# Chapter 10 NFC Data Exchange Format

NFC Data Exchange Format (NDEF) is a data packaging format defined by the NFC Forum; NFC devices can easily use NDEF Messages to exchange data regardless of the carrier, transmission protocol and packaged data.

NDEF is a data packaging format; it is not associated or dependent on the lower layers; no matter which carrier (card or NFC device) or transmission protocol (NFC A/B/F) is used, it's all the same to NDEF. In terms of data format, NDEF does not redesign the data format; instead it packages the existing data format into NDEF Message. Simply put, if a vCard is received, process it with the contact list; if a web address is received, process it with a browser; if a video is received, use a player to process it.

## 10.1 NDEF Message

The NDEF Message includes one or more NDEF Record; the first Record is called the MB (Message Begin) Record and the last is called the ME (Message End) Record. If there is only one Record in the Message, then this Record is the MB Record as well as the ME Record.

The NDEF Message does not have a special structure; it simply connects all Records together. You can package different types of Records in the same Message, and also divide the same types of Records into Chunks and package them in the same Message as well. The main reason for this design is due to the transmission capacity every time of NFC devices; we can divide the data into multiple Records and take turns to transmit them, and once they are all received, then combine them again. The combination range is within the Message.

NDEF does not take the packaged data format into consideration; therefore it the packaged data itself has data combination functions, the combination range is not limited to the Message range. As shown in (Table 10-1).

Table 10-1

| NDEF Message | | | | | | |
|---|---|---|---|---|---|---|
| $R_{MB}$ | $R_{Picture}$ | $R_{Vedio}$ | … | $R_{Text}$ | … | $R_{ME}$ |

## 10.2 NDEF Record

The NDEF Record is a very simple structure; it only includes Flag, Length and Payload, in which the payload refers to the original data. As shown in (Table 10-2).

Table 10-2

| Flag | | | | | | | |
|---|---|---|---|---|---|---|---|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| MB | ME | CF | SR | IL | TNF | | |
| **Length** | | | | | | | |
| The Length of Payload Type | | | | | | | |
| The Length of Payload Data | | | | | | | |
| The Length of Payload ID | | | | | | | |
| **Payload** | | | | | | | |
| Payload Type | | | | | | | |
| Payload ID | | | | | | | |
| Payload Data | | | | | | | |

- **MB**

  If set as 1, it means the first Record in the NDEF Message; there can only be one MB Record in a NDEF Message.

- **ME**

  If set as 1, it means the last Record in the NDEF Message; there can only be one ME Record in a NDEF Message.

- **Chunk Flag (CF)**

  If set as 1, it means the next Record and the current Record have chunk relationship; if set as 0, it means the next Record and the current Record has no chunk relationship.

- **Short Record (SR)**

  If set as 1, it means that payload length included in this Record is between 0-255.

- **ID Length (IL)**

  If set as 1, it means that this Record includes Payload ID Length and Payload ID; if set as 1, it means that this Record does not include ID Length and Payload ID.

- **Type Name Format (TNF)**

  This coding format consists of 3 bits; therefore, the type setting value is 0~7.

  The type represented by the type setting values and Payload Type must correspond with one another; Payload Type must be one of the types set in TNF type, as shown in (Table 10-3).

Table 10-3

| Setting value | Types |
|---|---|
| 0 | Empty Types. |
| 1 | NFC Forum Well Know Types; Payload Type is in accordance with RTD specifications. |
| 2 | Media Types; Payload type is in accordance with RFC 2046. |
| 3 | Uniform Resource Identifier (URI) Types; Payload Type is in accordance with RFC 3986. |
| 4 | NFC Forum External Types; Payload Type is in accordance with RTD specifications. |
| 5 | Unknown Types. |
| 6 | Types is the same as the previous Record; applied on Chuck Records. |
| 7 | Reserved. |

- **The Length of Payload Type**

  The length of the payload type field.

- **The Length of Payload Data**

  The length of the payload data field.

  When SR is set as 1, this field will take up 1 byte; when SR is set as 0, this field will take up 4 bytes.

- **The Length of Payload ID**

  The length of the payload ID field.

  When IL is set as 1, this field exists; when IL is set as 0, this field does not exist.

- **Payload Type**

  The payload data type, it must correspond to TNF type.

- **Payload ID**

  The payload ID uses URI coding format; it is used to connect one Record with another.

  The effect of payload ID is different from Chunk Record. In Chunk Record application, one Record can only have one payload ID, therefore the payload ID of every Chunk Record must be identical.

- **Payload Data**

  User data.

## 10.3 RTD Specifications

TNF in NDEF has many types specified, in which the NFC Forum Well Known Type (TNF is 1) and NFC Forum External Type (TNF is 4) are types in accordance with the Record Type Definition (RTD) specifications.

- **NFC Forum Well Known Types**

  NFC Forum Well Known Types use URN Uniform Resource Name (URN) naming rule, which is "urn:nfc:wkt:T", in which the "urn:" prefix means use URN naming rules, "nfc:wkt:" represents the namespace identifier (NID), where "wkt" is the abbreviation of Well Know Types and "T" is Type Name.

  NFC Forum Well Known Types can be divided into Global Type and Local Type. Global Type must begin with a capital letter, such as: Text Record is "T", URI Record is "U" and Smart Poster Record is "Sp".Local Type must begin with a common letter or number. Local Type is generally defined by the developer.

- **NFC Forum External Type**

  URN: "urn:nfc:ext:example.com:test".

  NID: "nfc:ext:", where "ext" is the abbreviation of External Types.

  Type Name: "example.com:test". The type name must include the domain name of the named groups.

