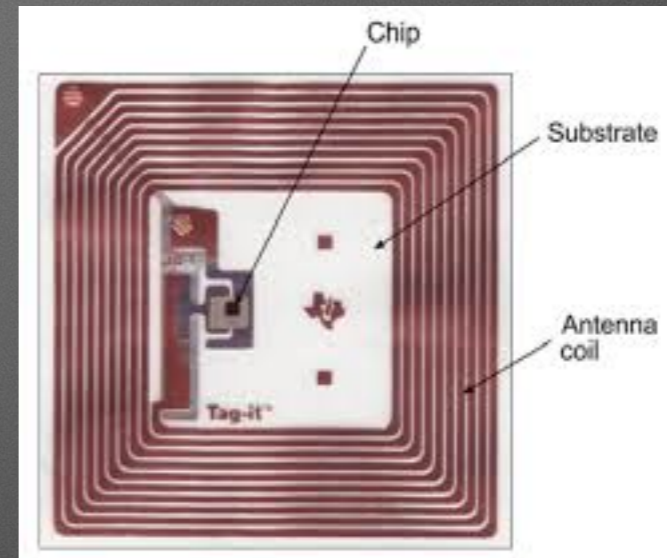


Designing a NFC Tag Reader with WebUSB support

基础篇



常见卡类型举例

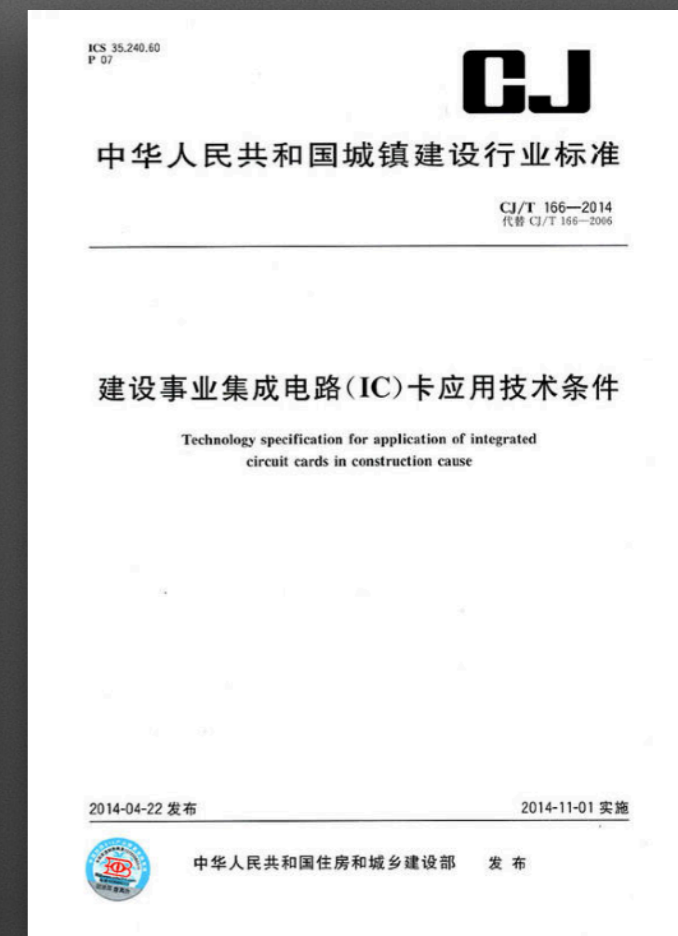
| 频率 | 标准 | ID/IC 卡 (TAG) | CPU卡 (智能卡) |
|------------------|---------------------|---|--------------|
| 125KHZ 134KHZ | ISO 11785 |  | |
| 13.56 MHZ | ISO 14443 TypeA | Mifare S50 (M1) | 各地交通卡、护照、银行卡 |
| | ISO 14443 TypeB | THR1064奥运会 门票 | 清华校园卡、身份证? |
| | ISO 18092 FeliCa |  | |
| | ISO 15693 |  | |
| 860-960 MHZ | ISO 18000-6 |  | |



