

# OpenHarmony 基础外设开发-----NFC



# 目 录

## CONTENTS

- [ 01 ] 什么是NFC
- [ 02 ] NFC工作原理
- [ 03 ] NFC接口
- [ 04 ] 如何控制NFC

# 01

## 什么是NFC

近场通信（Near Field Communication，简称NFC），是一种新兴的技术，使用了NFC技术的设备（例如移动电话）可以在彼此靠近的情况下进行数据交换，是由非接触式射频识别（RFID）及互连互通技术整合演变而来的，通过在单一芯片上集成感应式读卡器、感应式卡片和点对点通信的功能，利用移动终端实现移动支付、电子票务、门禁、移动身份识别、防伪等应用。

## 02

# NFC工作原理

NFC是一种短距高频的无线电技术，NFCIP-1标准规定NFC的通信距离为10厘米以内，运行频率13.56MHz，传输速度有106Kbit/s、212Kbit/s或者424Kbit/s三种。NFCIP-1标准详细规定NFC设备的传输速度、编解码方法、调制方案以及射频接口的帧格式，此标准中还定义了NFC的传输协议，其中包括启动协议和数据交换方法等。

## 03

# NFC接口

NFC接口的头文件

`/vendor/lockzhiner/rk2206/samples/b0_nfc/include/nfc.h`

OpenHarmony基础外设开发NFC接口主要有：

- (1) 初始化和注销NFC；
- (2) 控制NFC。

## 03

## NFC接口

功能分类	接口名	功能描述
初始化、销毁NFC	nfc_init	NFC模块初始化
	nfc_deinit	NFC模块注销
控制NFC	nfc_store_uri_http	向NFC写入URI信息
	nfc_store_text	向NFC写入txt信息

## 03

# NFC接口

```
unsigned int nfc_init(void);
```

该函数主要功能是初始化NFC模块。

返回0为成功，其余为失败。

```
unsigned int nfc_deinit(void);
```

该函数主要功能是NFC模块注销。

返回0为成功，其余则失败。

## 03

# NFC接口

```
bool nfc_store_uri_http(RecordPosEnu position, uint8_t *http);
```

该函数主要功能是向NFC写入URI信息。

- 参数position：信息标识；
- 参数http：需要写入的网络地址字符串。

返回true为成功，false则失败。



## 03

# NFC接口

```
bool nfc_store_text(RecordPosEnu position, uint8_t *text);
```

该函数主要功能是向NFC写入TXT信息。

- 参数position：信息标识；
- 参数text：需要写入的TXT字符串。

返回true为成功，false则失败。

## 04

# 如何控制NFC

### 1、打开sdk下面路径的文件

```
vendor/lockzhiner/rk2206/samples/b2_nfc/nfc_example.c
```

### 2、创建任务

在nfc\_example函数中，通过LOS\_TaskCreate函数创建nfc\_process任务。

```
task.pfnTaskEntry = (TSK_ENTRY_FUNC)nfc_process;
```

```
task.uwStackSize = 10240;
```

```
task.pcName = "nfc process";
```

```
task.usTaskPrio = 24;
```

```
ret = LOS_TaskCreate(&thread_id, &task);
```

# 如何控制NFC

```
void nfc_process(void)
```

```
{
```

```
    unsigned int ret = 0;
```

```
    /* 初始化NFC设备 */
```

```
    nfc_init();
```

```
    ret = nfc_store_text(NDEFFirstPos, (uint8_t *)TEXT);
```

```
    if (ret != 1) {
```

```
        printf("NFC Write Text Failed: %d\n", ret);
```

```
    }
```

```
    ret = nfc_store_uri_http(NDEFLastPos, (uint8_t *)WEB);
```

```
    if (ret != 1) {
```

```
        printf("NFC Write Url Failed: %d\n", ret);
```

```
    }
```

### 3、修改编译脚本

修改 `vendor/lockzhiner/rk2206/sample` 路径下 `BUILD.gn` 文件，指定 `nfc_example` 参与编译。

```
"/b2_nfc:nfc_example",
```

修改 `device/lockzhiner/rk2206/sdk_liteos` 路径下 `Makefile` 文件，添加 `-lnfc_example` 参与编译。

```
hardware_LIBS = -lhal_iohardware -lhardware -lnfc_example
```

### 4、编译固件

```
hb set -root .
```

```
hb set
```

```
hb build -f
```

## 5、烧写固件

## 6、通过串口查看结果

运行结果

```
=====NFC Example=====
```

```
Please use the mobile phone with NFC function close to the development board!
```

```
.....
```

# 谢谢聆听

单击此处添加副标题内容