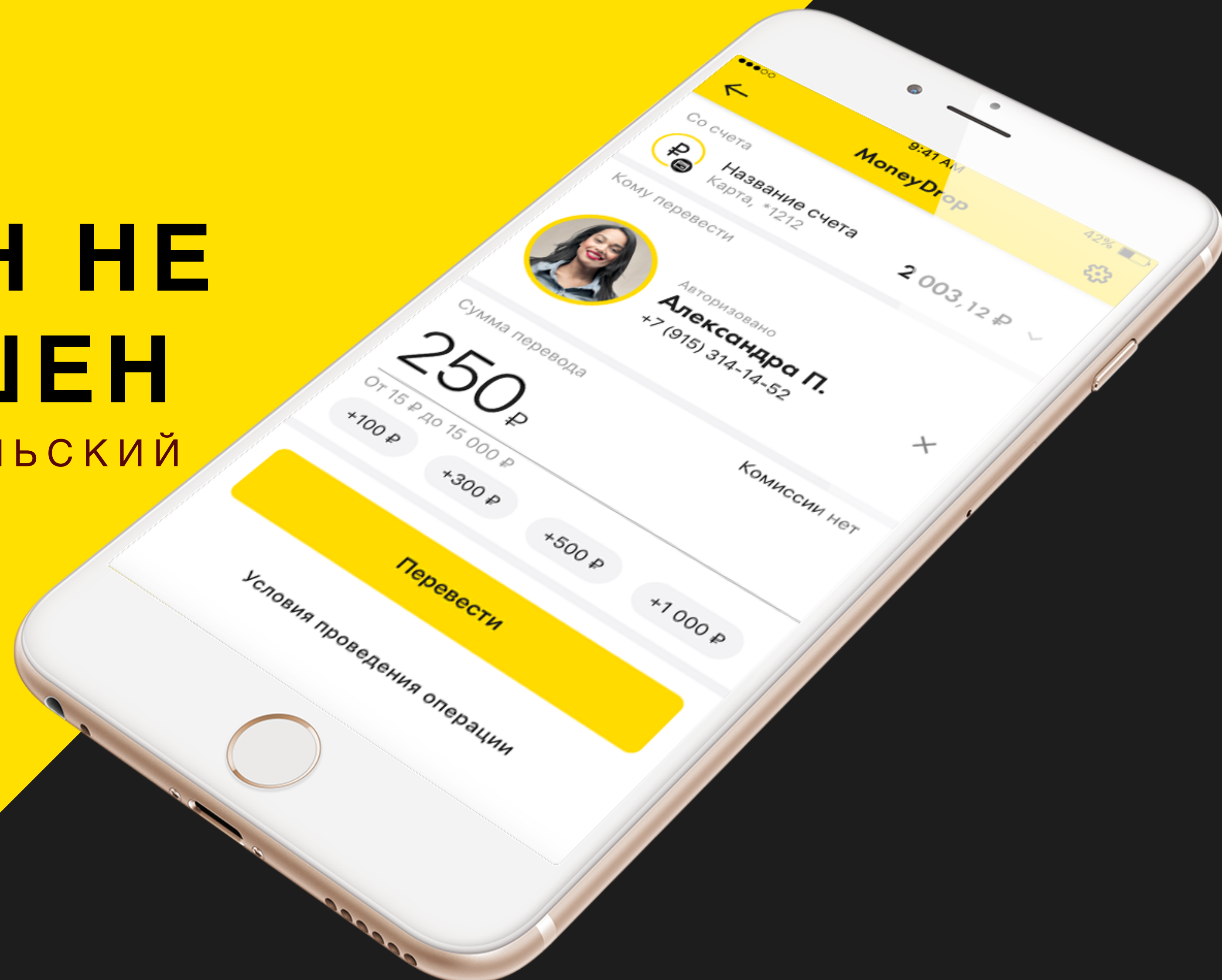


СОРЕВЛУЕТСЯ НЕ ТАК УЖ И СТРАШЕН

ИЛИ КАК УЛУЧШИТЬ ПОЛЬЗОВАТЕЛЬСКИЙ
ОПЫТ БЕЗ ОСОБЫХ УСИЛИЙ.