## 10.4 RTD Text Specifications

Text Record is one of NFC Forum Well Known Types; therefore, TNF is 1, Type Name is "T" ("urn:nfc:wkt:T") and content is Text Data. As shown in (Table10-4).

Table10-4

| Flag | | | | | | | |
|---|---|---|---|---|---|---|---|
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| UTF 8/16 | Reserved | The length of Language Code | | | | | |
| Language | | | | | | | |
| Language Code | | | | | | | |
| Content | | | | | | | |
| Text Data | | | | | | | |

- **UTF 8/16**

  If set as 0, it means the text data uses UTF 8 coding; if set as 1, it means the text data uses UTF 16 coding.

- **Reserved**

  Must be set to 0.

- **The length of Language Code**

  The length consists of 6 bits; therefore, its ranges from 0 to 63.

  Language Code uses ISO/IANA Language Code; therefore, its current maximum length is 33 bytes.

- **Language Code**

  For example: "en", "en-US" and "fi" etc.

- **Text Data**

  Text content with UTF 8 or UTF 16 coding.

## 10.5 RTD URI specifications

URI Record is one of NFC Forum Well Known Types; therefore, TNF is 1, Type Name is "U" ("urn:nfc:wkt:U") and content is URI Data. As shown in (Table 10-5).

Table 10-5

| Prefix Code |
| --- |
| URI Prefix Code |
| Content |
| URI Data |

- **URI Prefix Code**

  In order to shorten the length of URI Data, 1 byte of coding is used to represent the prefix data of URI Data. As shown in (Table 10-6).

Table 10-6

| Decimal | Hexadecimal | Content |
| --- | --- | --- |
| 0 | 0x00 | N/A |
| 1 | 0x01 | http://www. |
| 2 | 0x02 | https://www. |
| 3 | 0x03 | http:// |
| 4 | 0x04 | https:// |
| 5 | 0x05 | tel: |
| 6 | 0x06 | mailto: |
| 7 | 0x07 | ftp://anonymous:anonymous@ |
| 8 | 0x08 | ftp://ftp. |
| 19 | 0x09 | ftps:// |
| 10 | 0x0A | sftp:// |
| 11 | 0x0B | smb:// |
| 12 | 0x0C | nfs:// |
| 13 | 0x0D | ftp:// |
| 14 | 0x0E | dav:// |
| 15 | 0x0F | news: |
| 16 | 0x10 | telnet:// |

| | | |
|---|---|---|
| 17 | 0x11 | imap: |
| 18 | 0x12 | rtsp:// |
| 19 | 0x13 | urn: |
| 20 | 0x14 | pop: |
| 21 | 0x15 | sip: |
| 22 | 0x16 | sips: |
| 23 | 0x17 | tftp: |
| 24 | 0x18 | btspp:// |
| 25 | 0x19 | btl2cap:// |
| 26 | 0x1A | btgoep:// |
| 27 | 0x1B | tcpobex:// |
| 28 | 0x1C | irdaobex:// |
| 29 | 0x1D | file:// |
| 30 | 0x1E | urn:epc:id: |
| 31 | 0x1F | urn:epc:tag: |
| 32 | 0x20 | urn:epc:pat: |
| 33 | 0x21 | urn:epc:raw: |
| 34 | 0x22 | urn:epc: |
| 35 | 0x23 | urn:nfc: |
| 36~255 | 0x24~0xFF | Reserved |

- **URI Data**

  URI includes Uniform Resource Locator (URL) and Uniform Resource Name (URN); uses UTF 8 coding.

## 10.6 RTD Smart Poster Specifications

Smart Poster Record is one of NFC Forum Well Known Types; therefore, TNF is 1, Type Name is "Sp" ("urn:nfc:wkt:Sp"). It is a combined Record that uses URI Record as the core and uses other Records as aids or modification.

Smart Poster Record must include one URI Record, but cannot include over two URI Records, and other Records are attached Records of this URI Record.

URI itself is very powerful; it includes EPC coding, Internet address, SMS Message and making phone calls etc.

- **Title Record**

  Title Record is a Text Record, it optional in a Smart Poster Record.

  One Smart Poster Record can contain multiple Title Records, but the language cannot be repeated.

- **Action Record**

  Action Record is only a recommendation for how URL should act; the developer can choose to ignore the recommendation, it optional in a Smart Poster Record.

  Action Record can only be applied in Smart Poster Record, and its payload data is only 1 byte. As shown in (Table 10-7).

Table 10-7

| Payload Data | Meaning |
|---|---|
| 0x00 | Execute |
| 0x01 | Save |
| 0x02 | Edit |
| 0x03 – 0xFF | Reserved |

- **Icon Record**

  The Icon Record is optional in a Smart Poster Record.

  One Smart Poster Record can contain multiple Icon Records; when the Smart Poster Record is received, the developer can select one to display.

- **Size Record**

  Size Record is optional in a Smart Poster Record.

  When the URI connects to an external download, the Size Record can first notify the NFC device so that the NFC device can evaluate whether it has the capability to process it. The payload data of Size Record is only 4 byte. As shown in (Table 10-8).

Table 10-8

| Byte | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| Value | 0x12 | 0x34 | 0x56 | 0x78 |
| Length | 0x12345678 | | | |

- **Type Record**

  Type Record is optional in a Smart Poster Record. When the URI connects to an external download, the Type Record can first notify the NFC device so that the NFC device can evaluate whether it has the capability to process it.

