Training Mobile Knowledge

NTAG I^{2}C plus – Your entryway to NFC
Product support package

Public
MobileKnowledge
March 2016

Agenda

Session 9th March: NTAG I2C plus – Your entryway to NFC

► *Introduction, use cases, target markets and benefits*

► *Positioning and NTAG portfolio*

► *Memory map*

► *Key functionalities*

► *Demokit and PSP quick overview*

► *Ordering details and wrap up*

Watch recording

**Session 16th March:** *NTAG I2C plus – Product Support Package*

► Where to find NTAG I2C *plus* documentation

► NTAG I2C *plus* Demo Kit

► How to evaluate NTAG I2C *plus*

► How to develop applications around NTAG I2C *plus* from I2C and RF sides

NTAG I²C *plus* is the simplest, most cost-effective NFC solution

The simplest, most cost-effective NFC solution

- Easy access to data from both RF (Type 2 tag) and from I²C
- Field detection to wake up connected devices
- Energy Harvesting capabilities
- EEPROM for offline data access
- Maximum interoperability with NFC devices
- Flexible memory management
- Originality signature for protection against cloning
- Fast & convenient data exchange via a 64 bytes SRAM buffer
- Small footprint package (1,6*1,6*0,5mm)
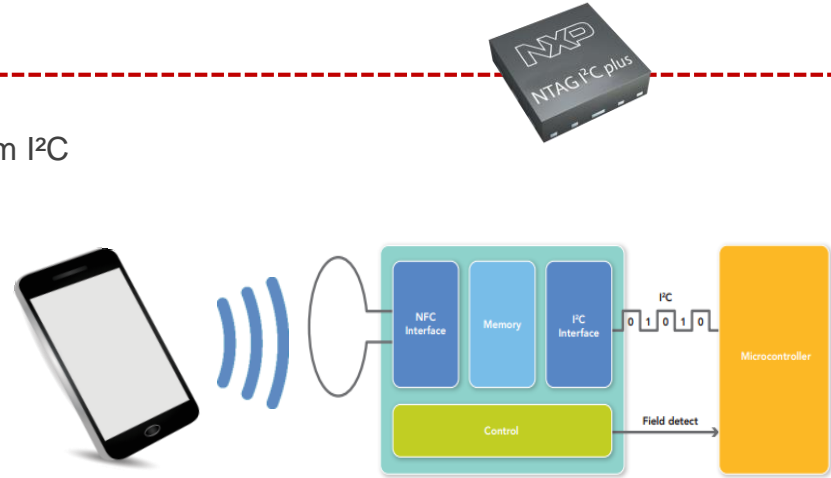
NTAG I^{2}C *plus* key features

Field detection
- Configurable field-detection output signal for data transfer synchronization and device wake up

Energy harvesting
- Energy harvesting from the NFC field to power external devices

SRAM memory
- 64 bytes SRAM volatile memory without write endurance limitation

SRAM mirroring
- Mapping the SRAM memory into the user memory

Pass through mode
- For fast data exchange of large files via the SRAM buffer

Originality signature
- Originality signature based on ECC for simple genuine authentication

NFC silence mode
- NFC silence to disable NFC interface

Memory access management
- Memory access configuration from both the NFC and I^{2}C interfaces.
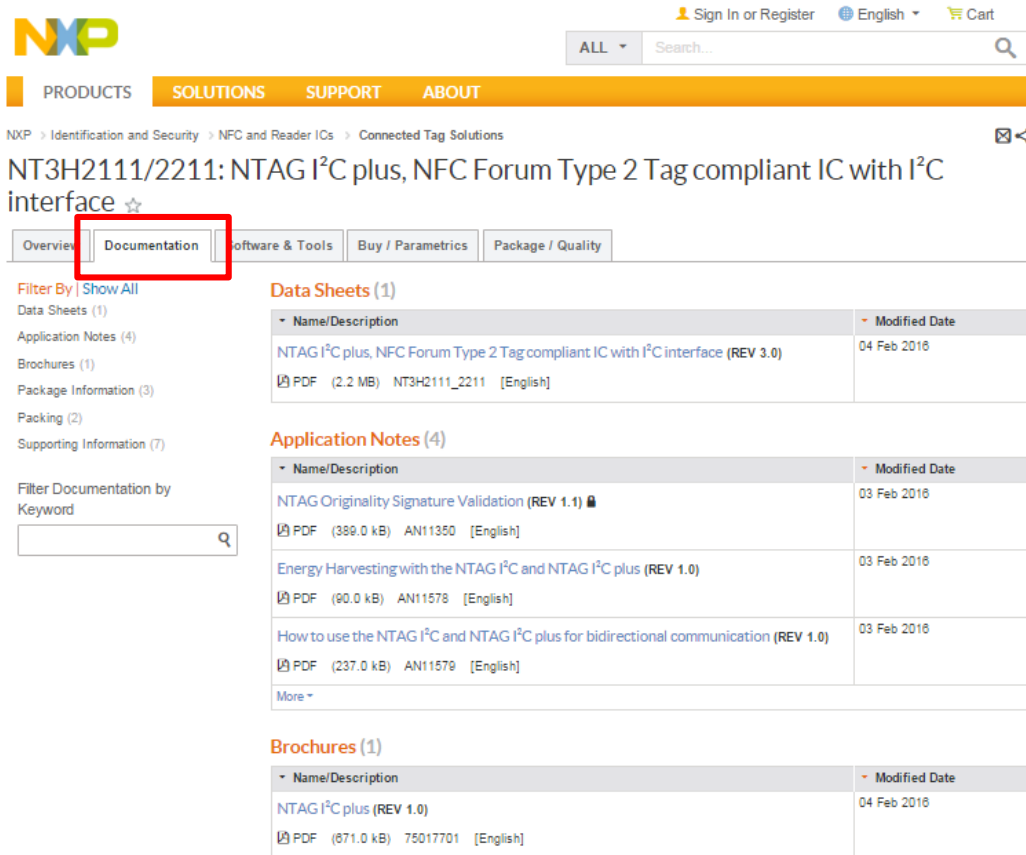
Where to find NTAG I^{2}C *plus* documentation

NXP Website → NFC → NTAG I^{2}C *plus*

NTAG I^{2}C *plus* → *Documentation, Software and Tools*

Software & Tools:
- Explorer Kit **without reader** → *Documentation, Downloads*
- Explorer Kit **with** reader → *Documentation, Downloads*

Where to find NTAG I²C *plus* documentation
NTAG I²C *plus* site

Where to find NTAG I²C *plus* documentation
NTAG I²C *plus* site

Where to find NTAG I2C *plus* documentation

Where to find NTAG I²C *plus* documentation
Explorer kit without reader

Where to find NTAG I²C *plus* documentation
Explorer kit without reader

Where to find NTAG I²C *plus* documentation
Explorer kit without reader

Where to find NTAG I2C *plus* documentation

```
┌──────────────┐      ┌──────────────┐      ┌──────────────┐
│     NXP      │ ───▶ │     NFC      │ ───▶ │    NTAG      │
│   Website    │      │              │      │   I2C plus   │
└──────────────┘      └──────────────┘      └──────────────┘
```