□ ПЛАН

- ▶ **Что это за технология?**
- ▶ **Где используется?**
- ▶ **Как работает протокол?**
- ▶ **Попробуем сами**
- ▶ **Спустимся чуть глубже**
- ▶ **Подводные камни, куда же без них**

**ПОЧЕМУ
ЛЕ?**



□ ЧТО ЖЕ
ТАКОЕ
ЭТОТ ВТЛЕ

-

○

+



□ ЧТО ЖЕ ТАКОЕ ЭТОТ ВТЛЕ

НЕТ

Меньше времени в эфире

Меньше расход энергии в эфире

Новая архитектура

Ускорено время подключения

—

○

+



Classic Bluetooth

Low Energy

□ **ОТЛИЧИЯ**

□ ОТЛИЧИЯ

ЧАСТОТА

Classic Bluetooth

2.4 ГГц

Low Energy

2.4 ГГц

□ ОТЛИЧИЯ

Classic Bluetooth

Low Energy

ЧАСТОТА

2.4 ГГц

2.4 ГГц

**ЗАДЕРЖКА
ПОДКЛЮЧЕНИЯ**

100мс

15/30 мс

□ ОТЛИЧИЯ

	Classic Bluetooth	Low Energy
ЧАСТОТА	2.4 ГГц	2.4 ГГц
ЗАДЕРЖКА ПОДКЛЮЧЕНИЯ	100мс	15/30 мс
РАССТОЯНИЕ	100м	100м

□ ОТЛИЧИЯ

	Classic Bluetooth	Low Energy
ЧАСТОТА	2.4 ГГц	2.4 ГГц
ЗАДЕРЖКА ПОДКЛЮЧЕНИЯ	100мс	15/30 мс
РАССТОЯНИЕ	100м	100м
ИНТЕРВАЛ ПЕРЕДАЧИ	0.625 мс	3 мс

□ ОТЛИЧИЯ

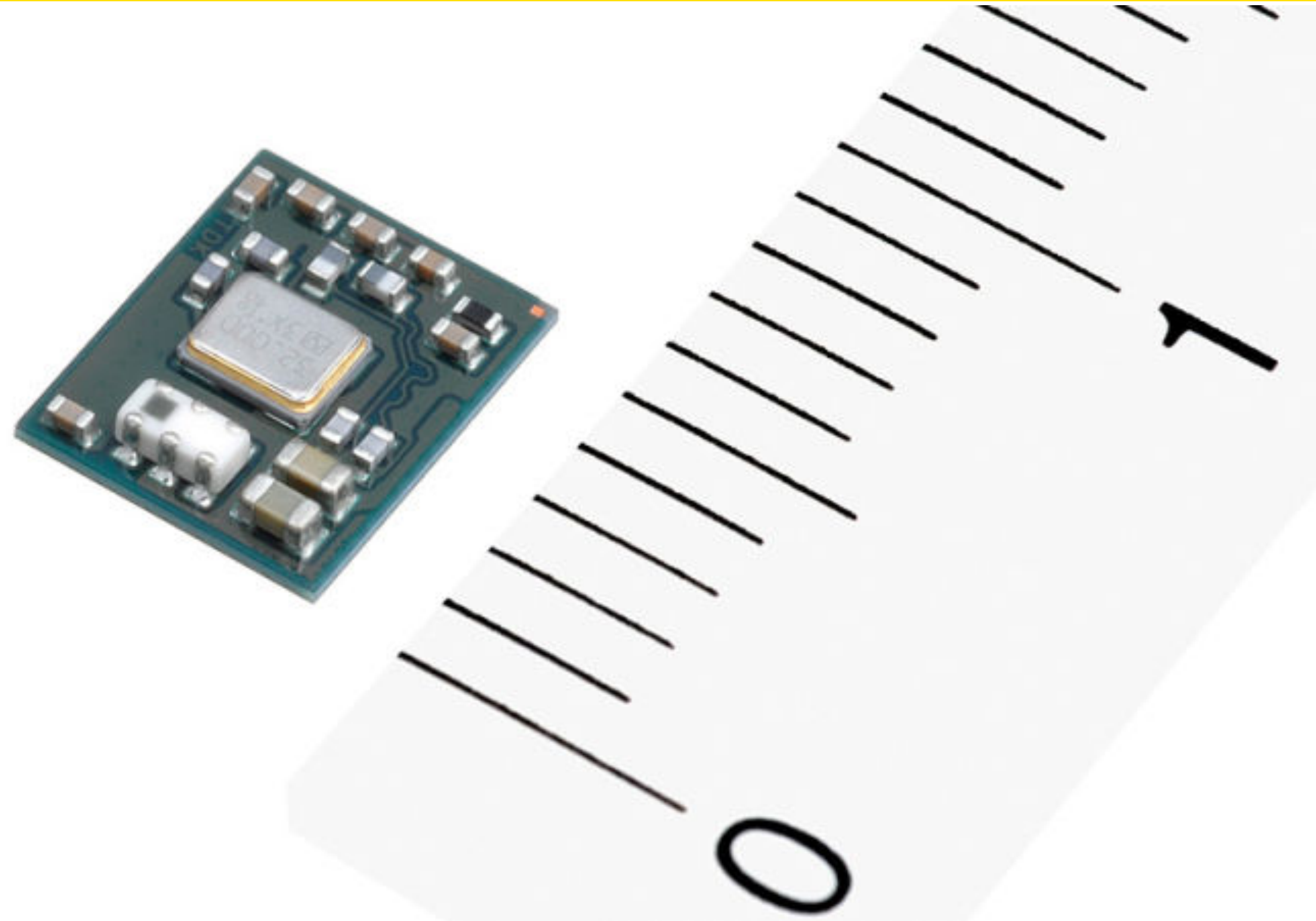
	Classic Bluetooth	Low Energy
ЧАСТОТА	2.4 ГГц	2.4 ГГц
ЗАДЕРЖКА ПОДКЛЮЧЕНИЯ	100мс	15/30 мс
РАССТОЯНИЕ	100м	100м
ИНТЕРВАЛ ПЕРЕДАЧИ	0.625 мс	3 мс
СКОРОСТЬ	24 Mbps	0.27 Mbps

