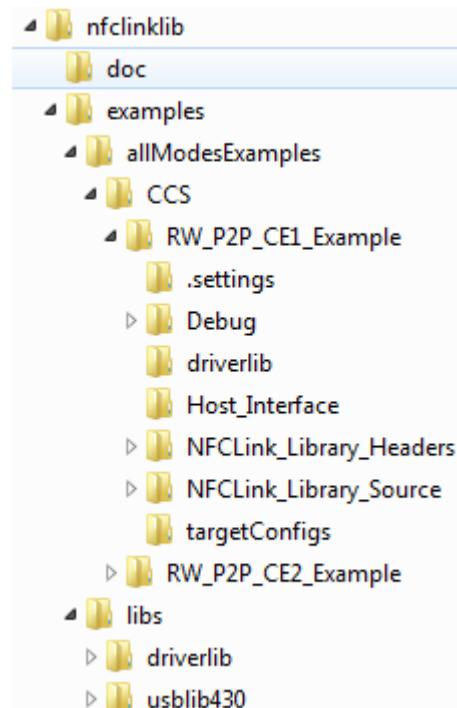
11/25/2014

# What is being installed?

The NFCLink installer will include the following folders:

1. **Docs** – User Guide and Quick Start Presentation.
2. **Examples/ allModesExamples** – Reader/Writer, Peer to Peer, and Card Emulation projects. (Only CCS for release)
  - a) **RW\_P2P\_CE1** – USB (CDC) interface to the host.
  - b) **RW\_P2P\_CE2** – UART Module interface to the host.
3. **Examples/Libs** – libraries used by the CCS projects.
  - a) **driverlib** - UART, GPIO and Timer drivers.
  - b) **usplib430** - USB drivers for the MSP430F5529.
4. **NFC\_Player\_installer** - host executable for Windows OS.  
The program will be installed in C:/Program Files(x86)/Stollmann.

**Note:** The folder structure complies with MSP430Ware structure, and will be part of it on the upcoming August release.



