Application Note MLX90121

Near Field Communication protocol

1 Scope

This document describes the Near Field Communication (NFC) demonstration, which consists of an exchange of information between two MLX90121s ”13.56 MHz RFID transceivers”, according to the NFC protocol.

The NFC protocol will be briefly discussed first, and then a complete description of the implementation on the MLX90121 transceiver will be further described.

2 Related Melexis product

MLX90121	13.56MHz RFID Transceiver

3 What is Near Field Communication protocol?

The Near Field Communication is very short-range radio frequency identification (RFID) protocol based on an inductive 13.56MHz RF link that provides easy and secure communications between various devices without user configuration. It allows users to transfer content between devices, make payment transactions, and access content from smart objects.

In the RFID world, the distinction is made between active devices or transceivers (readers), which are able to generate their own RF field, and passive devices or transponders, which use the RF field from active devices to generate the power supply and to communicate. This in mind, the NFC protocol implements two types of communication; active communication, which defines the communication between active devices and passive communication; which defines the communication between active and passive devices.

4 General description

The demonstration consists of a file transfer between two active devices embodied by the MLX90121 RFID transceiver and using the Near Field Communication protocol. The communication is initiated by the Initiator device, which controls the communication, whereas, the Target device seizes the transferred information and replies to the Initiator.

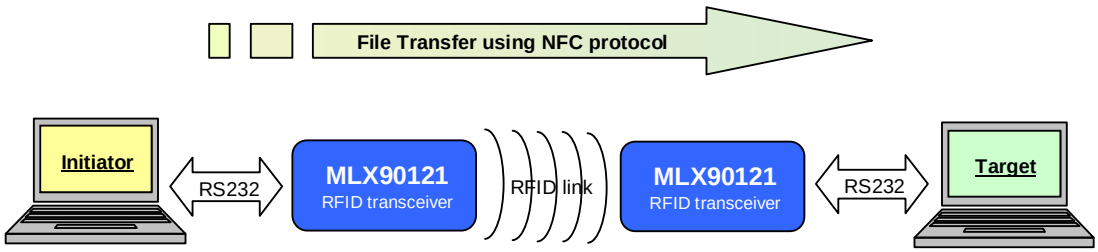


Figure 1: Near Field Communication general description

The user interface software provided by Melexis can be run on each computer connected to the MLX90121 device with a RS232 interface. This software allows the user to monitor the NFC communication and will be described in detail in this document.

Thanks to the high flexibility of the MLX90121 RFID device, the implementation of the active NFC protocol, with a data rate up to 212 kbit/s, is performed only by applying a relevant configuration to the internal registers.

5 NFC protocol specifications

This chapter describes all specifications of the NFC protocol implemented in the demonstration. For more information about NFC, please refer to the chapter "*Related documentation*".

5.1 Type of modulation

Data information is sent from the device by amplitude modulation (load modulation) using a modulation depth of 10%. A Manchester encoding is used without any sub-carrier to achieve a data rate of 212 kbit/s.

Carrier Modulation	Bit coding	Data rate
ASK 10%	Manchester	212 kbits/s

Table 1: Active communication mode, high data rate

Manchester encoding is implemented with a high polarity, means that a logical "one" is encoded with a transition from '1' to '0' while a logical "zero" is encoded with a transition from '0' to '1' as shown in the following picture.

Figure 2: Manchester representation

5.2 Active communication mode

The active communication mode will be used to transfer information between two MLX90121 transceivers. Thus, both devices will use their own RF field to transmit information according to the NFC protocol. The following picture shows the general behaviour of an active communication performed by the system.