Explorer Kit without reader → *Documentation, Downloads*

Software & Tools

Documentation, Software and Tools

Explorer Kit with reader → *Documentation, Downloads*

Where to find NTAG I²C *plus* documentation
Explorer kit with reader

Where to find NTAG I²C *plus* documentation
Explorer kit with reader

Where to find NTAG I²C *plus* documentation
Explorer kit with reader

NTAG I²C *plus*

Demo Kit

NTAG I²C *plus* Demo Kit

Demokit	NTAG I²C plus Demo Kit
Firmware	SW3647 – Explorer Board firmware C Source files
Peek and Poke	SW3652 – Peek & Poke
Android Application	https://play.google.com/store/apps/details?id=com.nxp.ntagi2cdemo&hl=de
Windows Application	SW3651 - Setup of PC Windows application for use with Identiv Reader
Documentation	NT3H2111/NT3H2211 NTAG I²C plus - Product datasheet AN11578 - Energy Harvesting with the NTAG I²C and NTAG I²C plus AN11579 - How to use the NTAG I²C and NTAG I²C plus for bi-directional NTAG I²C plus AN11350 - NTAG Originality Signature Validation AN11276 - Antenna design guide AN11786 - NTAG I²C *plus* memory configuration options UM3587 - NTAG I²C Explorer Peek and Poke GUI UM3587 - NTAG I²C Explorer Demo app UM10945 - NTAG I²C plus Explorer Kit Program and Debug Start-up UM10966 - NTAG I²C plus Explorer Kit and Android Demo

NTAG I²C *plus* Demo Kit

OM5569/NT322E includes:

- ► Explorer Board
- ► NXP NT3H2211 NTAG I²C plus
- ► Field detector board
- ► 10 NTAG I²C *plus* samples
- ► Flex antenna

OM5569/NT322ER includes an external USB reader

OM5569/NT322F includes an additional set of flex antennas

NTAG I²C *plus* Demo Kit

About the Explorer Board:

- ► 5 buttons (3 Color buttons, RESET and ISP)
- ► NXP LM75B digital temperature sensor
- ► NXP LPC11U24 32-bit ARM Cortex-M0 microcontroller
- ► thermal watchdog
- ► RGB light
- ► LCD display
- ► microUSB for PC connection
- ► JTAG for FW upload
- ► I2C Bus Connector

NTAG I²C *plus*

How to evaluate

Software Tools to evaluate NTAG I^{2}C *plus*

Android Application

Explorer
Demo kit
Applications

Pc Windows Application
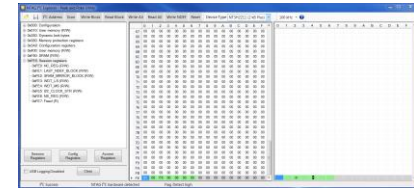
Peek and Poke

Software Tools to evaluate NTAG I^{2}C *plus*

Explorer
Demo kit
Applications

Android Application

Pc Windows Application

Peek and Poke
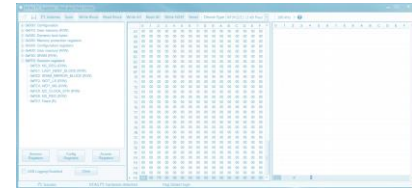
NTAG I²C *plus* Explorer Kit Demo Application

▶ The functionalities of the Android and Windows application are the same.

▶ Shows the I²C bi-directional communication.

▶ Read/Write NDEF to EEPROM.

▶ Speed tests of the SRAM and EEPROM communication.

▶ Read/Write registers.

▶ Read all memory.

▶ Upload new firmware.
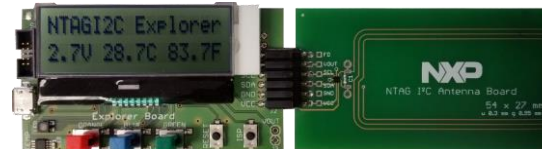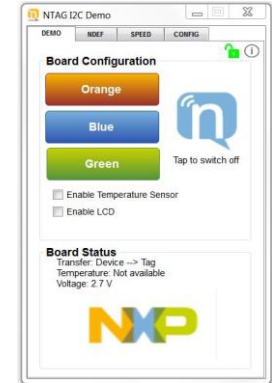
NTAG I²C *plus* Explorer Kit Demo Application
Led Demo Tab

Led Color Selection

Temperature and LCD enabling selection

Display Temperature of the sensor and output voltage on the Vout pin

Show board button pressed.

NTAG I²C *plus* Explorer Kit Demo Application
NDEF Tab

Read NDEF

Show read message

Enable a Reading NDEF message loop.

Write NDEF message

NDEF message to be written on Tag.

Enable AAR Selection Box

Write a default NDEF message

Performance

NTAG I²C *plus* Explorer Kit Demo Application
Speed Tab

There are two types of tests
- ► SRAM Test
- ► EEPROM Test

SRAM Tests

- ► Measure the data transfer speed from/to the application to/from the microcontroller through the SRAM.
- ► The application writes data to the SRAM several times, and the microcontroller reads from it.
- ► NTAG I²C PLUS in Pass-Through mode

EEPROM Tests

- ► It measures the speed at which the application is able to write and read the EEPROM memory of the NTAG I²C PLUS.
- ► An NDEF message is written to the EEPROM, and then, the same NDEF message is read.

Configuration of the Speed Test

Performance
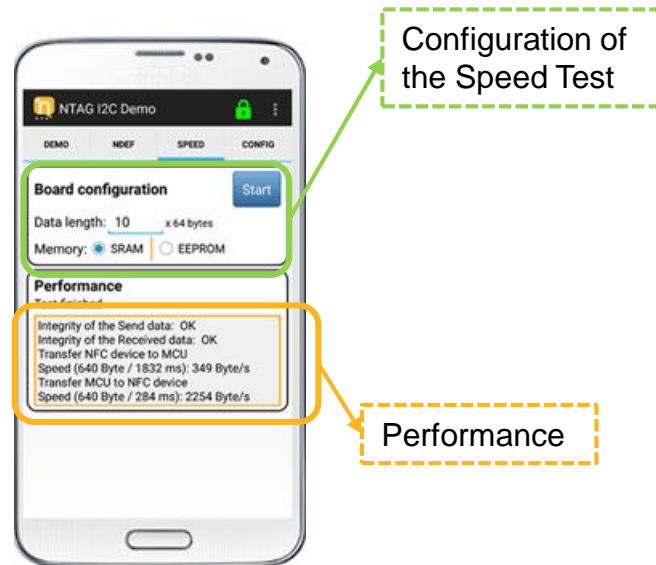
NTAG I²C *plus* Explorer Kit Demo Application
Config Tab

Read Tag Memory

► It shows the whole memory of the NTAG I²C PLUS product.