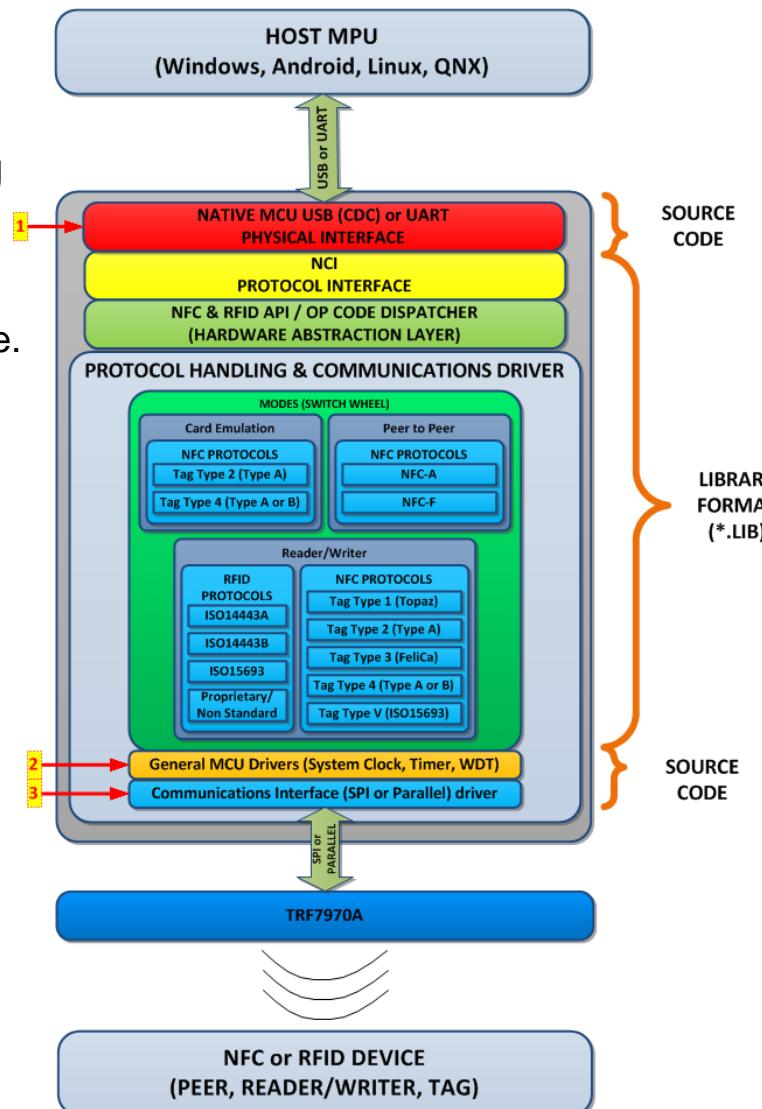
# PORTING TO OTHER MSP430s

11/25/2014

# MSP430s Porting Overview

The following layers will have to be modified when porting to other MSP430s:

- 1. Host Interface** – Code examples will include USB / UART. Future releases will include SPI and I2C implementations – i.e. using an Aardvark™ I<sup>2</sup>C/SPI Host Adapter for testing.
- 2. MSP430 Hardware** – Main application, MSP430 MCLK, Watch Dog Timer(WDT), GPIOs (for LEDs and debugging purposes), and one timer.
- 3. TRF7970A transceiver Interface** – SPI w/ Slave Select module. Future releases will include Parallel.



# Host Interface Modifications

There are **three** functions that **must** be modified to interface with the host.

## 1. Host\_Interface::Init() –

Initialization for the module and RX ISR. For RW\_P2P\_CE1 the function initializes and configures the USB module.

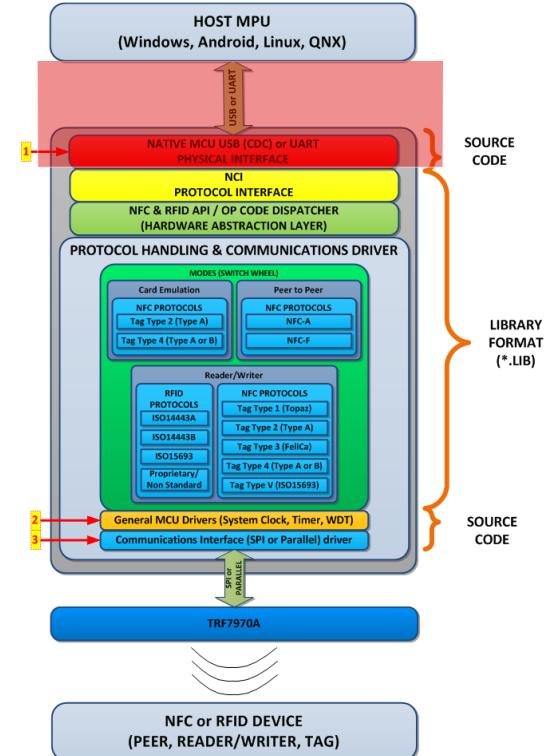
## 2. Host\_Interface::Write() –

Function that transmits to the host. For RW\_P2P\_CE2 the function writes to the UART\_TX buffer with len (length) bytes.

## 3. The RX ISR for the host interface –

```
BYTE USBCDC_handleDataReceived (BYTE intfNum) ← RW_P2P_CE1_Ex.  
_interrupt void USCI_A1_ISR(void) ← RW_P2P_CE2_Ex.
```

Store incoming bytes into Host\_Interface::recbuf.



**Note:** Include necessary files for the Host\_Interface module inside folder.

# General MSP430 Modifications

There are **five** functions that **must** be reviewed :

1. **HW\_Config::Init()** –

Disables the WDT. Sets up the Frequency of the MCLK – the current implementation uses the 32.768 kHz crystal (ACLK) , MCLK = DCO = 25MHz.

2. **HW\_Config::MCU\_Reset()** –

Reset the MSP430 by setting the BOR flag (this can be modified to a software power on reset depending on the MCU). When a host reset command is received, this function is used to reset the MCU.