**ГДЕ
ИСПОЛЬЗУЕТСЯ?**

□ ГДЕ ИСПОЛЬЗУЕТСЯ?」



□ ГДЕ ИСПОЛЬЗУЕТСЯ?」



APPLE



APPLE



□ РОЛИ BLUETOOTH LE

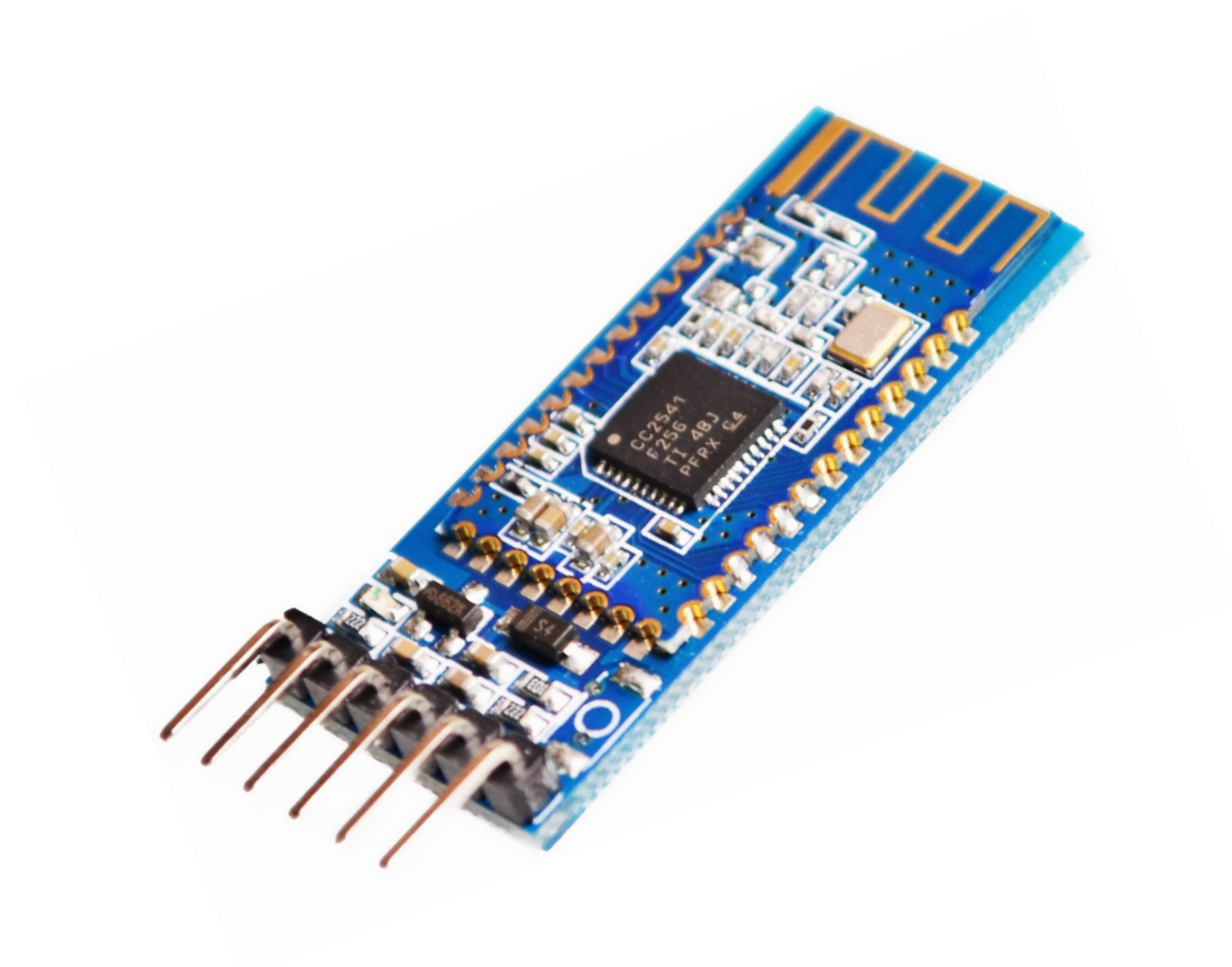
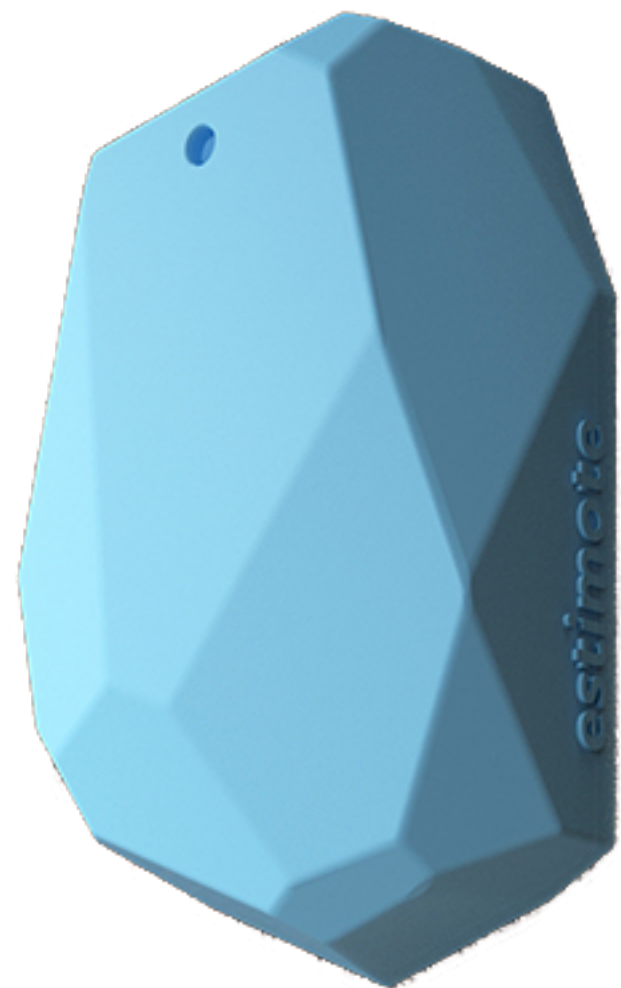
+ ○