► The pages from 0xE2h to 0xFFh in the sector 0 are hidden for security reasons (PWD, PACK, AUTH0, PT_I2C and ACCESS bytes).

Read Tag Memory

Reset Tag

► It sets the original content programmed by production.

► The access configuration is also set to default: PWD to 0xFFFFFFFF, AUTH0 to 0xFF, PACK to 0x0000, ACCESS to 0x00 and PT_I2C to 0x00.

Reset Tag

Read Session Registers

Read/Write Config Registers

Read Session Registers

► It shows the value of the session registers. They are read only from NFC interface.

Read/Write Config Registers

► It shows the configuration registers and allows the user to write them.

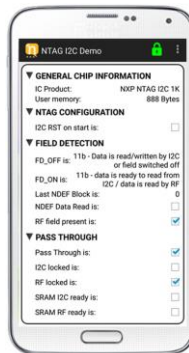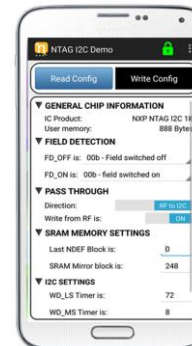# NTAG I²C *plus* Explorer Kit Demo Application
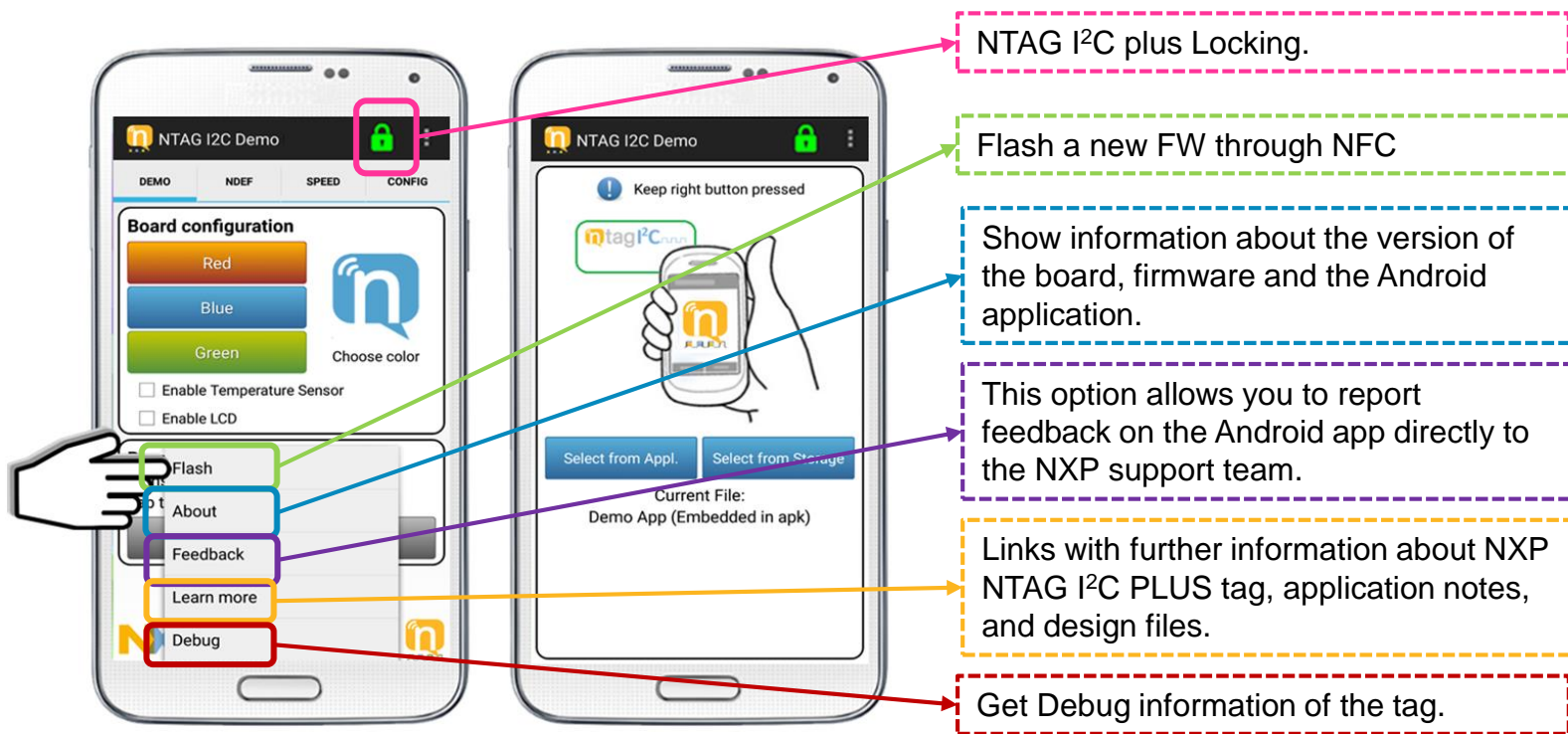## Config Tab

Read Tag Memory

Reset Tag

Read Session Registers

Read/Write Config Registers

NTAG I²C *plus* Explorer Kit Demo Application
Action Bar

NTAG I²C plus Locking.

Flash a new FW through NFC

Show information about the version of the board, firmware and the Android application.

This option allows you to report feedback on the Android app directly to the NXP support team.

Links with further information about NXP NTAG I²C PLUS tag, application notes, and design files.

Get Debug information of the tag.

NTAG I^{2}C *plus* Explorer Kit Demo Application
Authentication

► The NTAG I^{2}C *plus* offers authentication to protect memory operations. The device can be locked or not.

► The device status is shown on the locker icon and it can be:

| Device	State	Description	Icon
NTAG I^{2}C	All open.	No Authentication feature.	
NTAG I^{2}C *plus*	Unlocked	Memory is unprotected.	
NTAG I^{2}C *plus*	Locked and not authenticated	Memory is protected and user has not introduced the correct password. No Access.	
NTAG I^{2}C *plus*	Locked and authenticated.	Memory is protected and user has introduced the correct password.	