3. **HWTimer::Init()** –

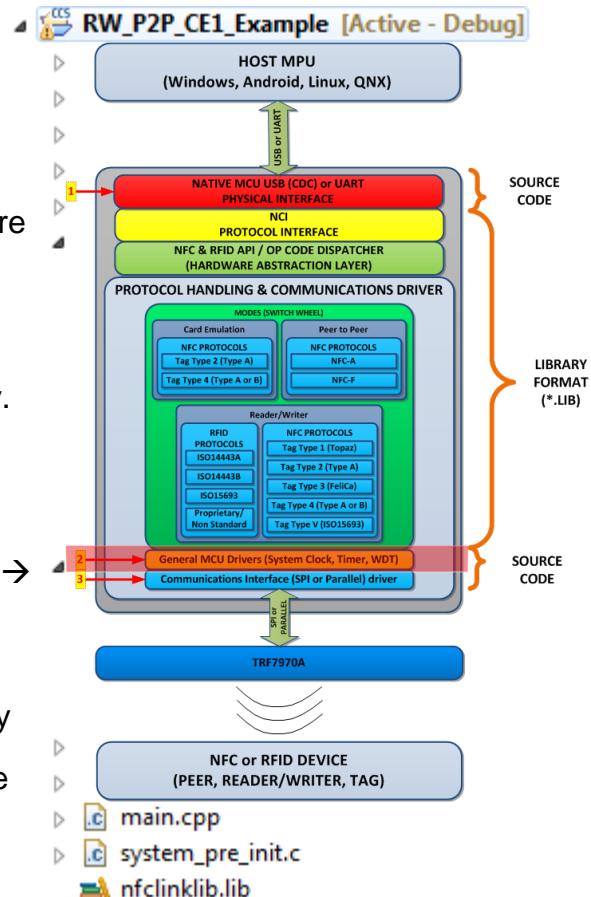
Initialize Timer A using reference of ACLK (32.768kHz) running continuously.

4. **HWTimer::ticks32()** –

Returns and stores the value of the timer's counter into timervalue.f[0]. Furthermore **TICKS\_PER\_MSEC** (inside `MSP430_hardware.h`) needs to be updated based on the CLK being used as reference. (i.e. For a 2MHz Clock →  $\text{TICKS\_PER\_MSEC} = \text{Ref. CLK} / 1000 = 2 \times 10^6 / 1000 = 2000$ )

5. **GPIO::Init()** –

Initializes the External Field LED (P1.1), External Field debug pin (P4.1), Any Mode LED (P1.3), RW/Initiator Mode LED (1.0), P2P LED (P8.1), CE (8.2), Serial TX debug pin(4.3) and Serial RX debug pin (4.2). These GPIOs will be **helpful** to provide feedback to our team.



# TRF7970A Interface Modifications

There are **five** functions that **must** be reviewed:

## 1. TRF797x\_setup() –

Initializes the interface to the TRF7970A (SPI / Parallel).

Initializes the TRF7970 EN pin, then sets up the IRQ pin with a rising edge interrupt. Afterwards, it writes to the TRF7970 to ensure it has been initialized properly. (RFID.cpp/h)

**Note:** For this release only SPI w/ SS will be supported.

## 2. RFSPI::init() –

Initializes the SPI module as 3 pin SPI, 8 bit Master, MSB, Clock Pol/ Phase = 0, SPI Clock ~ 4MHz using the SMCLK = 25MHz as reference. The Slave Select will be manually set – ensure that SLAVESELECT\_OUTPUT is using the correct GPIO (SPI.cpp/h)

