

“Nfc for Everybody” Design Document

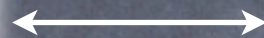
目的

アクセサリとしてNFC(今回はFeliCaのみ)のリーダー/ライターを接続し、NFC機能を持たないNexus OneやXOOMのような端末でもNFC対応アプリを動かせるようにします。

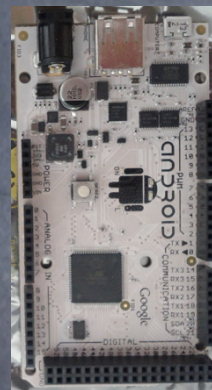
Components



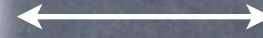
Accessory対応端末



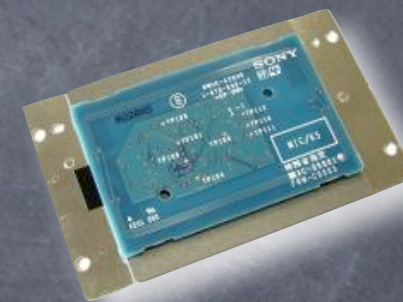
USB



ADK Board



Serial



FeliCaリーダー/ライター

RC-S620/S

できること

- ・カードから IDm(識別番号)や、NDEF Messageを読み取りIntent としてアプリへ投げる
 - ・taglet のような既存のNFC対応アプリがそのまま使える
- ↑ ここポイント

できないこと(実現方法がわかればやりたい)

- ・NDEF 以外の形式のカードの読み取り
- ・NFC対応アプリの中でも、Suica Reader のように独自にカードを読むアプリへの対応

実装の方針

Google I/O 2011 報告会で披露した、ADK を用いて FeliCa Push 送信を行うデモのコード <https://github.com/zaki50/FeliCaPushWithAndroidAccessory> をベースにする。

Flow

1. カードリーダーを初期化する
 2. カードをポーリングする
 3. カードが見つかったら IDm(と可能であれば NDEF Messages)を読み込む
 4. 読み込んだ情報をAndroid端末へ送信する
-
5. カードの情報を受信する
 6. インストールされているアプリから、適切なアプリを選択する
 7. Intent を作成し startActivity する

in Accessory

in Device

対象アプリを決定するコード(NDEF_DISCOVERED) (Android application)

```
private List<ActivityInfo> findForNdef(String mimetype) {
    final Intent i = new Intent(NfcAdapter.ACTION_NDEF_DISCOVERED);
    i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    i.setType(mimetype);
    final List<ResolveInfo> resolvedActivities;
    resolvedActivities = getPackageManager().queryIntentActivities(i,
        PackageManager.GET_META_DATA);
    final List<ActivityInfo> result = new ArrayList<ActivityInfo>();
    for (ResolveInfo resAct : resolvedActivities) {
        result.add(resAct.activityInfo);
    }
    return result;
}
```

アプリに投げるIntentを作成するコード (Android application)

```
private Intent createIntentForTag(byte[] idm, NdefMessage[] msgs)
    throws IllegalAccessException, InvocationTargetException {

    // sTagFactory is Method instance of Tag#createMockTag(...)
    final Tag tag = (Tag) sTagFactory.invoke(null, new Object[] {
        idm, //
        new int[] {
            4
        }, //
        /* TagTechnology.NFC_F */}, //
        new Bundle[] {
            new Bundle()
        }
    });
    final Intent i = new Intent(NfcAdapter.ACTION_TECH_DISCOVERED);
    i.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
    i.putExtra(NfcAdapter.EXTRA_ID, idm);
    i.putExtra(NfcAdapter.EXTRA_TAG, tag);
    i.putExtra(NfcAdapter.EXTRA_NDEF_MESSAGES, msgs);

    return i;
}
```

build materials

- ・ RC-S620/S
- ・ Arduino向けRC-S620/S制御
ライブラリ(OpenSource)
- ・ RC-S620/Sのフラットケーブル用
ピッチ変換基板
- ・ かわいい箱

ポーリングし、IDmを読み込むコード (a part of Arduino firmware)

```
static RCS620S rcs620s; // 制御ライブラリ提供クラス
int waitCardReleased = 0;

int do_polling()
{
    rcs620s.timeout = COMMAND_TIMEOUT;
    ret = rcs620s.polling();
    if (!ret) {return 0;}
    // idm can be read from rc620s.idm
    return 1;
}
```