Software Tools to evaluate NTAG I^{2}C *plus*

Explorer
Demo kit
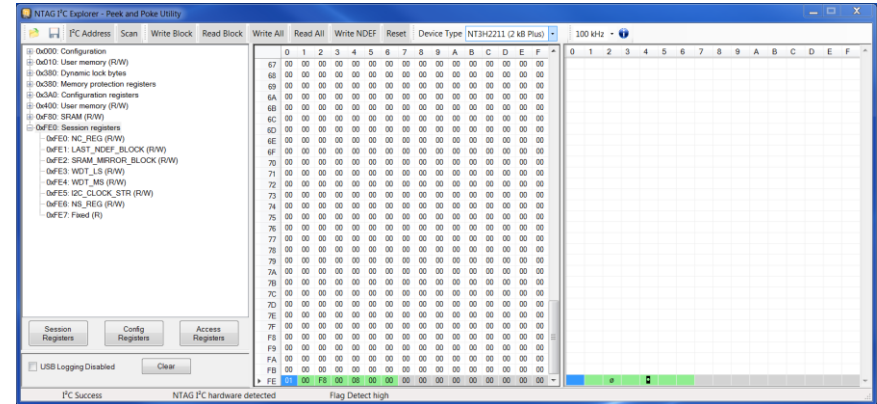Applications

Android Application

Pc Windows Application
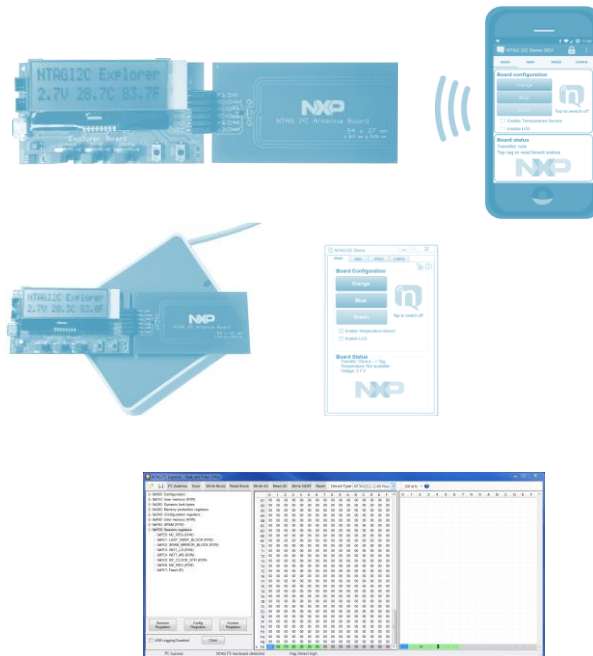
Peek and Poke

NTAG I²C *plus* Peek and Poke

- ► PC-based tool working via I²C serial bus interface.

- ► Exploration of detailed memory content of the EEPROM.

- ► Read/Write all memory.

- ► Read/Write a block of memory.

- ► Read/Write NDEF messages.

- ► Read/Write Access, Session and Configuration registers.

- ► Read/Write I²C address.

NTAG I²C *plus* Peek and Poke
Action Bar

Export/Import Memory Contents

Scan all I²C Address

Change I²C Address of the NTAG I²C *plus*

NTAG I²C *plus* Peek and Poke
Action Bar

NTAG I²C *plus* Peek and Poke
Action Bar

NTAG I²C *plus* Peek and Poke
Action Bar

Reset memory content to original value

Selection for the type of NTAG I²C

Information about the application

Selection the I²C clock frequency

NTAG I²C *plus* Peek and Poke
Memory Block Selection

► Fluent and intuitive graphical exploration of the tag memory

► The panels are connected. What is selected in one, is highlighted in the others.
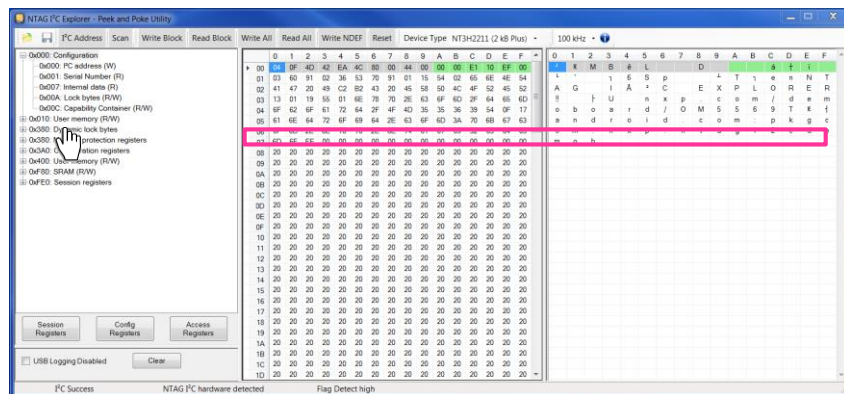
► The Tree View offers a better navigation through the different memory sections.
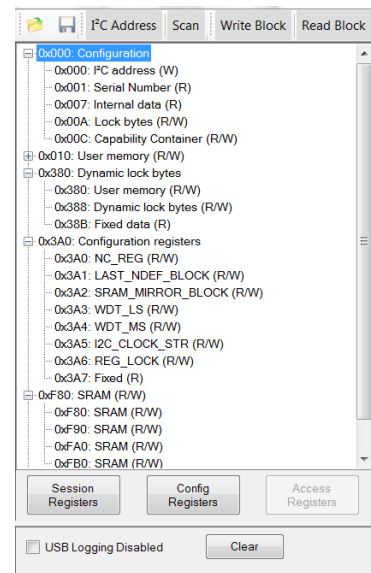
NTAG I²C *plus* Peek and Poke
Memory Block Selection – Unprotected Memory

NTAG I²C *plus* Peek and Poke
Memory Block Selection – Read Only Memory Area

Memory access configuration
From RF/NFC interface

► 32-bit password

► 16-bit PACK (password auth. acknowledge response)

► Optionally limited number of unsuccessful authentications
  ▪ up to 27 negative attempts

► Write or read/write memory access can be restricted to be allowed only after password authentication

► Memory can be split in open and protected segments
  ▪ Memory boundary is configurable

NFC page address (Sector 0)		Byte number from NFC perspective			
Dec	Hex	0	1	2	3
227	E3h	RFU	RFU	RFU	AUTH0
228	E4h	ACCESS	RFU	RFU	RFU
229	E5h	PWD			
230	E6h	PACK		RFU	RFU
231	E7h	PT_I2C	RFU	RFU	RFU