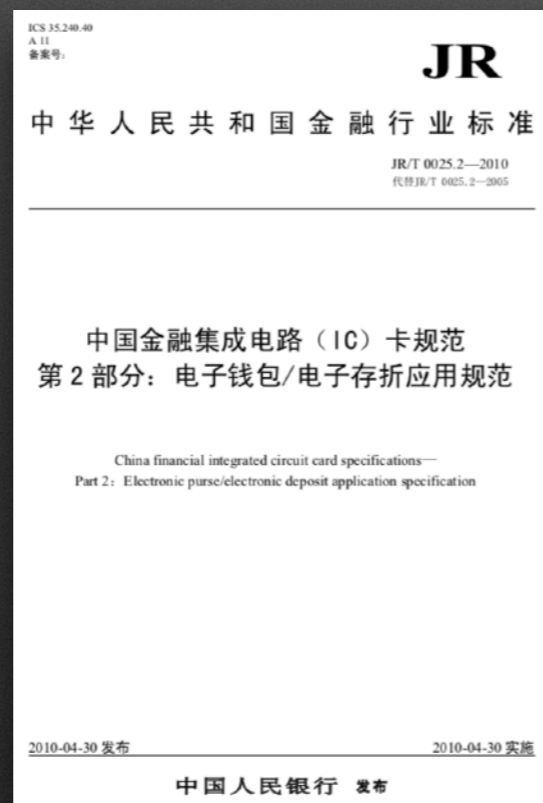
JT/T 978-2015



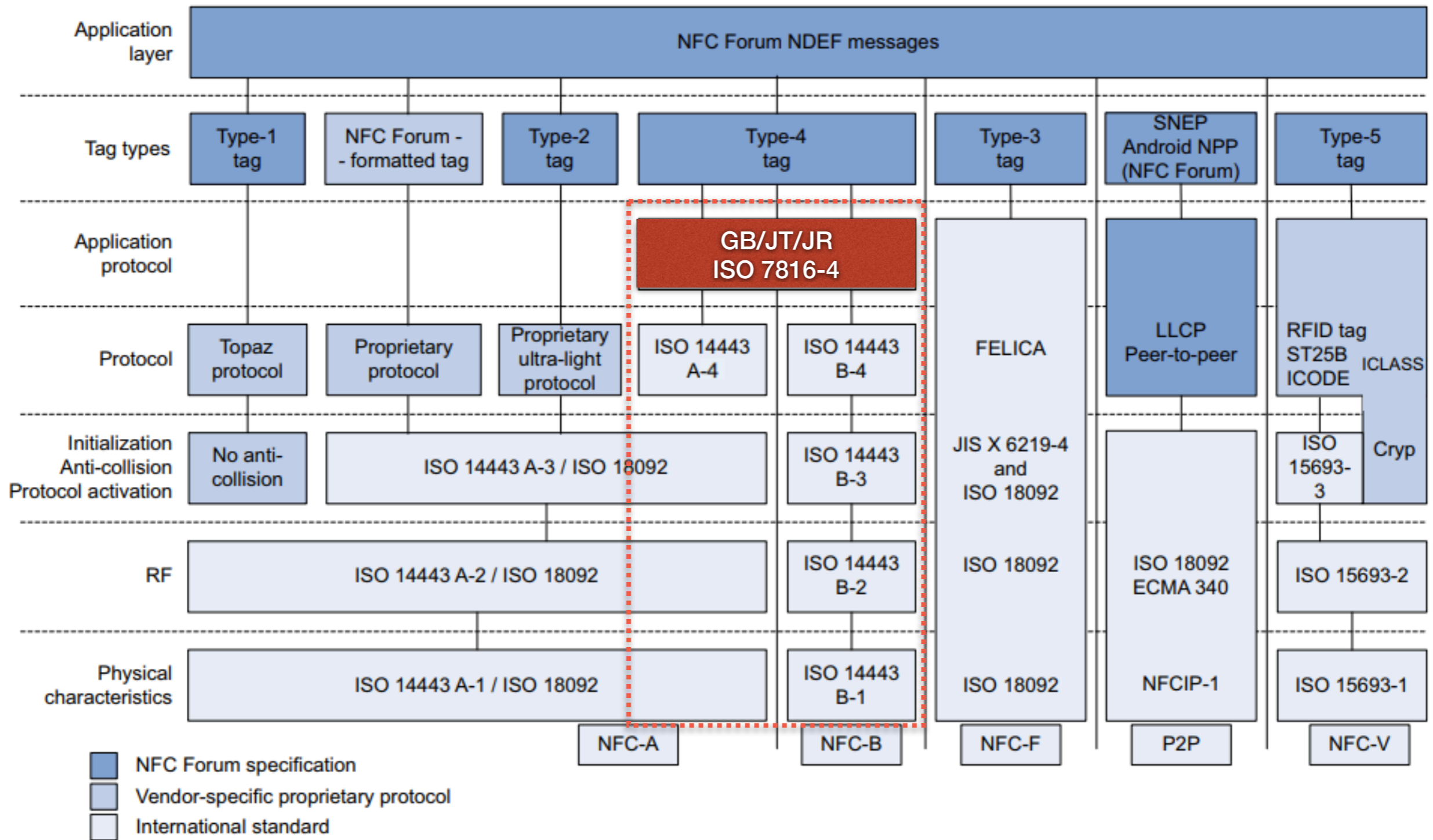
GB/T 31778



CJ/T 166



JR/T 0025.9-2010



MSv38937V1

文件结构 (CJ/T 166)