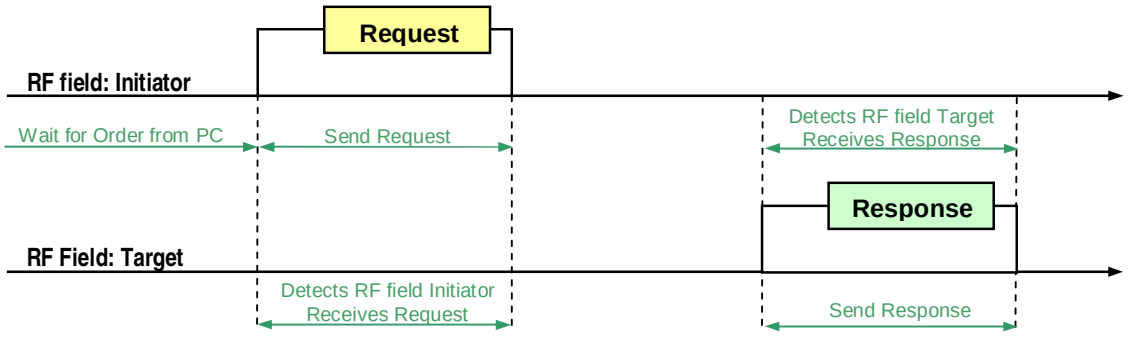


Figure 3: Active communication representation

The Initiator device starts the communication by setting its own RF field ON and by modulating it with the data information corresponding to the intended request. The Target device detects the RF field from the Initiator device and starts to decode the request. Then, by using the same method, the Target device replies using its own RF field with the corresponding response modulation.

Note: As the demonstration occurs between only two active devices, no RF collision avoidance protocol is supported for the moment.

5.3 Frame format

The following figure shows the frame representation used in the NFC demonstration according to the NFC specifications.

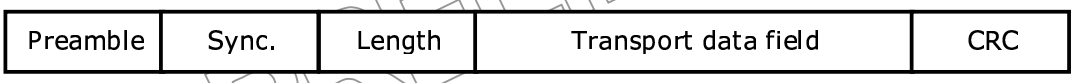
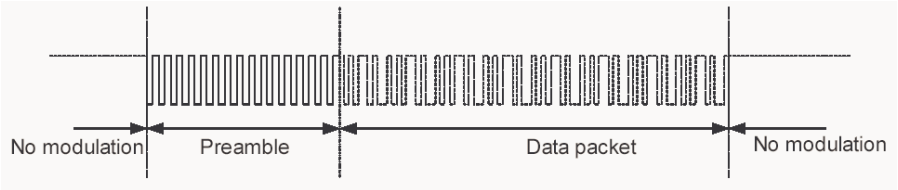


Figure 4: Frame format representation

5.3.1 Preamble

The communication starts with a preamble sequence of minimum 48 bits with all logical “zero” encoded.



Picture 1: Start and end of communication

5.3.2 Synchronization (SYNC)

The SYNC shall be 2bytes. The first byte **0xB2** and second shall be **0x4D**.

5.3.3 Length

The Length shall be an 8bit field and it will be set to the number of bytes to be transmitted plus 1 (CMD0 + CMD1 + Byte0 to n). LEN field shall be in the range of 2 to 255.

5.3.4 Transport data field

Please refer to the chapter “instruction set” further in this document.