EEPROM

Un-protected memory access → Read & Write

After 32-bit password authentication only →

WRITE Protection or READ & WRITE Protection

RF/NFC protected memory access
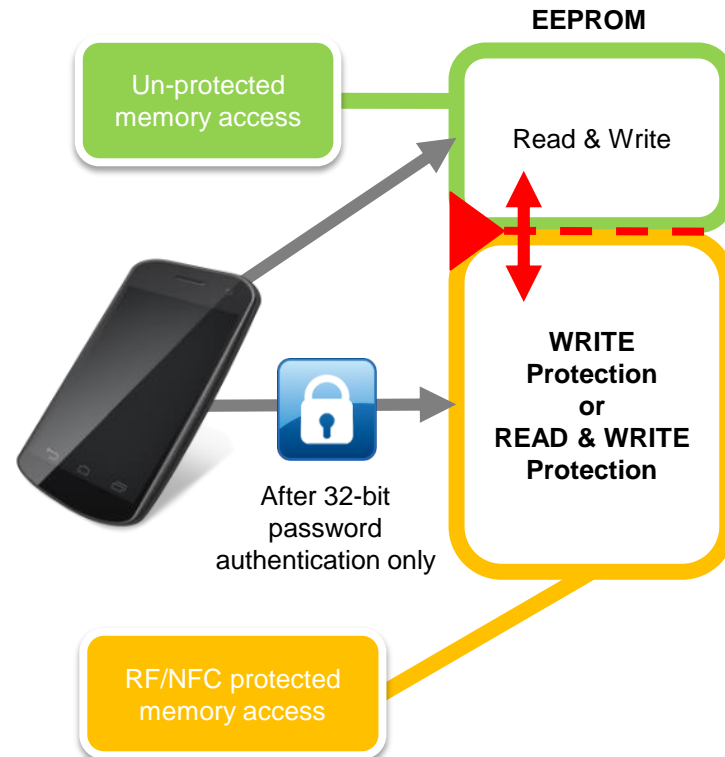
NTAG I²C *plus* Peek and Poke
Access Registers

Password protection for pass-through and mirror mode

Limit of times of false password attempts

Starting memory address of the protected area.

Password protection for Sector1 for NTAG I²C *plus 2K version.*

NFC Access protection to Sector1

Type of Access to protected area from I²C perspective

Sets if the protection is to write or to read and write

32bit password to protect the memory

16bit password acknowledge.

NTAG I²C *plus* Peek and Poke
Session Registers

Sets the Pass Through mode, the event on FD pin, the SRAM mirroring and the data frow direction.

Enables I²C clock stretching.

I²C block address containing last of NDEF message

Watchdog time control

Lock of memory Access from I²C or RF, if NDEF is read, if SRAM is ready to be read by I²C or by RF, error during EEPROM operation and if RF field is present.

I²C block address of SRAM when mirrored into the User memory.

Session Registers

NC_REG : 0xFE:0
- ☐ I2C_RST_ON_OFF
- ☐ PTHRU_ON_OFF
- FD_OFF
  - 00: Field switched off
- FD_ON
  - 00: Field presence
- ☐ SRAM_MIRROR_ON_OFF
- ☑ TRANSFER_DIR

Read NC_REG   Write NC_REG

LAST_NDEF_BLOCK : 0xFE:1

Last BLOCK NDEF message   00

Read NDEF   Write NDEF

WDT : 0xFE:[4:3]

08   48

Watchdog time-out: 20.000 ms

Read WDT   Write WDT

I2C_CLOCK_STR : 0xFE:5
- ☑ I2C_CLOCK_STR: Enabled

Read I2C_STR

SRAM_MIRROR_BLOCK : 0xFE:2

I²C block address   F8

Read SM_REG   Write SM_REG

Read All

NS_REG : 0xFE:6
- ☐ NDEF_DATA_READ (RO)
- ☐ I2C_LOCKED
- ☐ RF_LOCKED (RO)
- ☐ SRAM_I2C_READY (RO)
- ☐ SRAM_RF_READY (RO)
- ☐ EEPROM_WR_ERR
- ☐ EEPROM_WR_BUSY (RO)
- ☐ RF_FIELD_PRESENT (RO)

Read NS_REG   Write NS_REG

NTAG I²C *plus* Peek and Poke
Configuration Registers

All registers are the same as in the session registers

NTAG I²C *plus*

Developing applications

How to develop applications around NTAG I²C *plus*

Available material to develop your own solutions

- **Firmware**
- **Peek and Poke**
- **Pc Windows Application**
- **Android Application**

How to develop applications around NTAG I^{2}C *plus*

► Develop and test your own applications.

► The Peek and Poke, Windows app and Android application are a reusable code.

► Versatility of APIs to develop around NTAG I^{2}C *plus* (*in Java, C# and C++).

► Environments: Windows and Android platform.

```
53          {
54              return (int)busSpeed;
55          }
56
57
58          private byte[] CreateCommandToInitoI2CPort()
59          {
60              byte[] data = new byte[16];
61              data[0] = 0x0C;    // total number of bytes
62              data[1] = 0x00;    // Transaction identifier
63              data[2] = 0x29;    // Session identifier
64              data[3] = 0x01; // HID_I2C_REQ_INIT_PORT
65              data[4] = (byte)(this.busSpeed & 0xFF);    // I2C bus speed
66              data[5] = (byte)((this.busSpeed >> 8) & 0xFF);
67              data[6] = (byte)((this.busSpeed >> 16) & 0xFF);
68              data[7] = (byte)((this.busSpeed >> 24) & 0xFF);
69              data[8] = 0x00;
70              data[9] = 0x00;
71              data[10] = 0x00;
72              data[11] = 0x00;
73              data[12] = 0x00;
74              data[13] = 0x00;
75              data[14] = 0x00;
76              data[15] = 0x00;
77              return data;
78          }
79
80          /** @brief Initialize the communication with the port
81           *
82           *  @param device is the object representing the device to connect with.
83           *  @return the response from the command to init the port.
84           */
85          public I2CData init(I2C_Device device)
86          {
87              var command = CreateCommandToInitoI2CPort();
88              var returnVal = device.performTransaction(command);
89              initialised = true;
90              return returnVal;
91          }
92
93          /** @brief Reinitialize the communication with the port
94           *
95           *  @param device is the object representing the device to connect with.
96           *  @param busSpeedInKHz is the bus speed on the bus for the communication
97           *  @return the response from the command to init the port.
98           */
99          public I2CData reinit(int busSpeedInKHz, I2C_Device device)
100         {
101             I2CData result = null;
102             setBusSpeedInHz(busSpeedInKHz * 1000);
103             if (device.isConnected())
104             {
105                 result = init(device);
106             }
107             return result;
108         }
109     }
110 }
```

How to develop applications around NTAG I²C *plus*

Firmware

Peek and Poke

Pc Windows Application

Android Application

Available material to develop your own solutions

How to develop applications around NTAG I²C *plus*
Firmware development

► Firmware consists of:

  ▪ The boodloader

  ▪ The Explorer Demo Application

  ▪ The Blink

► LPCXpresso software

  https://www.lpcware.com/lpcxpresso/downloads/windows

► Use LPCLink2 to connect (or flash)