## 10.7 Objective

1. Understand NDEF format and its applications.

## 10.8 Experiment Principle

In this chapter's experiment, we will first use the NFC Trainer Application Program to create NDEF Message, and then connects to the KL-95019 Antenna Module via the personal computer (PC) to emulate the NFC Forum Tag. In the meanwhile, by connecting to the KL-95018 Antenna Module as reader/writer, we are able to read and analyze NDEF Message through the NFC Trainer Application Program. As shown in (Figure 10-1). The experiment in this chapter can allow students to learn about the NDEF format and its applications.

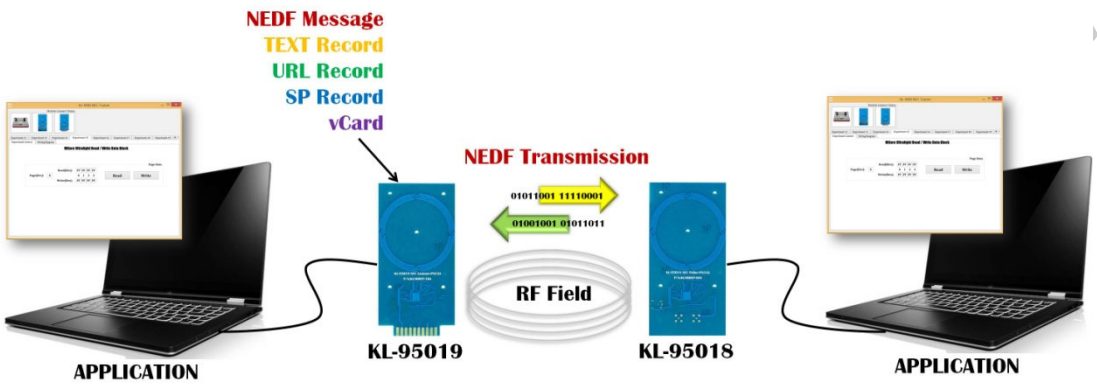



Figure 10-1

## 10.9 Experimental Equipment Layout

Different experimental equipment will be used for the different experiments set in the KL-900E NFC Trainer; therefore, this section will give a simple description about the experimental equipment and layout methods that will be used for this chapter. We will also give simple introductions on the design and positions of each piece of equipment; the equipment names will also be unified in this section for when they are mentioned in the experiment procedures.

This chapter is an experiment that uses the KL-95018 Antenna Module and the KL-95019 Antenna Module combination, and will use the PC as an aid. We can use the KL-95011 NFC Trainer Main Unit to integrate and connect the experiment modules and the PC to increase the convenience for the experiment and also reduce other environmental variables for the experiment. As shown in (Figure 10-2).

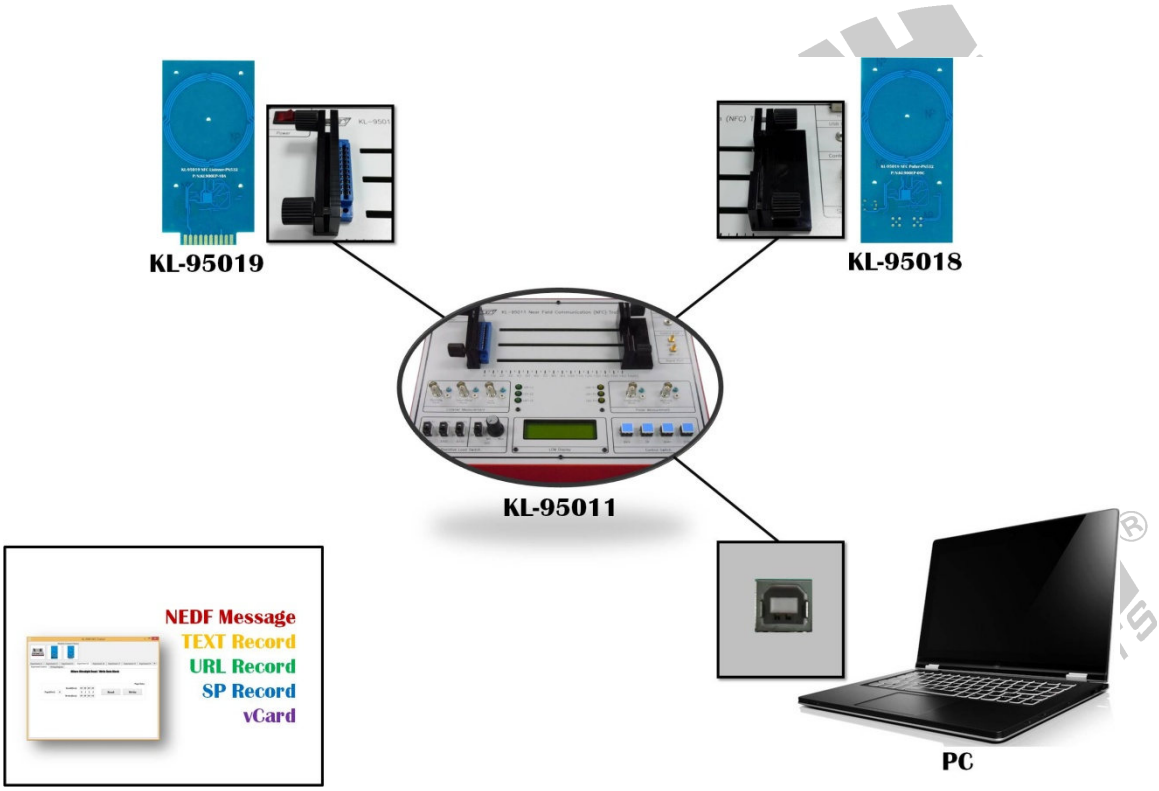
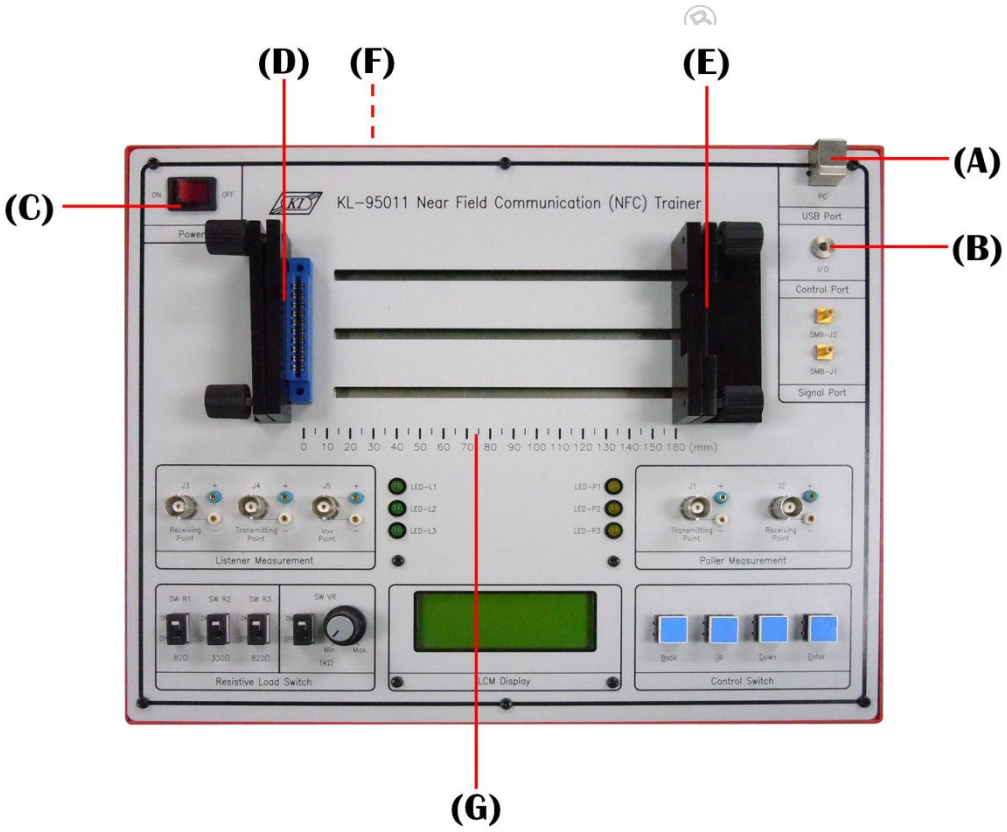
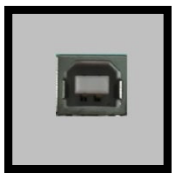