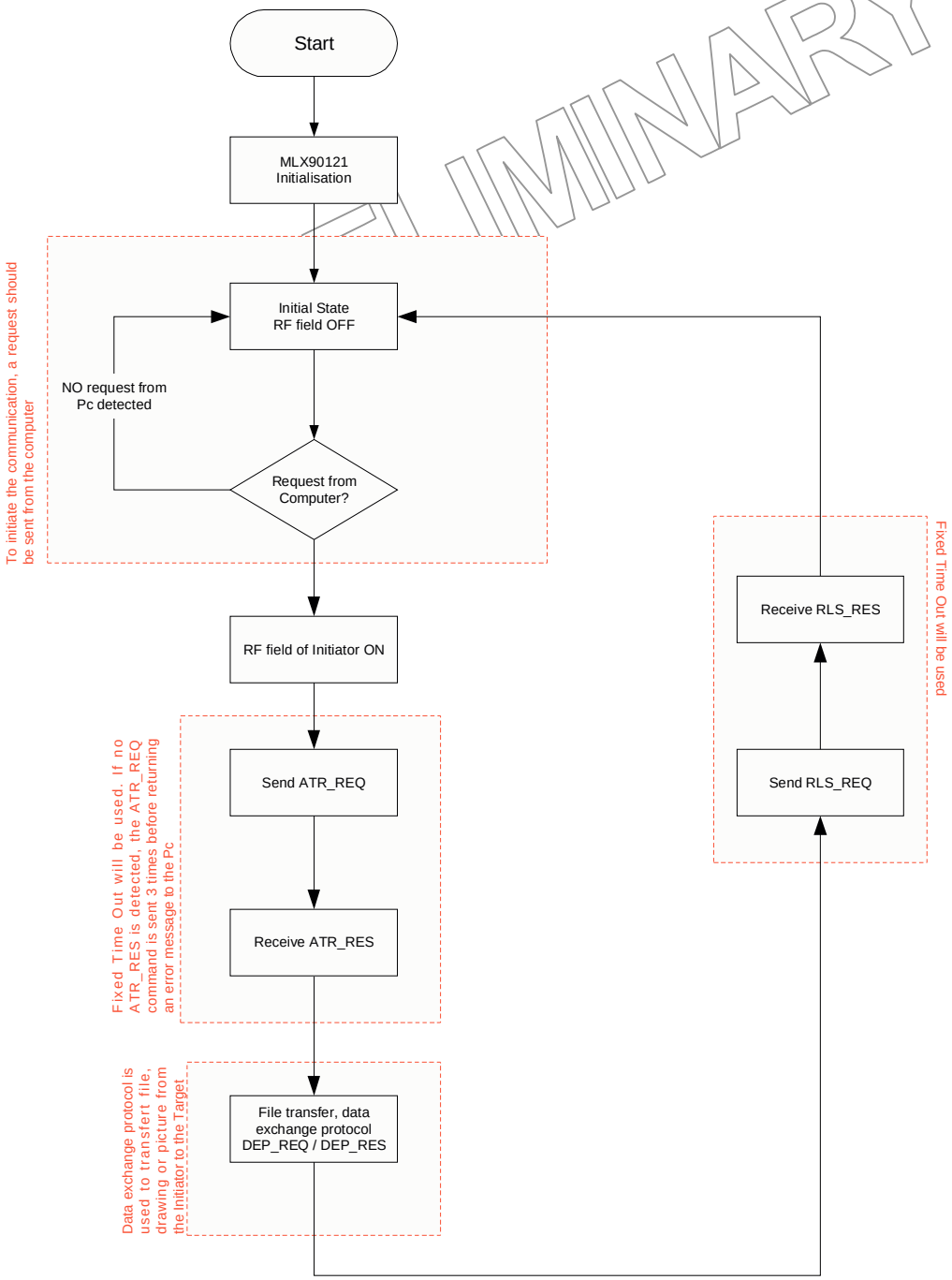
5.3.5 Checksum (CRC)

CRC is a 16bits field calculated on Length and Transport data fields. Following formula will be used to calculate CRC with a pre-set value of 0.

$$G(x) = x^{16} + x^{12} + x^5 + 1$$

5.4 General protocol flow chart

The picture below shows the general flow chart implemented for the NFC demonstration.



Picture 2: General protocol flow chart

5.5 Instruction set

5.5.1 Attribute commands

The attribute request is used by the Initiator to control the presence of an NFC target in the field. This is why all parameters included in this command are fixed and cannot be changed for the moment.

5.5.1.1 ATR_REQ

CMD0	CMD1	Byte0 to 9	Byte 10	Byte11	Byte12	Byte13
0xD4	0x00	0x00	0x00	0x0F	0x0F	0x30

5.5.1.2 ATR_RES

CMD0	CMD1	Byte0 to 9	Byte 10	Byte11	Byte12	Byte13	Byte14
0xD5	0x01	0x00	0x00	0x0F	0x0F	0x00	0x30

Note: Byte13 correspond to the time out of the Target device. It is not used for the moment and it is set to zero.

5.5.2 Data Exchange protocol commands

The data exchange protocol is half-duplex protocol supporting block oriented data transmission with error handling (Acknowledgment from the Target). After Target detection (ATR request and response), the Initiator starts to send data information using the data exchange protocol. Then the Target replies on each block with an acknowledgement command. If this command is not received, the data block will be sent again.

For more information, please refer to the chapter "*related document*".