► An application can also be flashed with Android application.
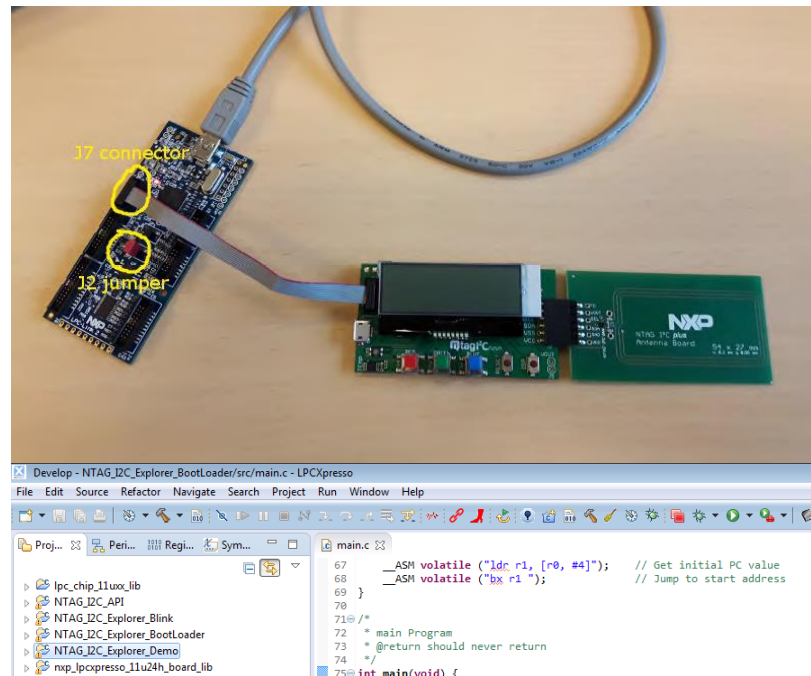
How to develop applications around NTAG I^{2}C *plus*

Available material to develop your own solutions

Firmware

Peek and Poke

Pc Windows Application

Android Application

How to develop applications around NTAG I²C *plus*
Peek and Poke API

► API directly offers the functionalities of the Peek and Poke (Read all memory, read config registers, …)

► The architecture of the Peek and Poke offers several layers of abstraction

- NTAG I²C Communication
- NTAG I²C Commands
- NTAG I²C API

► Developed in *C#*

How to develop applications around NTAG I2C *plus*

Firmware

Peek and Poke

Pc Windows Application

Android Application

Available material to develop your own solutions
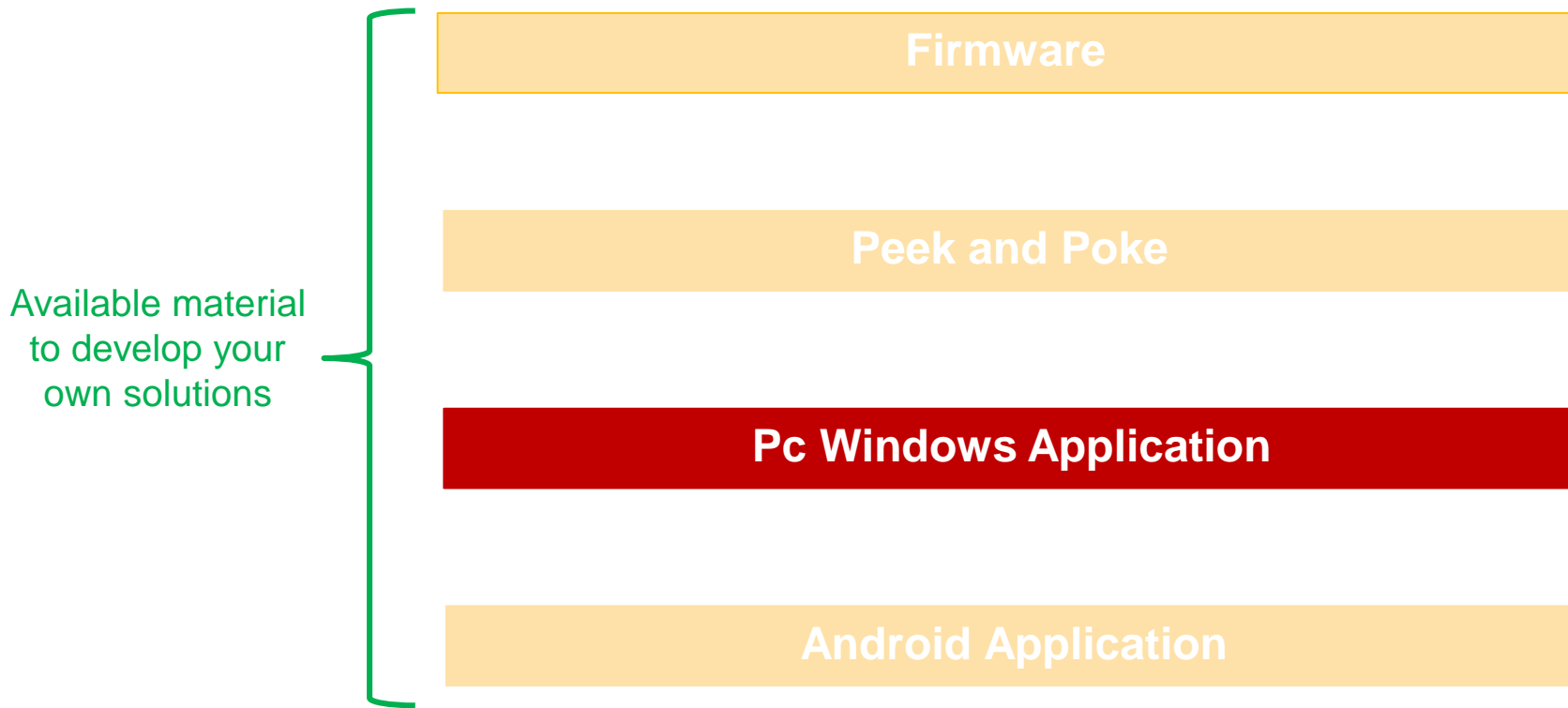
How to develop applications around NTAG I²C *plus*
Windows App API

► Microsoft Visual Studio.

► Developed in C++

► API directly offers the functionalities of the Windows app (write NDEF, read EEPROM, set Password,…)

► PCSC communication.