Figure 10-2

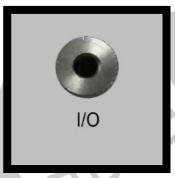
- **KL-95011 NFC Trainer Main Unit**




| Item | Name | Description |
|---|---|---|
| (A) | KL-95011 USB Port | Connects to the USB port of the PC |
| (B) | KL-95011 I/O Port | Connects to the Poller Antenna Module I/O Port |
| (C) | KL-95011 Power Switch | Power switch |
| (D) | KL-95011 Listener Holder | Holds the NFC Listener Antenna Module |
| (E) | KL-95011 Poller Holder | Holds the NFC Poller Antenna Module |
| (F) | KL-95011 Power Cord Connector | Connects to external power |
| (G) | KL-95011 Measurement Scale | Measures the distance of module |



(A)



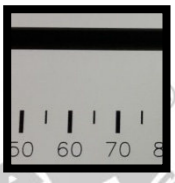

(B)


(C)

(D)

(E)

(F)

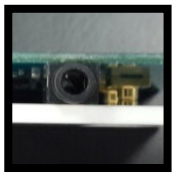



(G)

- **NFC Poller Antenna Module**

| Item | Name | Description |
|---|---|---|
| **(A)** | KL-95018 Antenna Module | Send/receive signal |
| **(B)** | Poller Antenna Module I/O Port | Connects to the KL-95011 I/O Port |

(A)



(B)

- **NFC Listener Antenna Module**

| Item | Name | Description |
|---|---|---|
| **(A)** | KL-95019 Antenna Module | Send/receive signal |

(A)

- **Experiment Connection Leads**

| Item | Name | Description |
|---|---|---|
| **(A)** | USB Cable | Connects the USB Port of the PC and the KL-95011 USB Port |
| **(B)** | I/O Cable | Connects the Poller Antenna Module I/O Port and the KL-95011 I/O Port |
| **(C)** | Power Cord | Connects the external power and the KL-95011 Power Cord Connector |

**(A)**

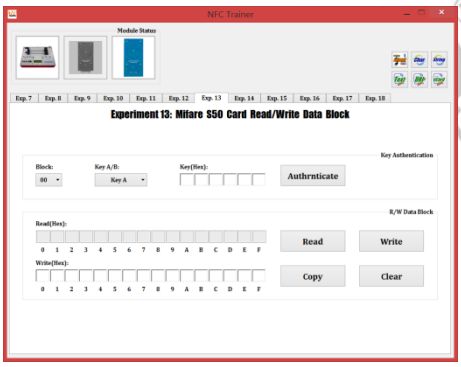
**(B)**

**(C)**

- **Application Program**

| Item | Name | Description |
|---|---|---|
| **(A)** | PC | Operating system Windows XP or above |
| **(B)** | NFC Trainer Application Program | Application program |

**(A)**




**(B)**

## 10.10 Experiment and Results

### Experiment (15) NFC Data Exchange Format

In this experiment, we will first use the NFC Trainer Application Program to create NDEF Text Record, NDEF URI Record and NDEF Smart Poster Record, and then package them into NDEF Message respectively, then connects to the KL-95019 Antenna Module via the PC to emulate the NFC Forum Tag. In the meanwhile, by connecting to the KL-95018 Antenna Module as reader/writer, we are able to read and analyze NDEF Message through the NFC Trainer Application Program.