Broadcaster

Observer

Central

Peripheral

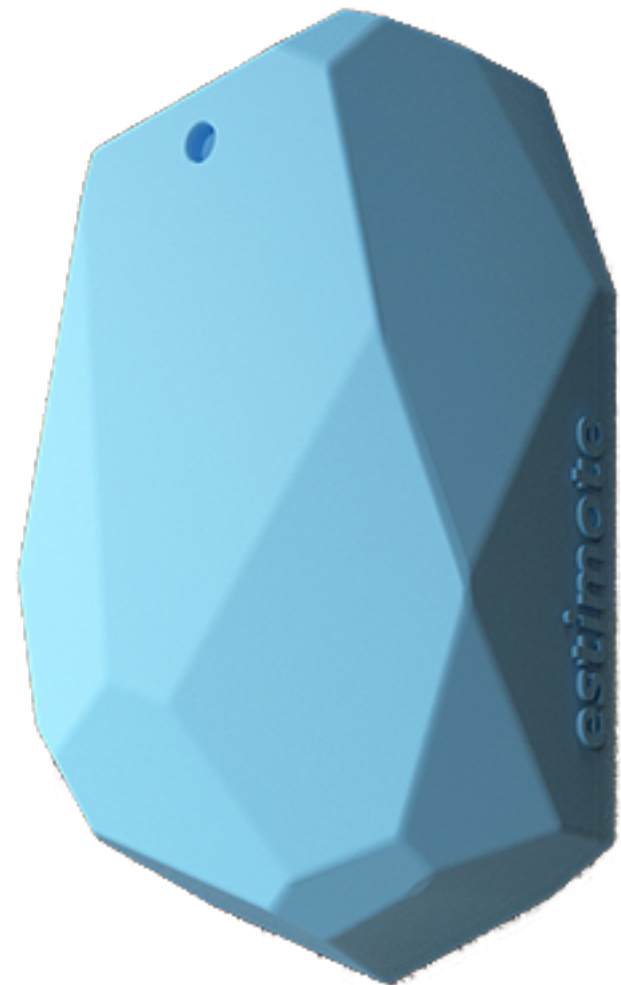


□ РОЛИ BLUETOOTH LE

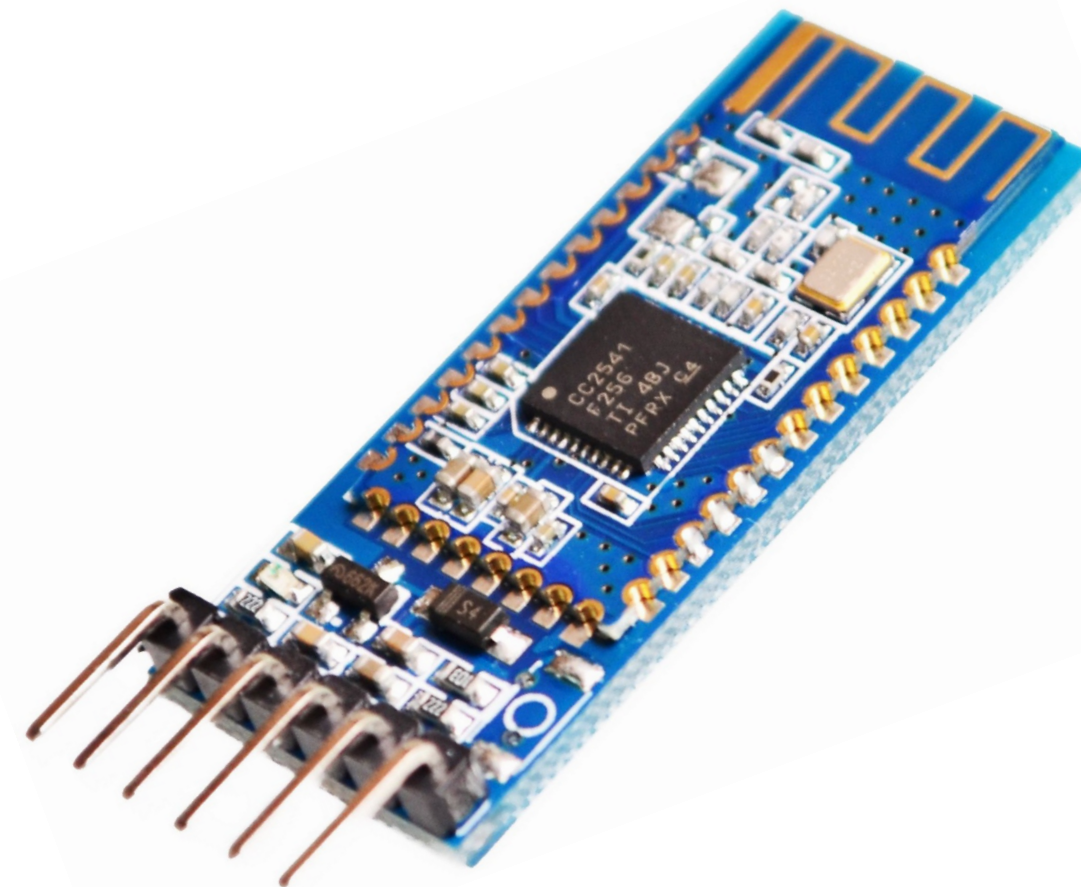
+

○

Broadcaster



Observer



Central



Peripheral



□ **ЧТО НАМ
ДОСТУПНО?**

iOS/macOS

- ▶ Peripheral и Central
- ▶ Фоновый режим
- ▶ Восстановление стејта

watchOS/tvOS