**Note:** Please see the datasheet to match

## 3. TRF797x\_IRQHandler() –

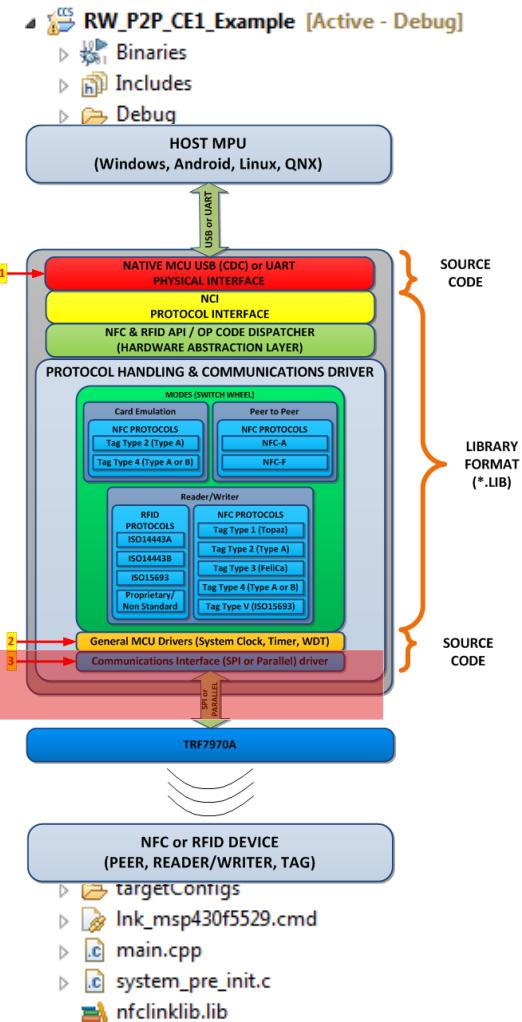
Ensure that the interrupt service routine for the IRQ pin (rising edge interrupt) is setup correctly. (RFID.cpp)

## 4. RFSPI::waitForBus() –

Waits for the SPI module to be idle. (Needs to be modified depending on the USCI) (SPI.cpp)

## 5. RFSPI::transfer() –

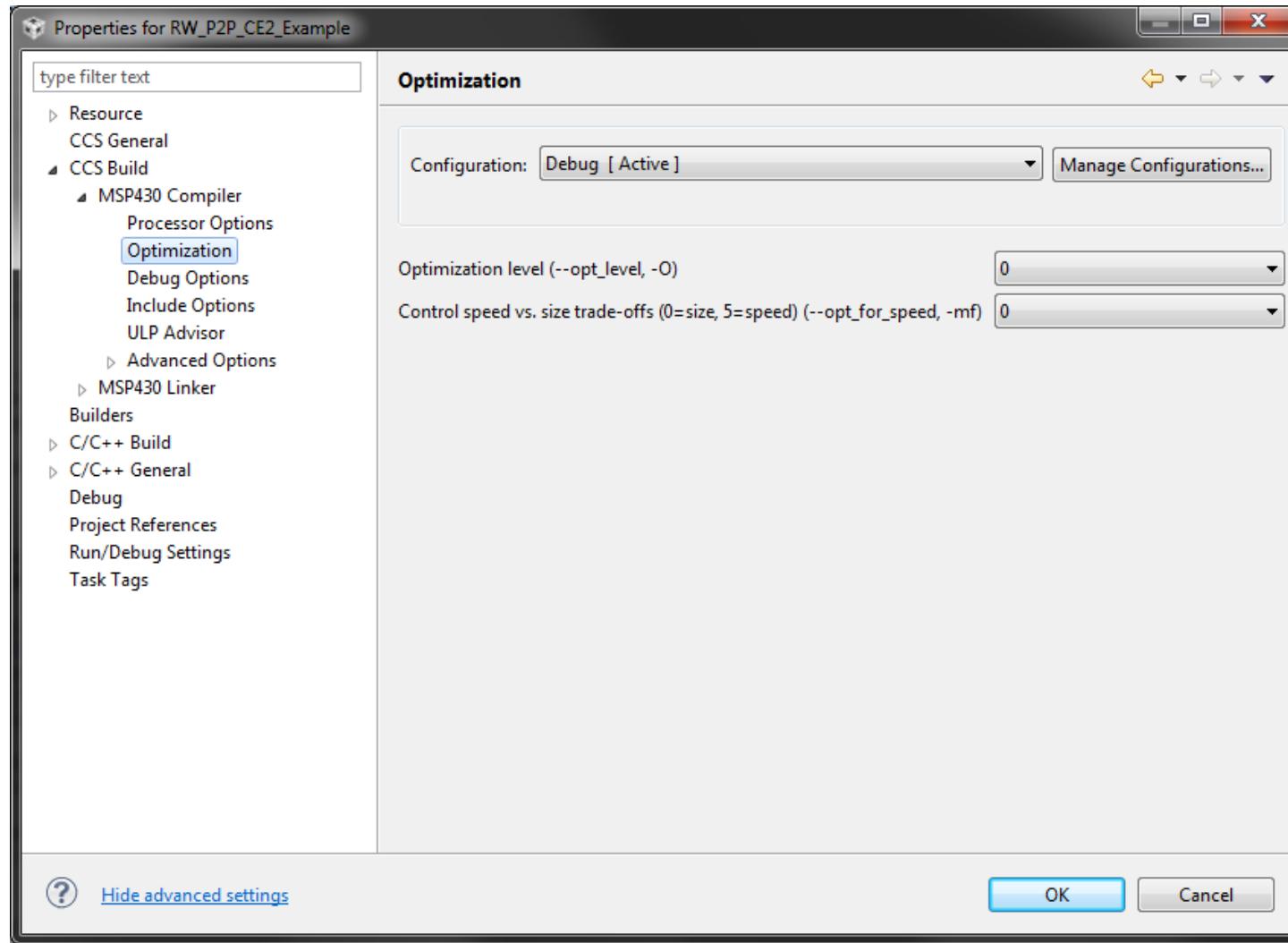
Writes to the SPI TX register, and returns the value of the SPI RX register. (Ensure USPITXBUF and USPIRXBUF are defined appropriately). (SPI.h)