#### Step 1 KL-95011 NFC Trainer Main Unit installation

1. Flip the KL-95011 Power Switch to OFF.

2. First use the Power Cord to connect to the KL-95011 Power Cord Connector, and then connect it to an external power. As shown in (Figure 10-3) and (Figure 10-4).

Figure 10-3

Figure 10-4

3. First use the USB Cable to connect to the KL-95011 USB Port, and then connect it to the USB port of the PC. As shown in (Figure 10-5) and (Figure 10-6).


Figure 10-5

Figure 10-6

**Step 2 KL-95018 Antenna Module installation**

1. Use the I/O Cable to connect to the Poller Antenna Module I/O Port, and then connect it to the KL-95011 I/O Port. As shown in (Table 10-9), (Figure 10-7) and (Figure 10-8).

Table 10-9

| KL-95011 NFC Trainer Main Unit | NFC Poller Antenna Module |
|---|---|
| KL-95011 I/O Port | Poller Antenna Module I/O Port |

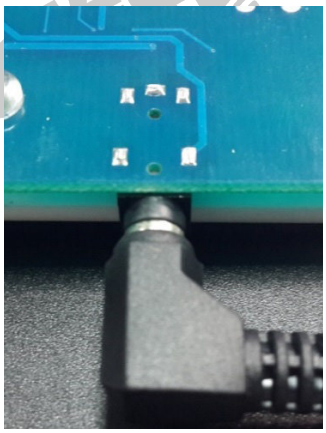


Figure 10-7


Figure 10-8

2. Fix the KL-95018 Antenna Module on the KL-95011 Poller Holder.
3. Move the KL-95011 Poller Holder to the right-most side. As shown in (Figure 10-9).

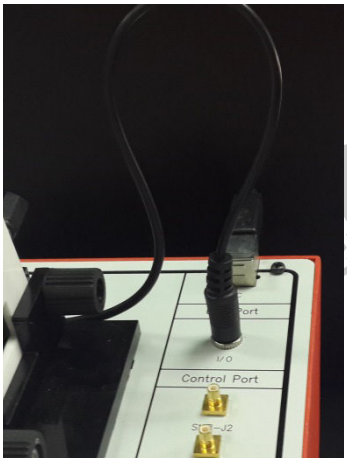



Figure 10-9

**Step 3 KL-95019 Antenna Module installation**

1. Fix the the KL-95019 Antenna Module above the KL-95011 Listener Holder, as shown in (Figure 10-10).

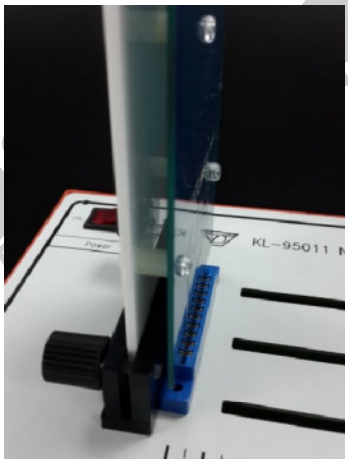



Figure 10-10

**Step 4 Adjust the distance between the NFC Poller Antenna Module and the NFC Listener Antenna Module**

1. Move the NFC Poller Antenna Module to the position on the KL-95011 Measurement Scale marked 20 (mm), as shown in (Figure 10-11).