表11 公共应用基本信息文件

| | | | |
|------------|-------------------|----|---------------|
| 文件标识 (SFI) | 0x15 | | |
| 文件类型 | 二进制文件 | | |
| 文件大小 | 001EH | | |
| 文件存取控制 | 读 = 自由 | | 改写 = DAMK线路保护 |
| 字节 | 数据元 | 长度 | 格式 |
| 01~02 | 发卡方代码 | 2 | BCD |
| 03~04 | 城市代码 | 2 | HEX |
| 05~05 | 算法支持 | 1 | BCD |
| 06~06 | 行业代码 | 1 | BCD |
| 07~08 | 占位符 (0000) | 2 | HEX |
| 09~09 | 应用类型标识 (启用标志) | 1 | BCD |
| 10~10 | 应用版本 | 1 | BCD |
| 11~12 | 互通标识 | 2 | HEX |
| 13~20 | 应用序列号 | 8 | HEX |
| 21~24 | 应用启动日期 (YYYYMMDD) | 4 | BCD |
| 25~28 | 应用有效日期 (YYYYMMDD) | 4 | BCD |
| 29~30 | 预留 | 2 | HEX |

注1: 互通标识为本城市的城市代码;

注2: 应用类型标识 (启用标志) 00为未启用, 非00为启用; 预留部分不允许占用。

文件结构 (CJ/T 166)

表12 个人基本信息文件

| | | | | | | | |
|-------|---------|--------|----|------|------------|---------------|-------|
| 文件标识符 | 00 16 | SFI | 16 | 文件长度 | 0040H (64) | 文件结构 | 二进制文件 |
| 读控 | | PIN 认证 | | 写控 | | DAMK (明文+MAC) | |
| 字节 | 数据元 | | | 长度 | 格式 | 说明 | |
| 01-01 | 持卡人类型标识 | | | 1 | BCD | —— | |
| 02-02 | 持卡人职工标识 | | | 1 | BCD | —— | |
| 03-22 | 持卡人姓名 | | | 20 | ASC | —— | |
| 23-54 | 持卡人证件号码 | | | 32 | ASC | —— | |
| 55-55 | 持卡人证件类型 | | | 1 | BCD | —— | |
| 空间预留 | 9 字节 | | | | | | |

文件结构 (CJ/T 166)

表17 本地消费交易明细文件

| | | | |
|------------|-----------------|--------------|-----|
| 文件标识 (SFI) | 0x18 | | |
| 文件类型 | 循环记录文件 | | |
| 记录长度 | 0017H | | |
| 文件存取控制 | 读 = 自由 | 改写 = COS内部操作 | |
| 字节 | 数据元 | 长度 | 格式 |
| 01~02 | 电子钱包消费 | 2 | HEX |
| 03~05 | 预留 | 3 | HEX |
| 06~09 | 交易金额 | 4 | HEX |
| 10~10 | 交易类型 | 1 | BCD |
| 11~16 | 交易终端编号 | 6 | HEX |
| 17~20 | 交易日期 (YYYYMMDD) | 4 | BCD |
| 21~23 | 交易时间 (HHMMSS) | 3 | BCD |