5.5.2.1 DEP_REQ

CMD0	CMD1	PFB	Byte0 to 127
0xD4	0x06	Depends on data information (MI & PNI bits)	Data information

5.5.2.2 DEP_RES

CMD0	CMD1	PFB
0xD5	0x07	Depends on data information (ACK & PNI bits)

5.5.3 Release commands

The release commands are used to close the communication. Then both Initiator and Target will return to the initial state (RF field OFF)

5.5.3.1 RLS_REQ

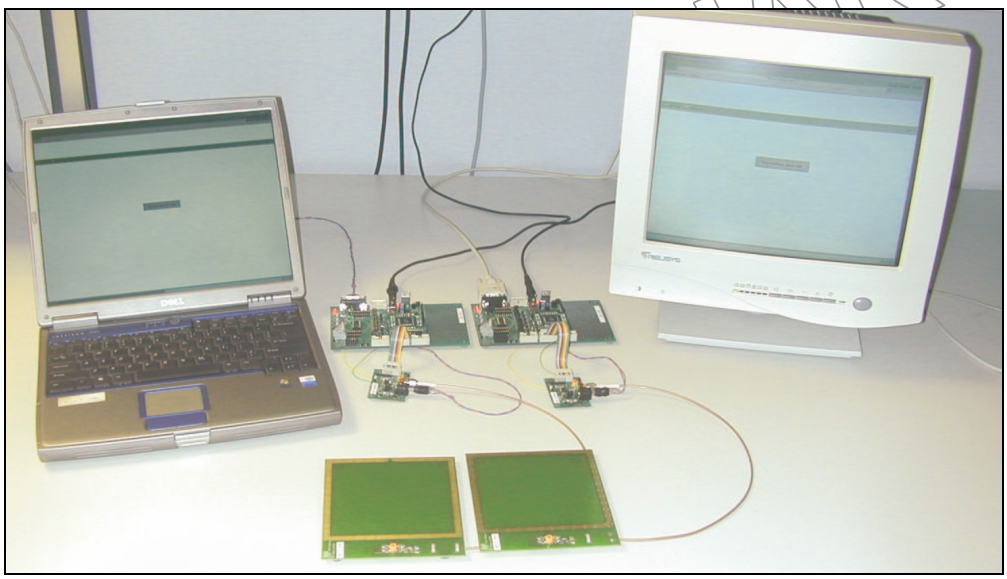
CMD0	CMD1
0xD4	0x0A

5.5.3.2 RLS_RES

CMD0	CMD1
0xD5	0x0B

6 Demonstration setup

The demonstration setup consists of two development kits DVK90121, which include a microcontroller board called DVKRFID connected to the evaluation board EVB90121 and a 13.56MHz PCB antenna. These two DVK90121s are used as NFC base stations and are connected to the computer through the RS232 interface.



Picture 3: NFC demonstration

The user interface software, that can be run on both computers, allows the user to monitor the NFC communication.

6.1 DVK90121 hardware connections

To better understand the hardware connection of the DVK90121, please refer to the documentation "*DVK90121 User Manual*" available on our web site www.melexis.com.

To be able to detect the RF field from another device, an internal signal from the analog chain of the MLX90121 transceiver should be used on pin 3. This is done with a specific configuration of the register *PowerState* described in the next chapter. This internal signal available on pin 3 should be connected to the internal comparator input of the microcontroller (pin 5).

6.2 MLX90121 internal registers configuration

The high flexibility of the MLX90121 transceiver allows the implementation of the NFC protocol without any external components only by using a specific configuration of the internal registers, as described in the following table.

Name	Address	Value	Comments
AnalogConfig	0	0x26	10% modulation / Comparator in level "medium"
PowerState	1	0x0D	Transmitter ON, Internal signal from analog chain on pin 3
DigitalConfig	3	0x2B	Trigger Mode En / MV En / LIMITER / Encoder En
EncoderSym0	4	0x0F	Logical Manchester '0'
EncoderSym1	5	0xF0	Logical Manchester '1'
EncoderTimeRef	A	0x01	Encode symbol at 212kbit/s
DecoderTimeRef	B	0x0F	-
LTC	C	0x22	-