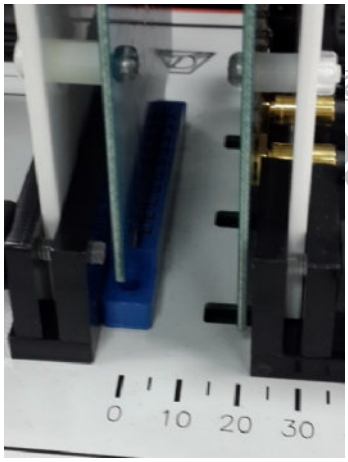



Figure 10-11

## Step 5 Open NFC Trainer Application Program

1. Flip the KL-95011 Power Switch to ON.

2. Open the NFC Trainer Application Program on the PC.

3. Select experiment item. As shown in (Figure 10-12).

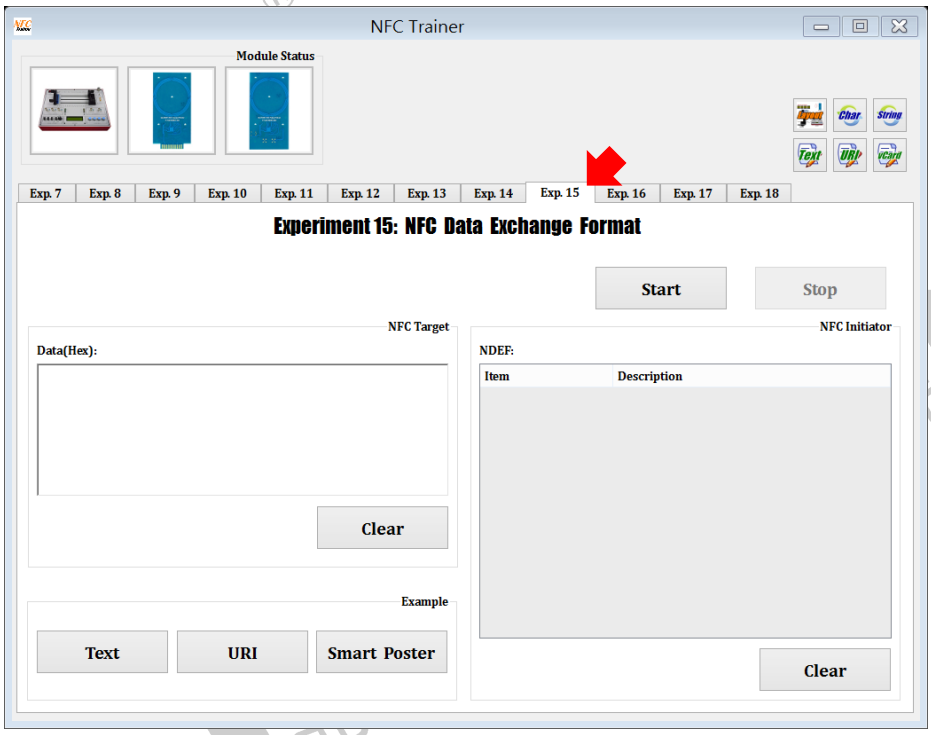



Figure 10-12

## Step 6 Open NDEF Text Record Creator

1. Click "NDEF Text Record Creator" with the mouse to open NDEF Text Record Creator. As shown in (Figure 10-13).

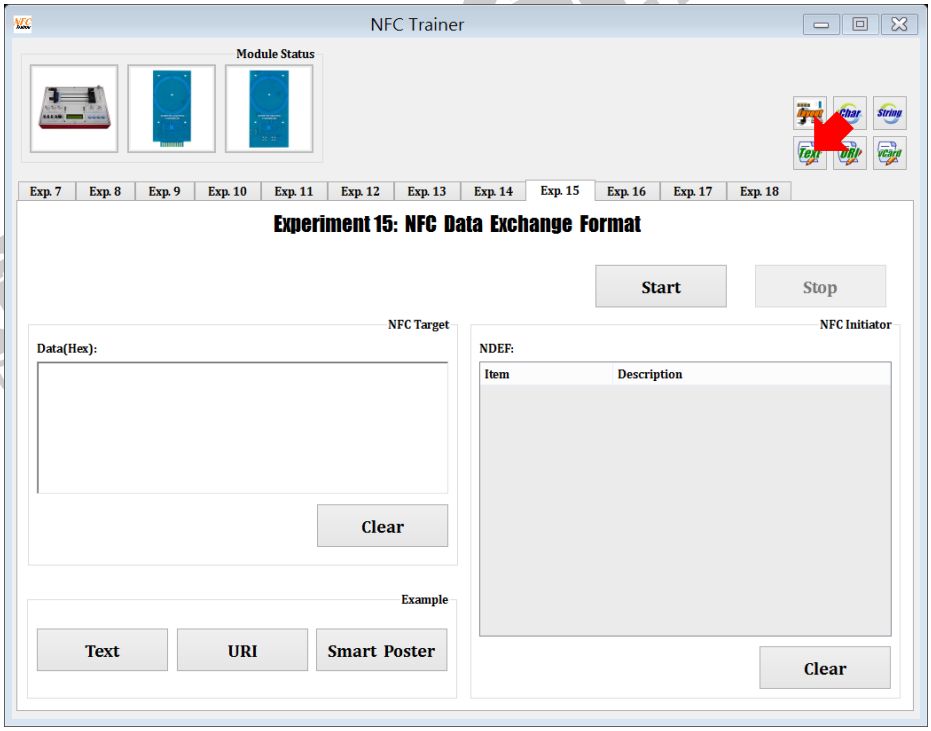



Figure 10-13

## Step 7 Create NDEF Text Record

1. Click "Example" with the mouse to generate an example.

2. Click "Create" with the mouse to create Text Record. As shown in (Figure 10-14).

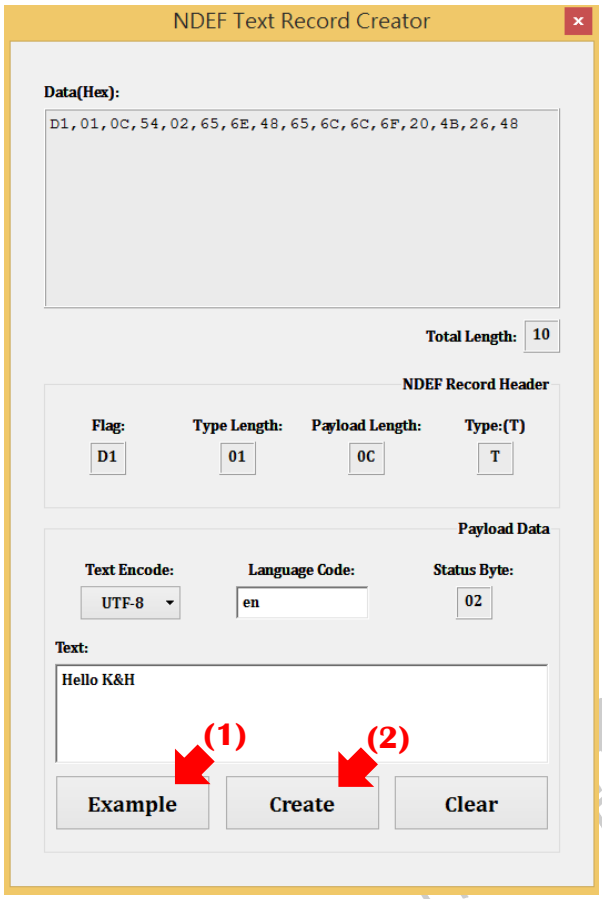



Figure 10-14

## Step 8 Analyze NDEF Message information

1. The content of NDEF Message is shown in (Table 10-10).

Table 10-10

| D1 | 01 | 0C | 54 | 02 | 65 | 6E | 48 | 65 | 6C | 6C | 6F | 20 | 4B | 26 | 48 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