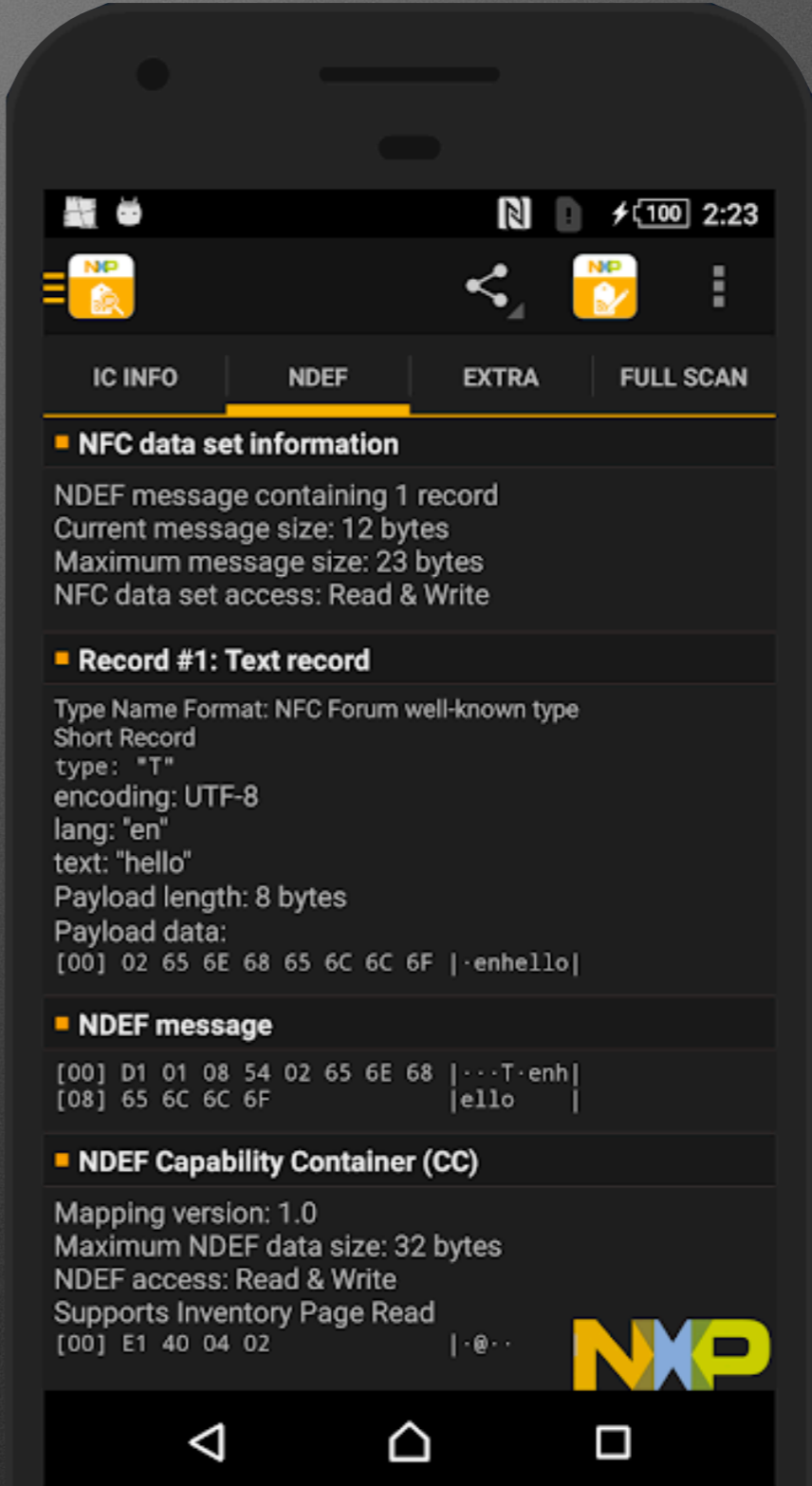
工具篇

TagInfo App

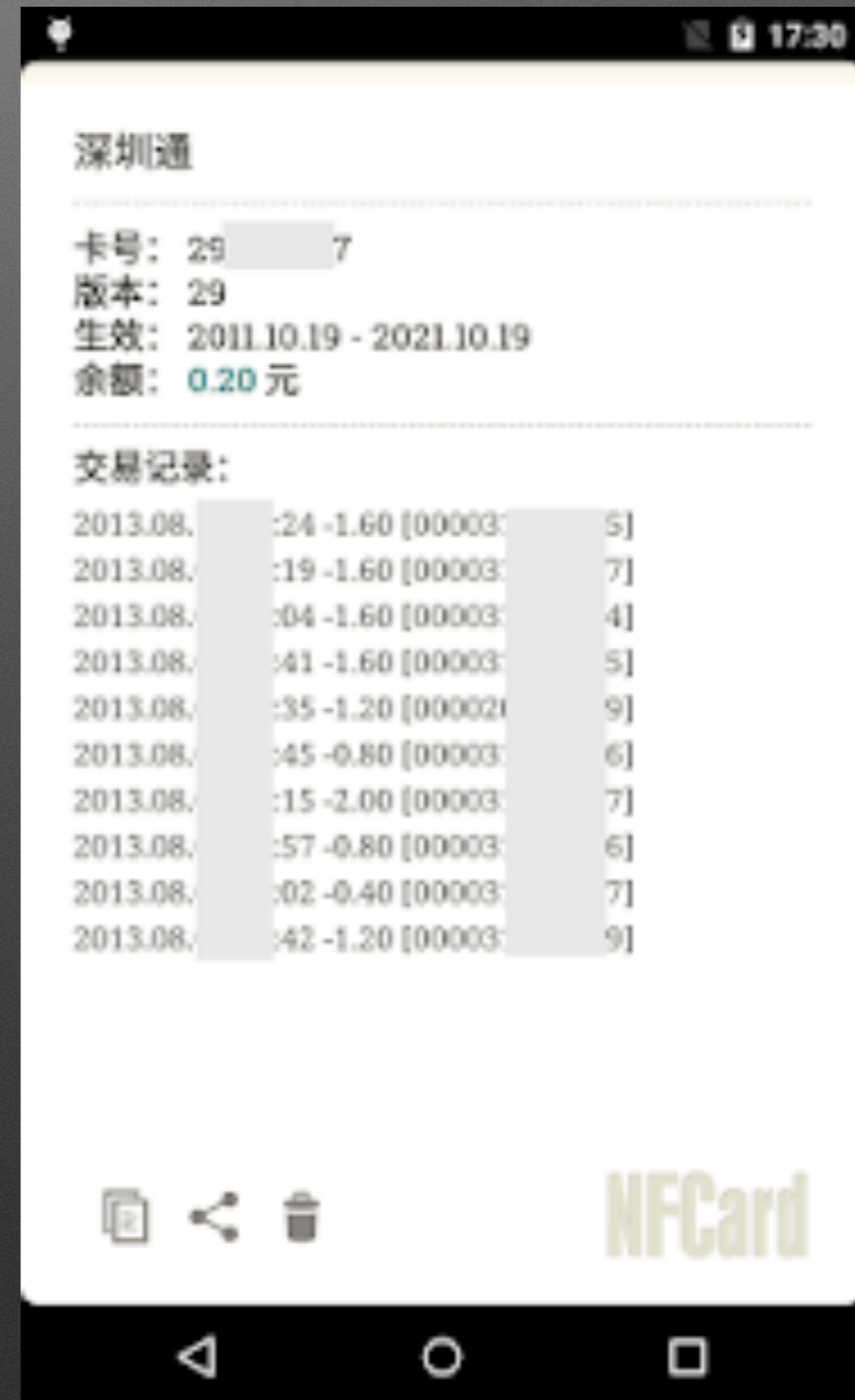


New on App Store: NFC TagInfo by NXP

Download now!
Read & interact with NFC tags on iOS 11

The promotional banner includes the app icon, an illustration of a hand holding a smartphone near an NFC tag, and the NXP logo.