# MEMORY FOOTPRINTS

11/25/2014

# CCS Optimization Options



# NFCLink CCS Memory Footprint Overview

Mode + USB CDC (CCS Project stand alone w/o library)	Optimization Level (0 – 4)	Flash (kB)	RAM (kB)
All Modes	0	113.2	5.7
	4	104.6	5.7

**Note :** The control speed vs. size trade offs (0 = size, 5 = speed) for all the memory footprints was set to 0 – optimized for smallest footprint (**size**) .

**Note 2:** The memory footprints listed above are preliminary.

# SUMMARY

11/25/2014

# Summary

- NFCLink Overview
- NFCStack + Evaluation Overview
- NFCLink Components
- NFCLink File Structure
- To download NFCLink SW: [www.ti.com/nfclink](http://www.ti.com/nfclink)
- For more information on NFC: [www.ti.com/nfc](http://www.ti.com/nfc)

# Thank you

# Backup

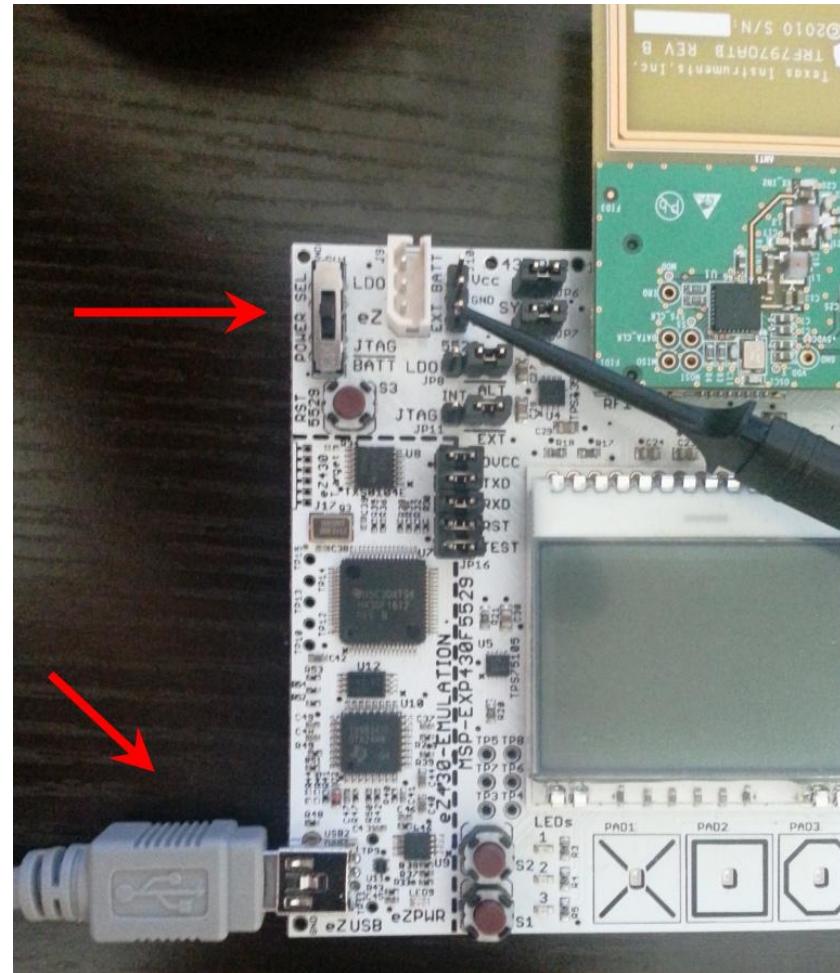
11/25/2014

# Download Provided Firmware Image to MSP-EXP430F5529 board

- Download, Unzip and Install MSP430 Flash Programming Tool from Elprotronic website
- <http://www.elprotronic.com/files/FET-Pro430-Setup.zip>
- Connect USB-A end of USB cable to PC

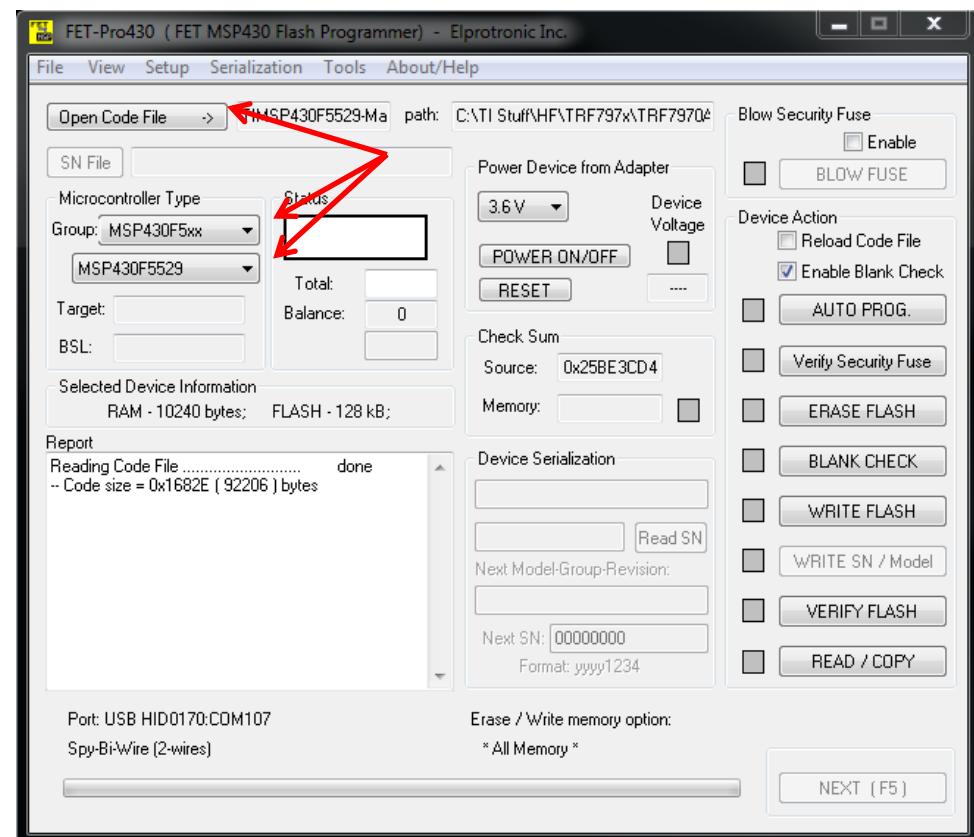
# Download Provided Firmware Image to MSP-EXP430F5529 board (cont.)

- Place POWER SEL switch on MSP-EXP430F5529 board (upper left hand side of the board) in the eZ position (middle)
- Connect mini-USB cable end of USB cable to ezUSB connector on MSP-EXP430F5529 board. (bottom left side of board)
- Open Elprotronic FET-Pro430 tool Graphical User Interface (GUI)