► Easy to develop your own application using this code as example and the available API.
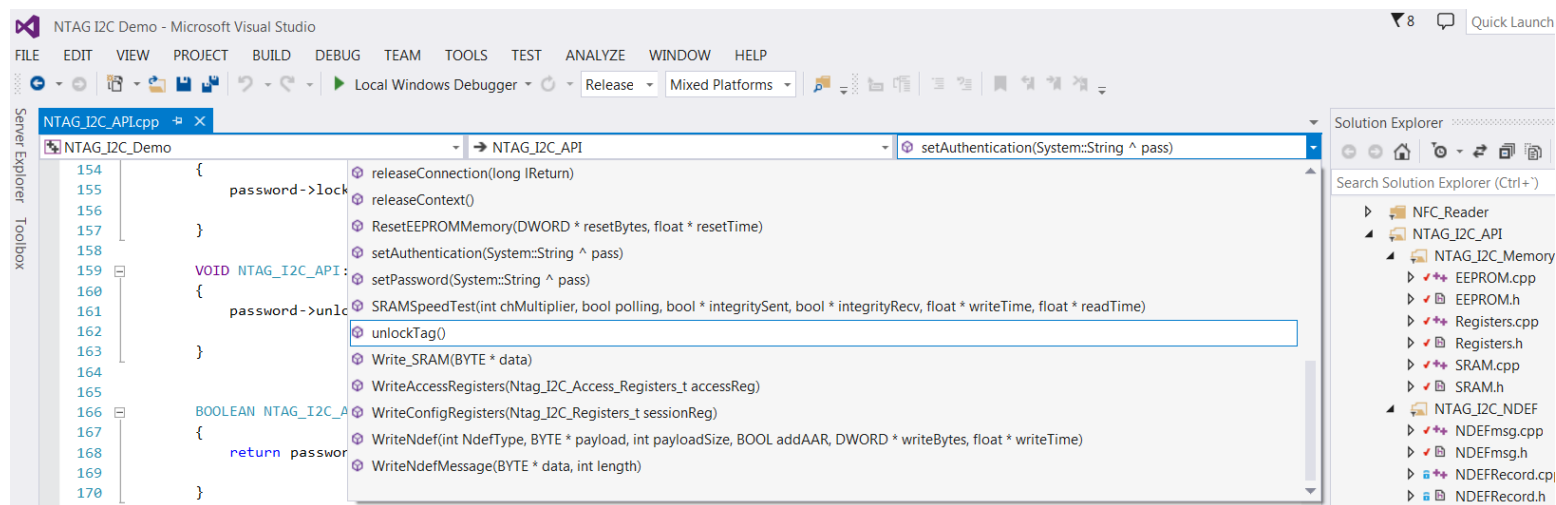
How to develop applications around NTAG I²C *plus*

Available material to develop your own solutions
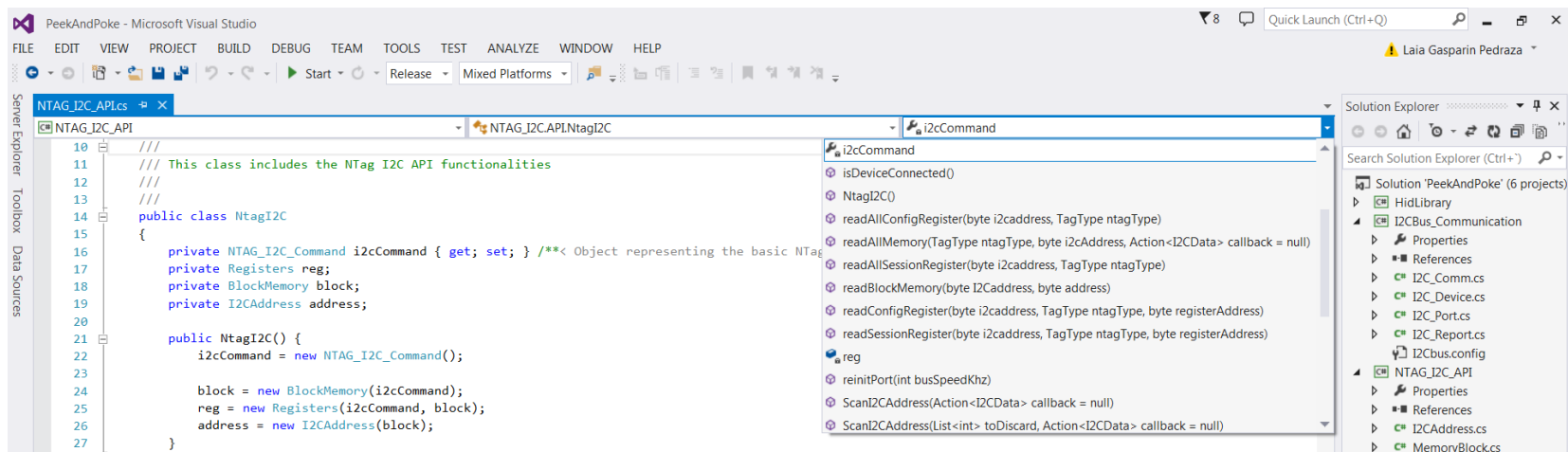
Firmware