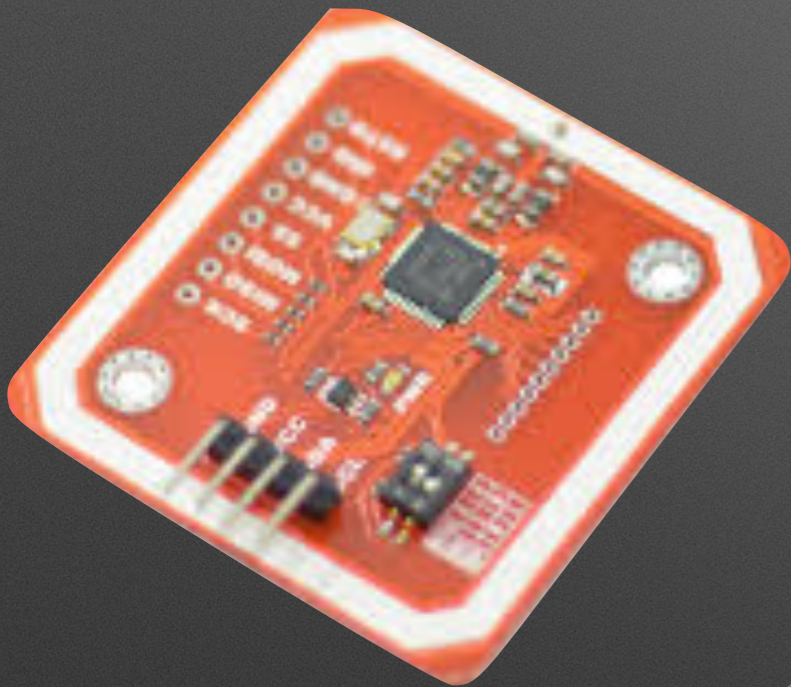
NFC Card App



<https://github.com/z4yx/nfc card>

读卡器

PN532+Arduino

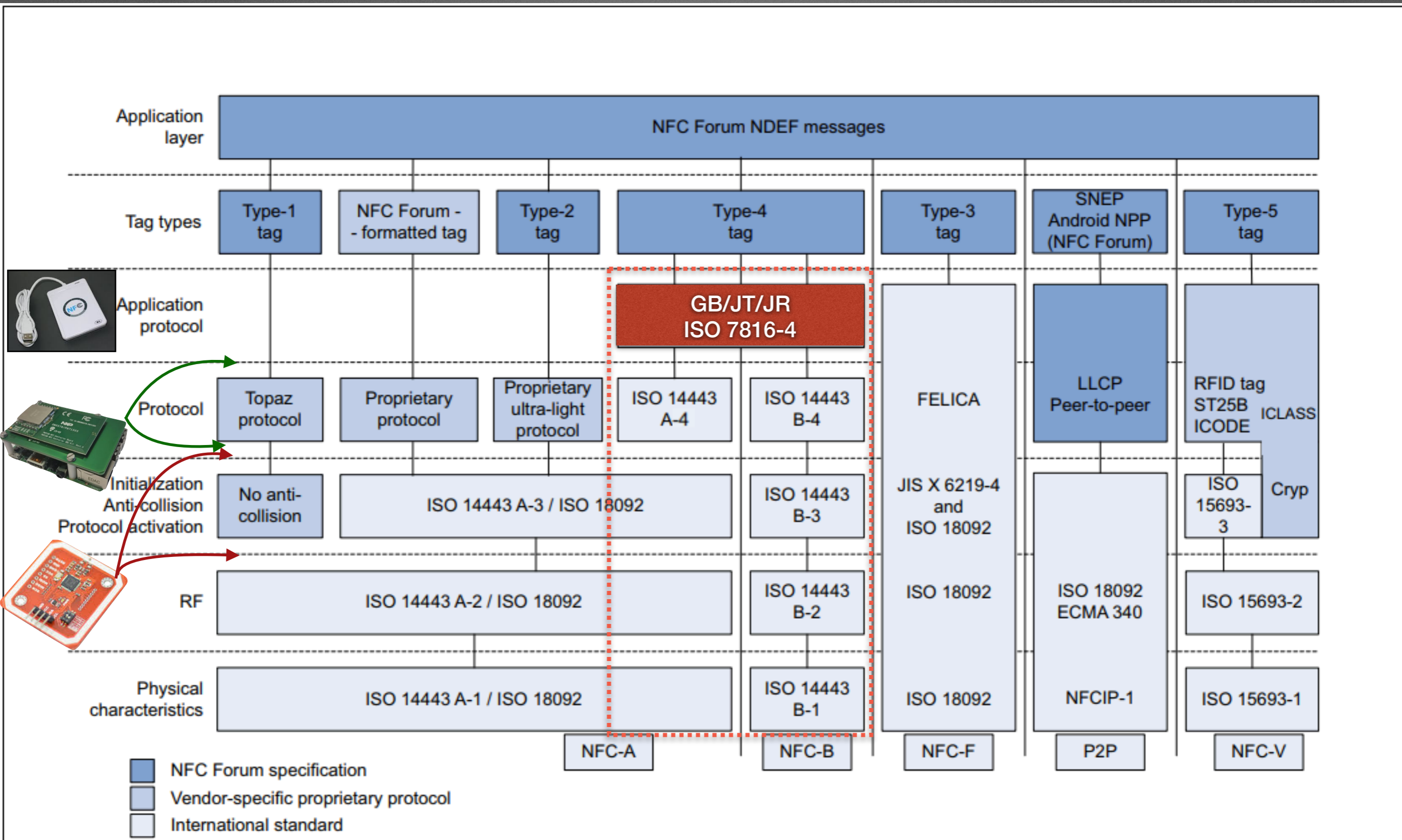


ACR122U+PC



PN7150+Pi





MSv38937V1

编程篇

PC上的读卡应用

- 开发本地应用

- 串口 / USB串口

- USB-CCID

- Bluetooth

- 跨平台问题

- 驱动问题

- 开发Web应用

- ActiveX

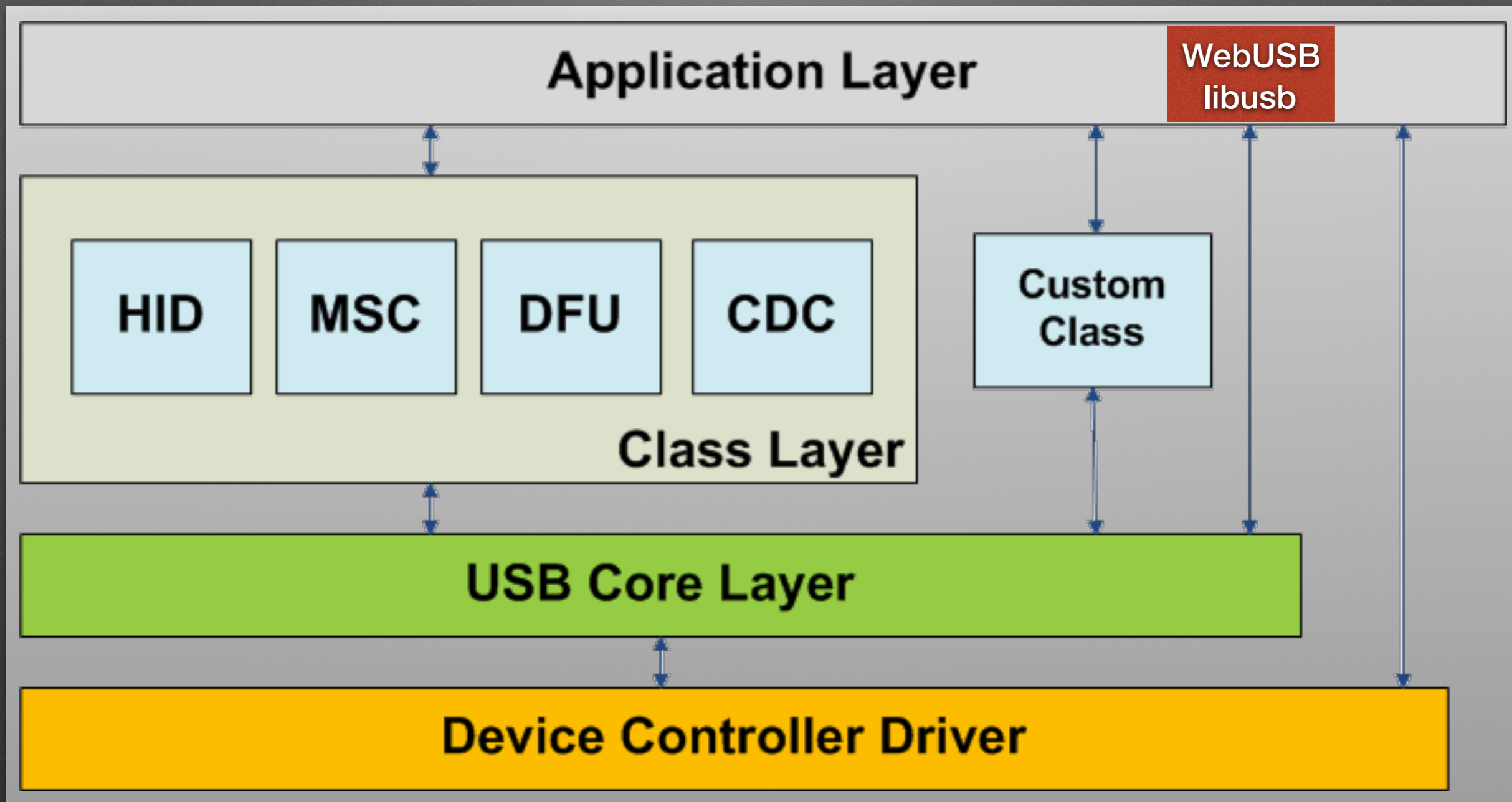
- WebUSB

- WebBluetooth

IE-Only

Chrome-Only

Chrome-Only



WebUSB API (Draft) - Device

| Offset | Field | Size | Value | Description |
|--------|------------------------|------|----------|---|