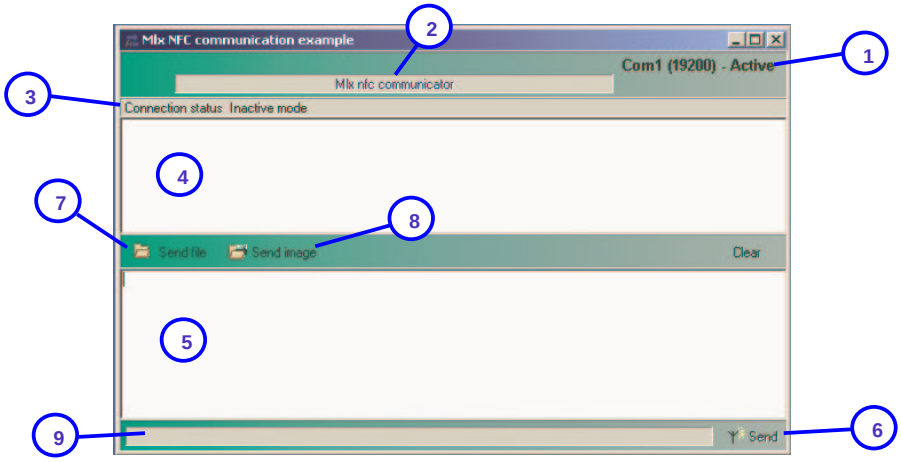
Note: To decode the incoming modulation, the LIMITER output of the log. amplifier is used. This makes Dout very noisy and the microcontroller needs to be synchronized at the beginning of the frame (preamble).

Note: Only two symbols (sym0 and sym1) of the internal hardware encoder are used to encode data information. The data rate is predefined in the *EncoderTimeRef* register.

Note: The register *PowerState* is configured to extract internal signals from the analog chain of the MLX90121.

6.3 User interface software

The following picture shows the user interface software used to control the NFC communication.

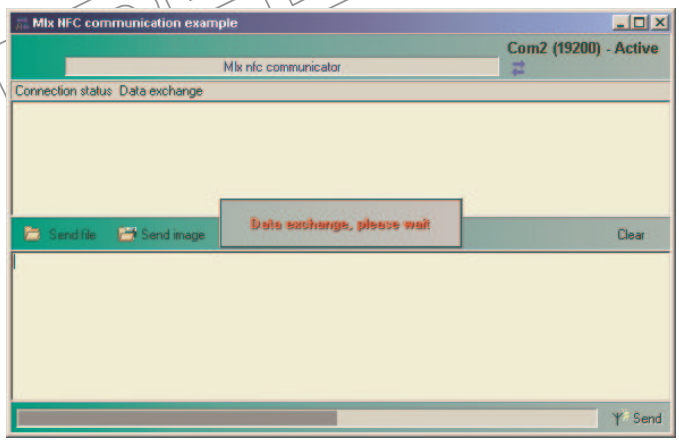


Picture 4: User interface software for NFC communication

1) COM-port information: Shows COM name, baud rate and status. Double click to change.

2) Device information: Program polls the connected device all the time and asks a device name. If there is no device connected to the PC, this line will be empty.
3) Communication status: Shows the progress of connection and communication.
4) Read Text dialog window: The text dialog send by another NFC device is displayed in this window.
5) Write Text dialog window: The text written in this window is sent using NFC communication. Then the corresponding text will be displayed on the Read Text dialog window of another NFC device.
6) Send button: When pressed, program will start sending text from window **5**.
7) Send file button: When pressed, the open-file dialog will appear and allows to the user to choose a file to send. When chosen, the corresponding file will be automatically sent.
8) Send image button: When pressed, the open-image dialog will appear and allows to the user to choose a image file to send. When chosen, the corresponding image will be automatically sent.
9) Progress bar: When text, file or image is sent, it shows graphically the communication progress.

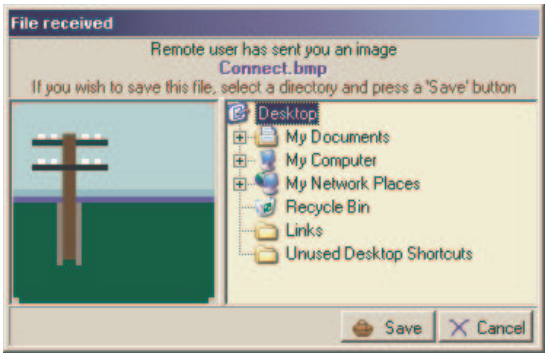
When the program starts to send data, the following window will be displayed



Picture 5: File transfer window

Note: The connection status bar will reflect the current state of the communication, a small window "*Data exchange, please wait*" will appear and the progress bar will show the exchange progress. Controls buttons, like "Send file", "Send image", "Send" will be temporarily blocked.