2. Analyze Flag. As shown in (Table 10-11).

Table 10-11

| Flag | | | | | | | |
|---|---|---|---|---|---|---|---|
| D1 | | | | | | | |
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| MB | ME | CF | SR | IL | TNF | | |
| 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |

- The content of Flag is 0xD1.
  This is the only Record in the NDEF Message.
  The next Record and the current Record has no chunk relationship.
  This Record does not include ID Length and Payload ID.
  This Record is one of NFC Forum Well Known Types.

3. Analyze the Length. As shown in (Table 10-12).

Table 10-12

| Length | |
|---|---|
| The Length of Payload Type | **01** |
| The Length of Payload Data | **0C** |
| The Length of Payload ID | N/A |

- The content of Payload Type Length is 0x01.
  The Length of Payload Type is 1 byte.

- The content of Payload Data Length is 0x0C.
  The Length of Payload Data is 12 bytes.

4. Analyze Payload. As shown in (Table 10-13).

Table 10-13

| Payload | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Payload Type | **54** | | | | | | | | | | | |
| Payload ID | N/A | | | | | | | | | | | |
| Payload Data | **02** | **65** | **6E** | **48** | **65** | **6C** | **6C** | **6F** | **20** | **4B** | **26** | **48** |

- The Content of Payload Type is 0x54.
- The Payload Type means "T".
- The Payload Data is a T-Type data of NFC Forum Well Know Types.
- The Payload Data must be analyzed in accordance with RTD Text Specifications.

**Step 9 Analyze RTD Text Record**

1. The content of RTD Text Record is shown in (Table 10-14).

Table 10-14

| 02 | 65 | 6E | 48 | 65 | 6C | 6C | 6F | 20 | 4B | 26 | 48 |
|---|---|---|---|---|---|---|---|---|---|---|---|

2. Analyze Flag. As shown in (Table 10-15).

Table 10-15

| Flag | | | | | | | |
|---|---|---|---|---|---|---|---|
| 02 | | | | | | | |
| Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| UTF 8/16 | Reserved | The length of Language Code | | | | | |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

- The Content of Flag is 0x02.
  The Payload Data is uses UTF 8 coding.
  The Length of Language Code is 2 bytes.

3. Analyze Language Code. As shown in (Table 10-16).

Table 10-16

| Language | | | | |
|---|---|---|---|---|
| Language Code | | 65 | 6E | |

- The Content of Language Code is 0x65,0x6E.
- The Language Code means "en".

4. Analyze Text Data. As shown in (Table 10-17).

Table 10-17

| Content | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Text | | 48 | 65 | 6C | 6C | 6F | 20 | 4B | 26 | 48 | |

- The Text Data means "Hello K&H".

## Step 10 NDEF Message settings

1. Copy the output content of NDEF Text Record Creator to the input window of the main window. As shown in (Figure 10-15).