- ▶ watchOS 4+/tvOS 9+
- ▶ Только central роль
- ▶ Максимум 2 подключения
- ▶ Apple watch series 2+/ AppleTv 4+
- ▶ Отключение при переходе в фон

□ ЧТО НАМ
ДОСТУПНО?

iOS/macOS

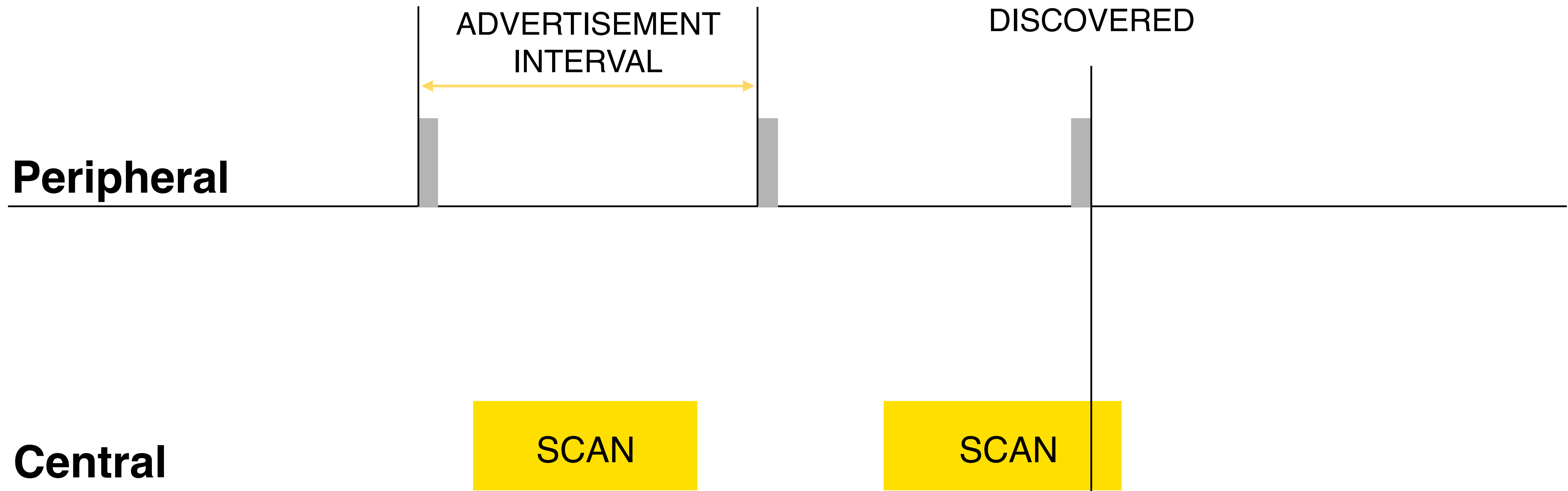
- ▶ Peripheral и Central
- ▶ Фоновый режим
- ▶ Восстановление стејта
- ▶ **Connection interval - 15ms!**