| 0 | bLength | 1 | Number | Size of this descriptor. Must be set to 24. |
| 1 | bDescriptorType | 1 | Constant | DEVICE CAPABILITY descriptor type ([USB31] Table 9-6). |
| 2 | bDevCapabilityType | 1 | Constant | PLATFORM capability type ([USB31] Table 9-14). |
| 3 | bReserved | 1 | Number | This field is reserved and shall be set to zero. |
| 4 | PlatformCapabilityUUID | 16 | UUID | Must be set to {3408b638-09a9-47a0-8bfd-a0768815b665}. |
| 20 | bcdVersion | 2 | BCD | Protocol version supported. Must be set to 0x0100. |
| 22 | bVendorCode | 1 | Number | bRequest value used for issuing WebUSB requests. |
| 23 | iLandingPage | 1 | Number | URL descriptor index of the device's landing page . |

Platform Descriptor Control Transfer

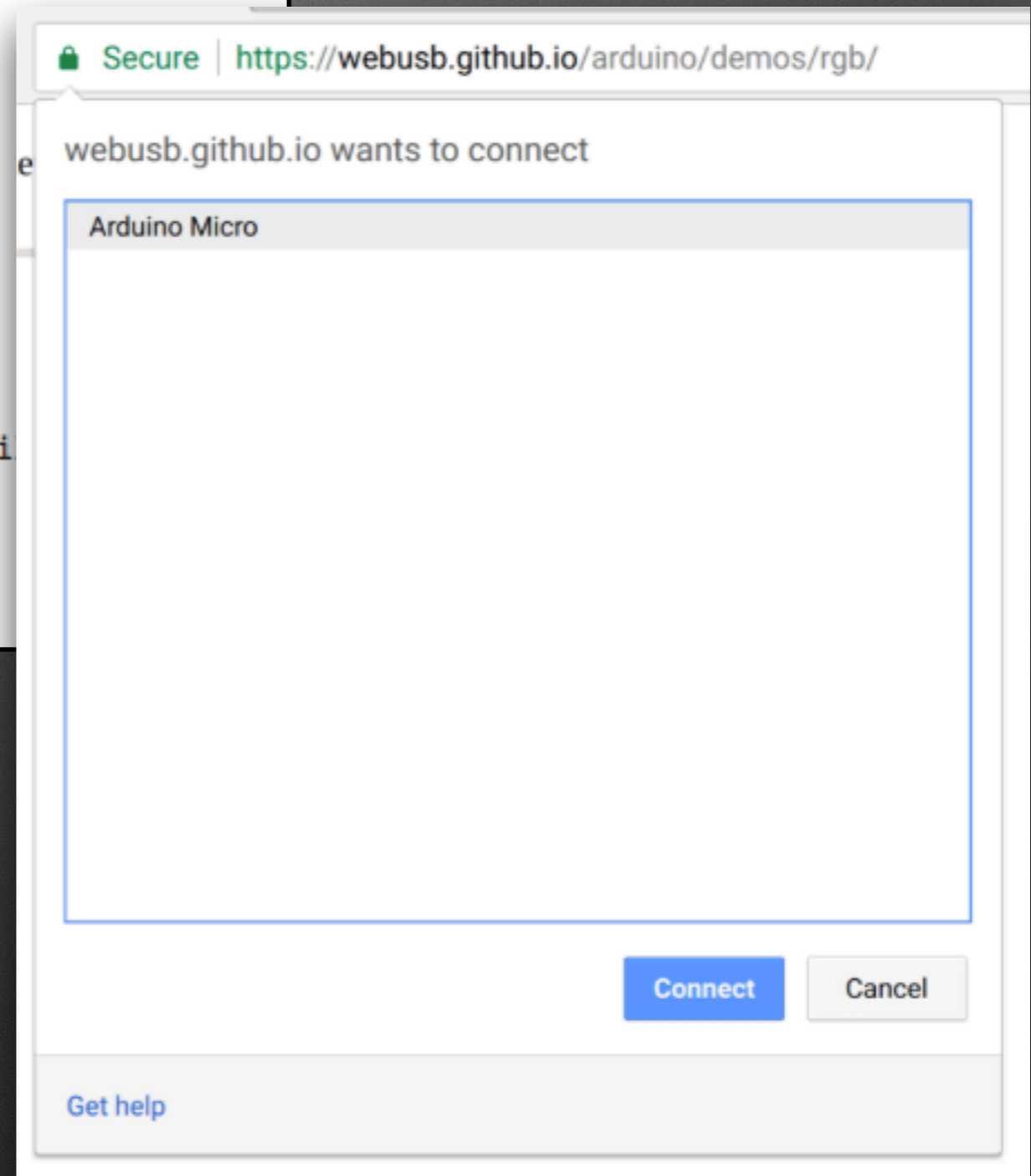
| bmRequestType | bRequest | wValue | wIndex | wLength | Data |
|---------------|-------------|------------------|---------|-------------------|------------|
| 11000000B | bVendorCode | Descriptor Index | GET_URL | Descriptor Length | Descriptor |