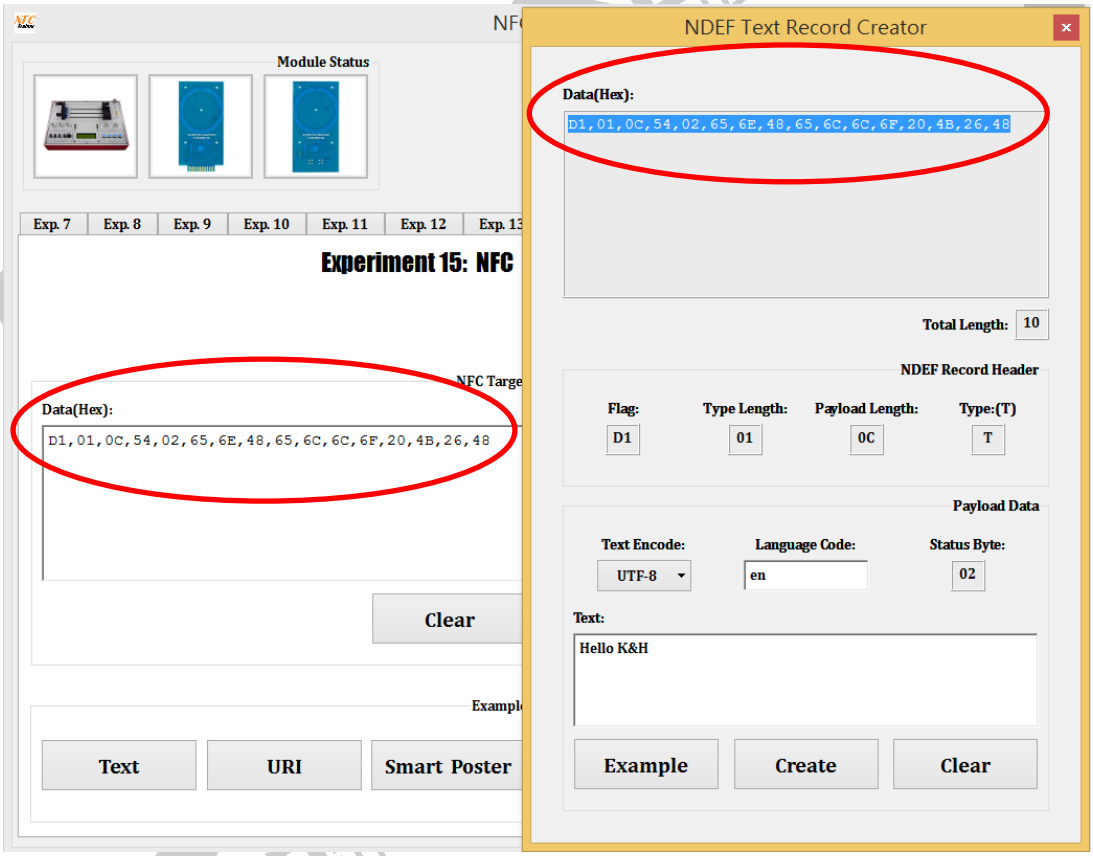



Figure 10-15

## Step 11 Reading NDEF Message

1. Click "Start" with the mouse to read the NDEF Message.

2. Double-click "Tree Menu" with the mouse to expand the menu. As shown in (Figure 10-16).

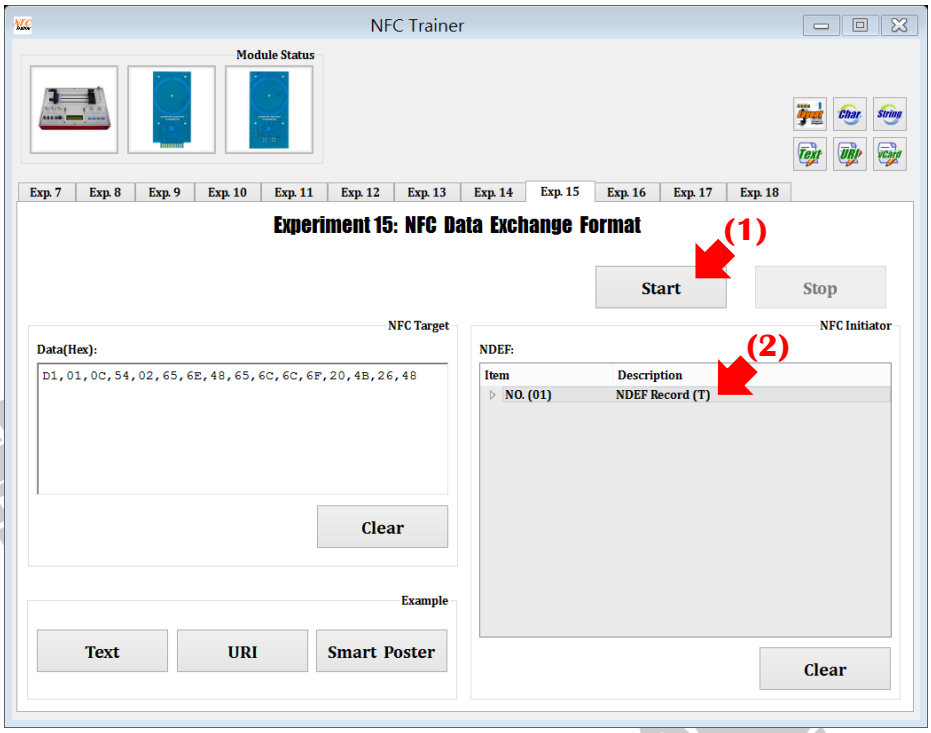



Figure 10-16

3. Operation screen, as shown in (Figure 10-17).

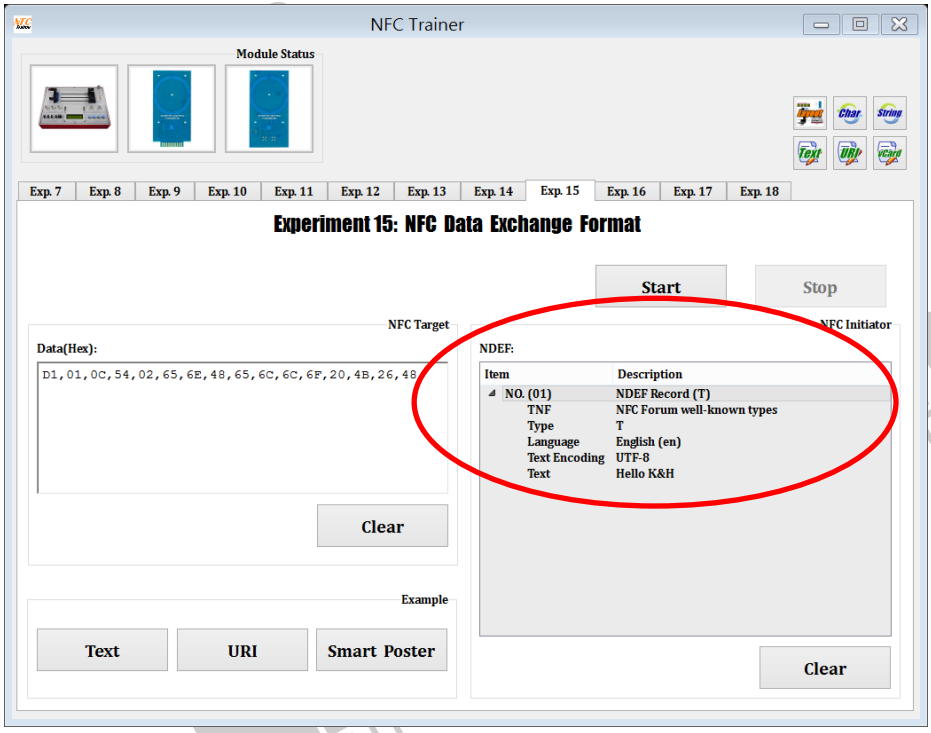



Figure 10-17