watchOS/tvOS

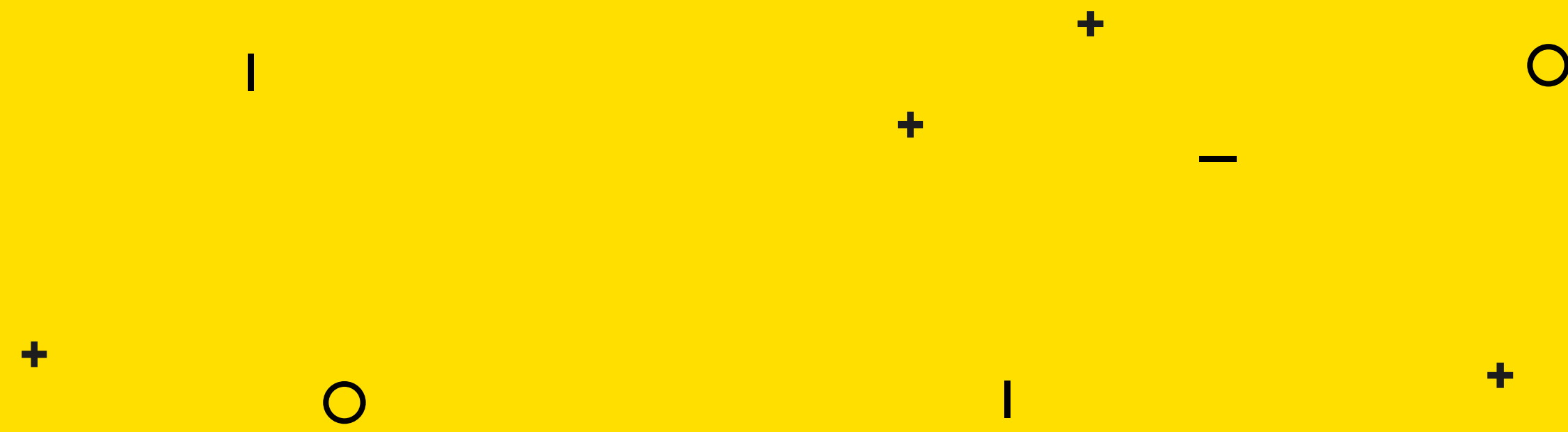
- ▶ watchOS 4+/tvOS 9+
- ▶ Только central роль
- ▶ Максимум 2 подключения
- ▶ Apple watch series 2+/ AppleTv 4+
- ▶ Отключение при переходе в фон
- ▶ **Connection interval - 30ms**

КАК РАБОТАЕТ ПРОТОКОЛ?