| Offset | Field | Size | Value | Description |
|--------|-----------------|----------|--------|--|
| 0 | bLength | 1 | | |
| 1 | bDescriptorType | 1 | | |
| 2 | bScheme | 1 | | |
| 3 | URL | Variable | String | UTF-8 encoded URL (excluding the scheme prefix). |



WebUSB API (Draft) - Host

```
serial.requestPort = function() {  
  const filters = [  
    { 'vendorId': 0x2341, 'productId': 0x8036 },  
    { 'vendorId': 0x2341, 'productId': 0x8037 },  
    { 'vendorId': 0x2341, 'productId': 0x804d },  
    { 'vendorId': 0x2341, 'productId': 0x804e },  
    { 'vendorId': 0x2341, 'productId': 0x804f },  
    { 'vendorId': 0x2341, 'productId': 0x8050 },  
  ];  
  return navigator.usb.requestDevice({ 'filters': fi  
    device => new serial.Port(device)  
  });  
}
```



WebUSB API (Draft) - Host

```
return this.device_.open()
  .then(() => {
    if (this.device_.configuration === null) {
      return this.device_.selectConfiguration(1);
    }
  })
  .then(() => this.device_.claimInterface(2))
  .then(() => this.device_.selectAlternateInterface(2, 0))
  .then(() => this.device_.controlTransferOut({
    'requestType': 'class',
    'recipient': 'interface',
    'request': 0x22,
    'value': 0x01,
    'index': 0x02}))
  .then(() => {
    readLoop();
  });
```

libusb_set_configuration

libusb_claim_interface

libusb_set_interface_alt_setting

libusb_control_transfer

WebUSB API (Draft) - Host

```
let readLoop = () => {  
  this.device_.transferIn(5, 64).then(result => {  
    this.onReceive(result.data);  
    readLoop();  
  }, error => {  
    this.onReceiveError(error);  
  });  
};
```

libusb_bulk_transfer

```
serial.Port.prototype.send = function(data) {  
  return this.device_.transferOut(4, data);  
};
```

栗子

- <https://github.com/webusb/arduino>

WebUSB ❤️ Arduino

This repository contains an Arduino library for WebUSB-enabling your sketches. Example sketches and JavaScript code are available in the demos directory.

The WebUSB object is a copy of the Arduino SDK's built-in USB serial library. It creates a WebUSB-compatible vendor-specific interface rather than one marked as USB CDC-ACM. This prevents operating system drivers from claiming the device and making it inaccessible to the browser. This library also implements:

- The WebUSB landing page descriptor, providing a hint to the browser about what page the user should navigate to to interact with the device. In Google Chrome the presence of this descriptor causes the browser to display a notification when the device is connected. The user can click on this notification to navigate directly to the provided URL.
- Microsoft OS 2.0 Descriptors which instruct the Windows operating system (8.1 and above) to automatically the `WinUSB.sys` driver so that the browser can connect to the device.

Compatible Hardware

WebUSB requires an Arduino model that gives the sketch complete control over the USB hardware. This library has been tested with the following models:

- Arduino Leonardo
- Arduino/Genuino Micro
- Arduino/Genuino Zero
- Arduino MKR1000
- Arduino MKRZero
- Arduino MKRFox1200
- Adafruit Feather 32u4

刷卡入会

- <https://github.com/z4yx/webusb-cardreader>
- <https://github.com/tuna/registr/blob/master/static/js/card-reader.js>

加入我们

LANGUAGE ▾

姓名
张宇翔

院系 (选填)

学号 (选填)
2017210832

电话

Email

性别

男
 女

想干什么

撸代码
 办活动
 搞大新闻
 打酱油

走你