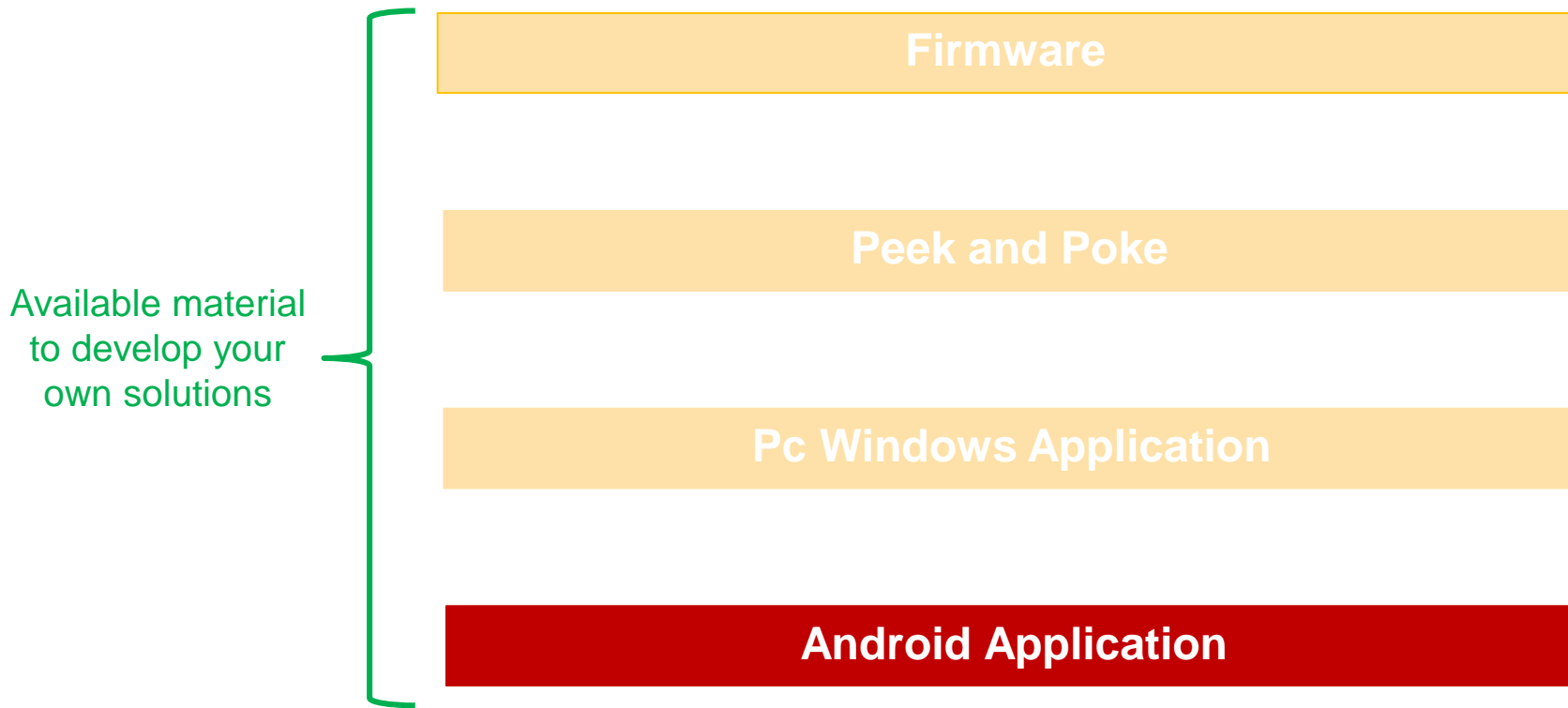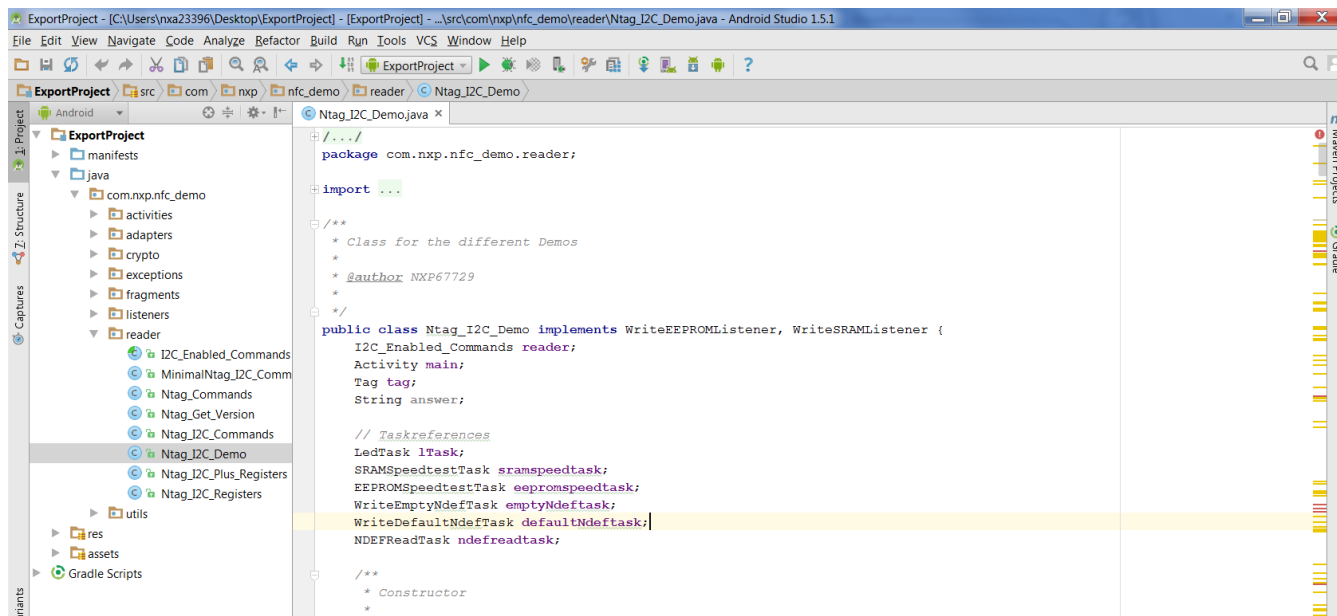| Peek and Poke |
| --- |

Pc Windows Application

Android Application

How to develop applications around NTAG I²C *plus*
Android app

► Source code for Android Studio to develop your own app.

► Faster development to go to market.

► Developed in Java.

► APIs available for the RF commands and the higher level functionalities (read SRAM block, …)

Wrap up

NTAG I²C *plus* is supported by advanced tools

NFC frontend development kit OM5569-NT322E
Available at eDemoboard and Distis

Windows and Android application
A mobile and desktop application to test features of the NTAG I2C plus

Peek and Poke
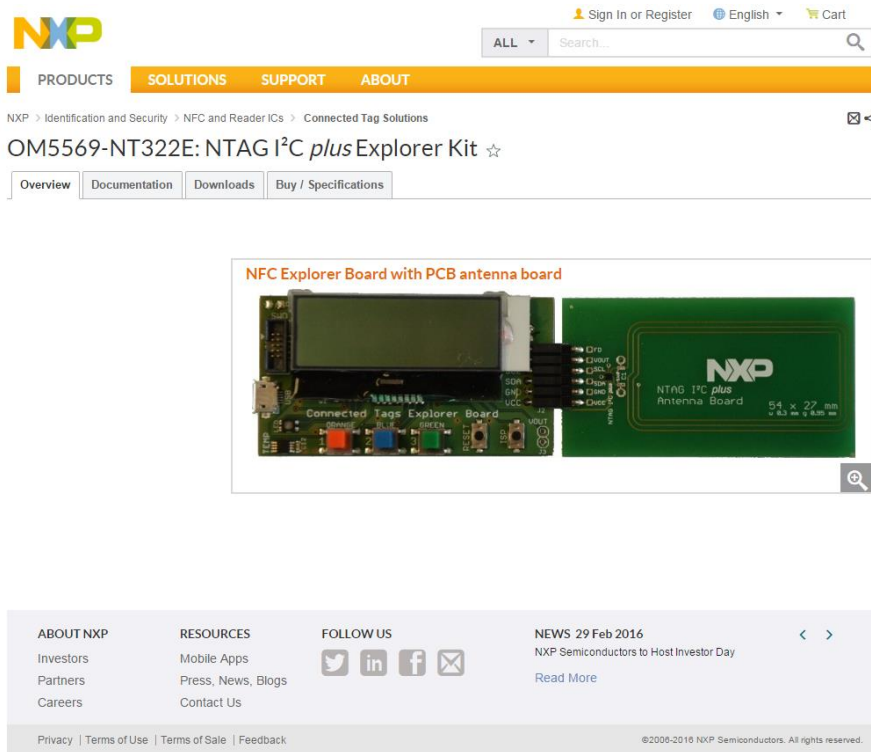*A desktop application to navigate and explore the memory.*

Android code, C# and C++ APIs
Example code and APIs to develop application upon it.

Do you need more?
Resources and useful links

► NFC Everywhere
http://www.nxp.com/nfc

► NT3H2111/2211: NTAG I²C *plus* product website
http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/connected-tag-solutions/ntag-ic-plus-nfc-forum-type-2-tag-compliant-ic-with-ic-interface:NT3H2111_2211

► OM5569-NT322E demokit website
http://www.nxp.com/products/identification-and-security/nfc-and-reader-ics/connected-tag-solutions/ntag-ic-iplus-i-explorer-kit:OM5569-NT322E

► NXP Tech community
http://nxpcommunity.force.com/community/CommunityOverview

Thank you