DISCOVER



□ CONNECTION



Peripheral

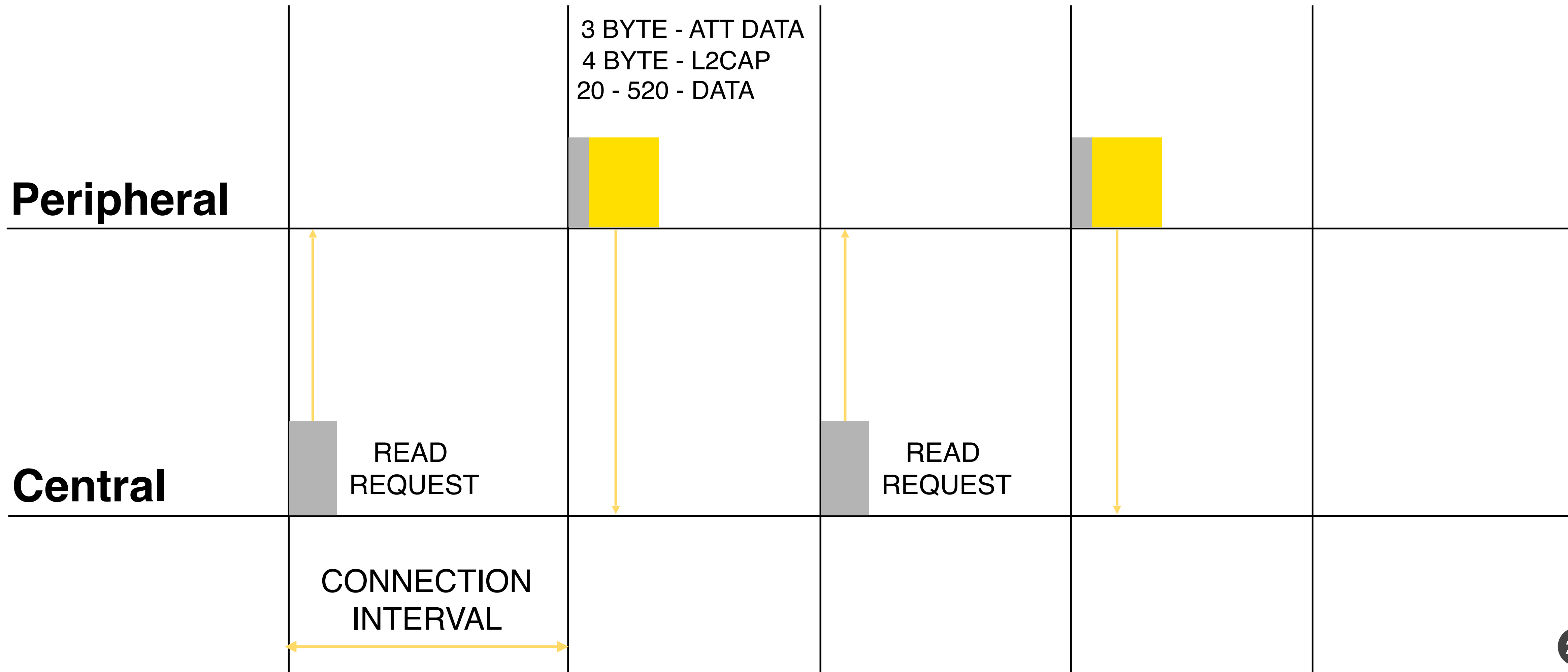
Central

REQUEST