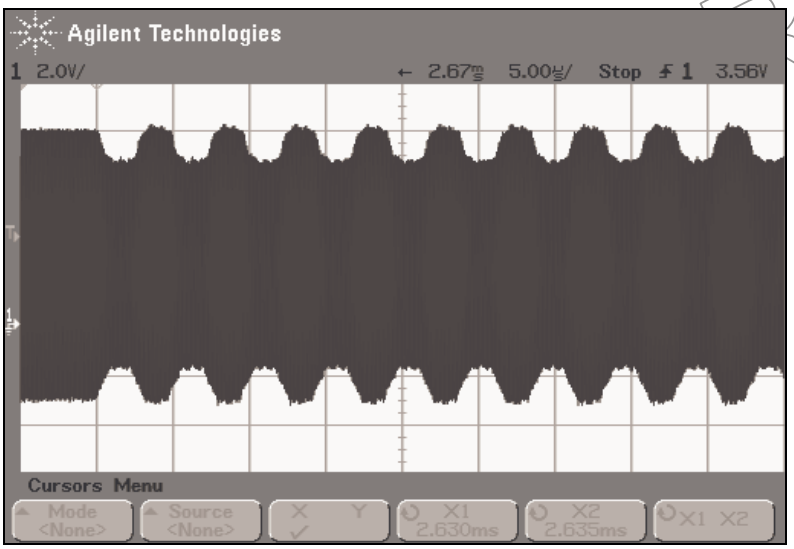
When program receives a file or an image, a message will be showed and the user is able to save this file.



Picture 6: Save transferred file

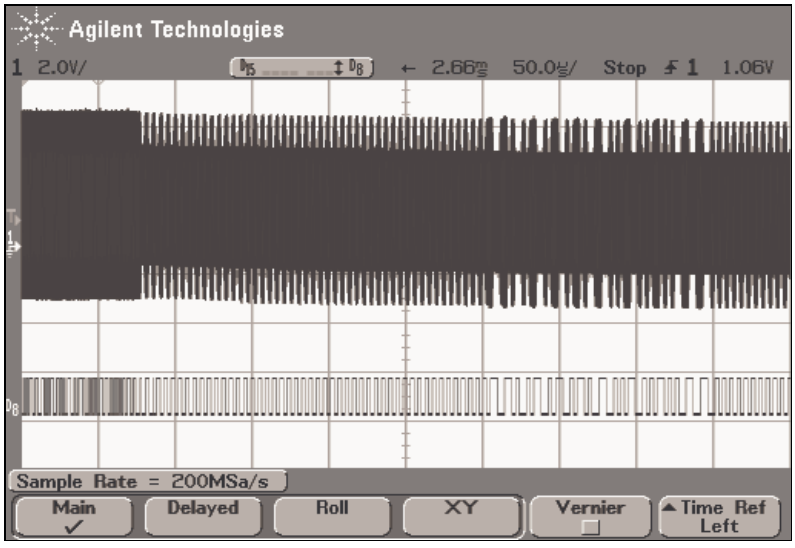
7 Waveforms

Some waveforms can be captured on the EVB90121 circuit to illustrate the NFC communication.



Picture 7: Preamble sent by the initiator device

Note: The data information is modulated on the Initiator antenna with a modulation depth measured at 13% and a data rate of 212kbit/s.



Picture 8: ATR_REQ received by the Target

Note: On this capture, D8 represents the decoded data Dout. A lot of noise appears before the request as a result of the use of the LIMITER output of the log. amplifier. The microcontroller is however able to detect the preamble of the corresponding request.

8 Related documents

- ECMA-340 “Near Field Communication – Interface and Protocol (NFCIP-1)”
- ISO/IEC 18092 “ISO/IEC JTC1 adopted ECMA-340 under their fast track procedure”
- ECMAS-352 “Near Field Communication Interface and Protocol – 2 (NFCIP-2)”