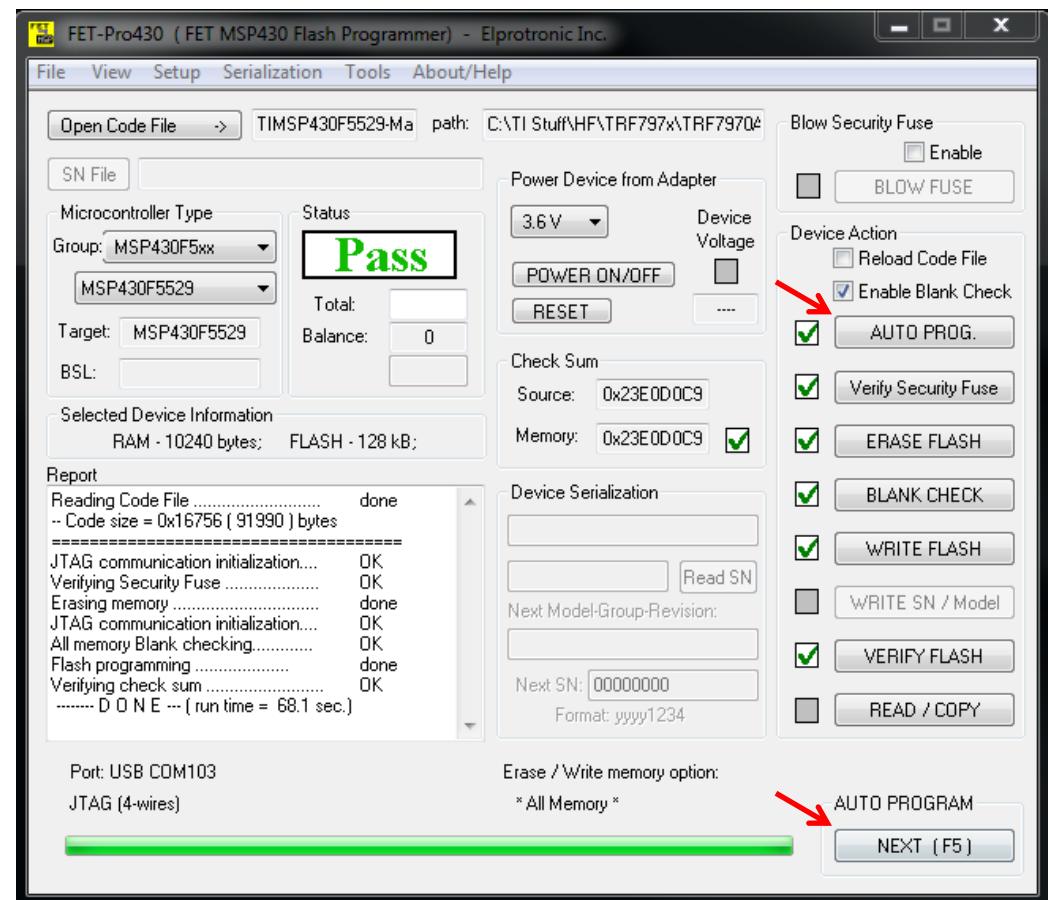
# Download Provided Firmware Image to MSP-EXP430F5529 board (cont.)

- In upper left corner of the GUI, press Open Code File Button and navigate to where provided FW image is stored on PC and select it.
  - Latest image version is:  
**TIMSP430F5529-May\_29\_Ver\_0x0801001F.hex**
- Choose Microcontroller Type Group: (using dropdown)  
MSP430F5xx
- Choose (using dropdown)  
MSP430F5529



