柏群RC522读卡器SDK使用说明

在项目settings.gradle文件添加仓库

maven {

url 'https://raw.githubusercontent.com/arkuo0214/BoQunSDK/master/repository'

}

在项目app模块引用库

implementation("com.boqun.nfc:nfc.rc522:1.0.2")

使用示例：

class NfcActivity : AppCompatActivity() {

private var rc522Manager: RC522Manager? = null

private val nfcStateCallback = NfcStateCallback()

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

/\*\*\*\*/

rc522Manager = RC522Manager(

RC522Config(

portPath = "/dev/ttyS0"

)

).apply {

//仅读取，为flase时是写入模式

readOnly = true

registerCallback(nfcStateCallback)

}

}

override fun onDestroy() {

super.onDestroy()

rc522Manager?.unregisterCallback(nfcStateCallback)

}

/\*\*

\* NFC状态回调

\*/

private inner class NfcStateCallback : RC522Manager.Callback {

override fun onCardSelect(tag: String) {

Log.d(TAG, "onCardSelect: " + tag)

}

override fun onCardRead(buffer: ByteArray) {

val data = buffer.bytesToHexString()

Log.d(TAG, "onCardRead: " + data)

}

override fun onCardWrite(data: ByteArray) {

Log.d(TAG, "onCardWrite: " + data)

}

override fun onCardError(reason: Int) {

Log.d(TAG, "onCardError: " + reason)

}

}

companion object {

const val TAG = "NfcActivity"

}

}

## 开放参数设置说明：

授权模式分为A和B两种，默认为A

* authMode

授权密钥，有默认密钥

* authKey

读写块地址，默认使用sector=2和block=1

* blockAddress

读取和写入模式，默认true

* readOnly

写入数据，默认为空

* writeBuffer

示例：

rC522Manager = RC522Manager(

RC522Config(

portPath = "/dev/ttyS0"

)

).apply {

registerCallback(this@NfcActivity)

authMode = AuthMode.AUTH\_A

authKey = AuthKey.default()

blockAddress = BlockAddress(2, 1)

readOnly = true

writeBuffer = byteArrayOf(0xA1.toByte(), 0xA2.toByte(), 0xA2.toByte())

}