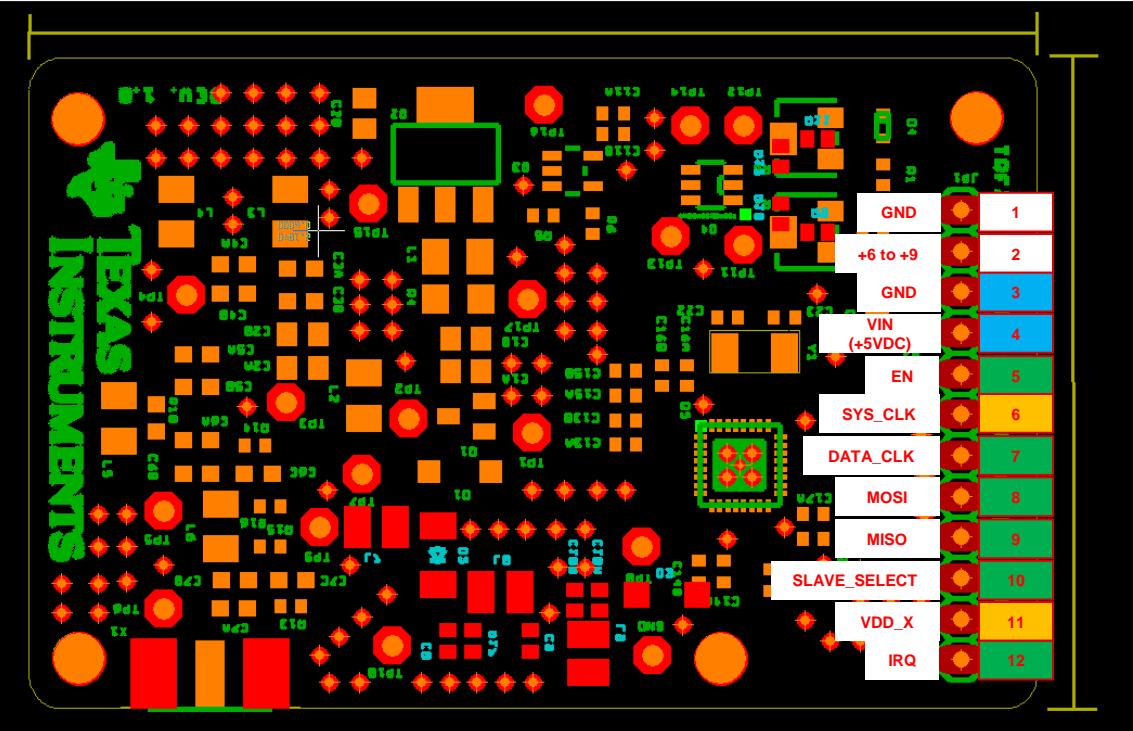
# Download Provided Firmware Image to MSP-EXP430F5529 board (cont.)

- Next, press the AUTO PROG. button in the Elprotronic tool and allow the tool to complete the steps.
- NOTE: If more than one board is to be programmed at this time, the NEXT (or F5) button can be used to program additional units instead repeating all the steps on previous slide.

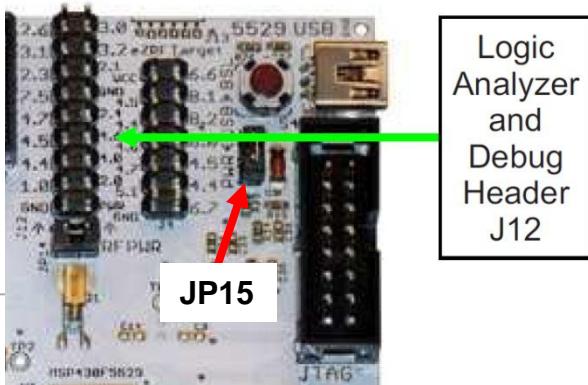


# NFCLINK + EMVCO AMPLIFIER EXAMPLE

# NFCLink MCU Board to TRF7970A EMVCo Amplifier Hardware Connections



OTHER CONNECTION	MSP-EXP430F5529 HDR J12	SIGNAL NAME	TRF7970A AMP BOARD JP1
EXT. P/S	N/C	GND	PIN 1
EXT. P/S	N/C	+6VDC TO +9VDC	PIN 2
	GND	GND	PIN 3
JP15 (-F5529 BD.)	N/C	VIN (+5VDC)	PIN 4
	P2.3	EN	PIN 5
	N/C	SYS_CLK	PIN 6
	P3.2	DATA_CLK	PIN 7
	P3.0	MOSI	PIN 8
	P3.1	MISO	PIN 9
	P2.6	SLAVE_SELECT	PIN 10
	N/C	VDD_X	PIN 11
	P2.0	IRQ	PIN 12



# References

- ISO/IEC15693-3
- ISO/IEC14443-3, ISO/IEC14443-4
- ISO/IEC7816-4
- ISO/IEC18092
- NFC Forum Specifications
  - <http://www.nfc-forum.org/specs/>
- MSP-EXP430F5529 Users Guide
  - <http://www.ti.com/lit/pdf/slau330>  
(schematic available in Section 5.3)
- TRF7970A Data Sheet
  - <http://www.ti.com/product/trf7970a>