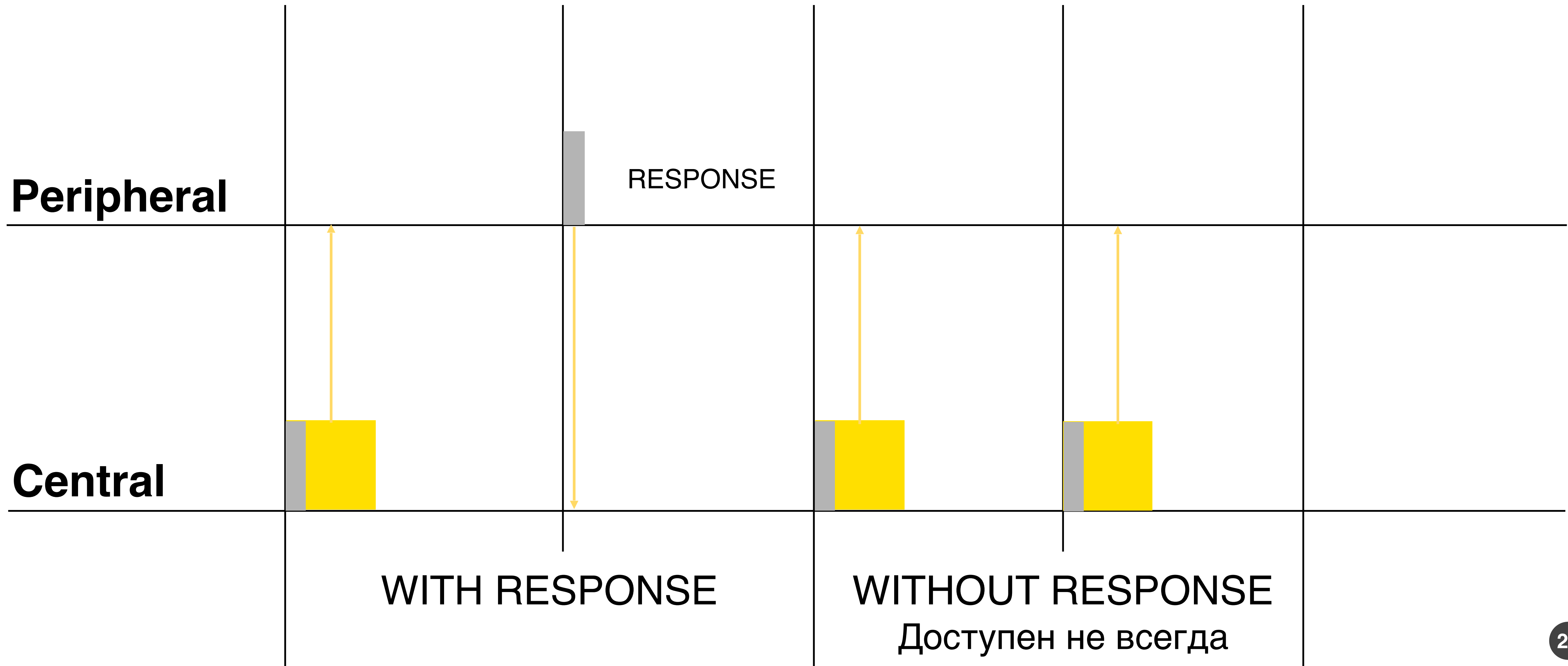
RESPONSE

CONNECTION
INTERVAL

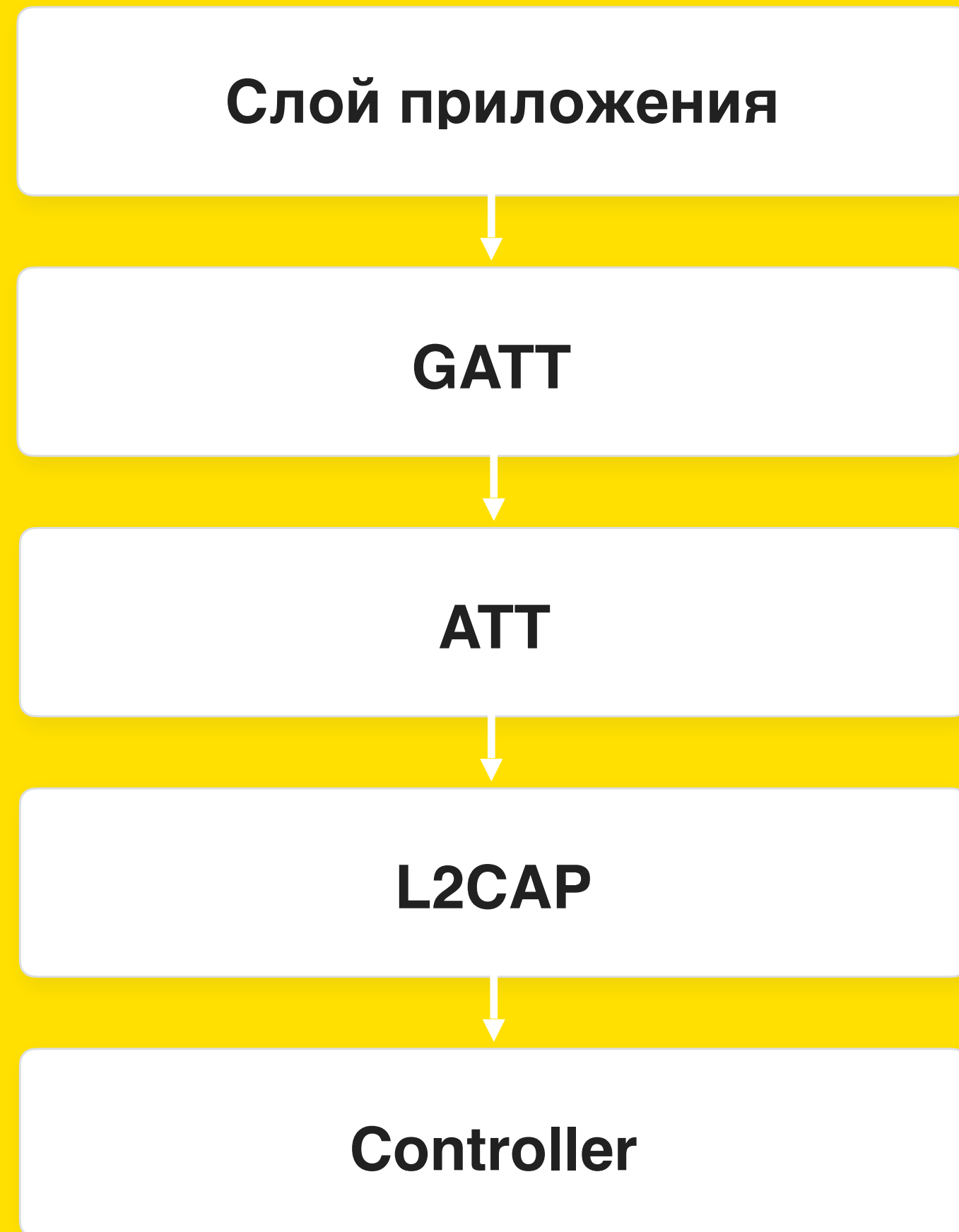
ЧТЕНИЕ



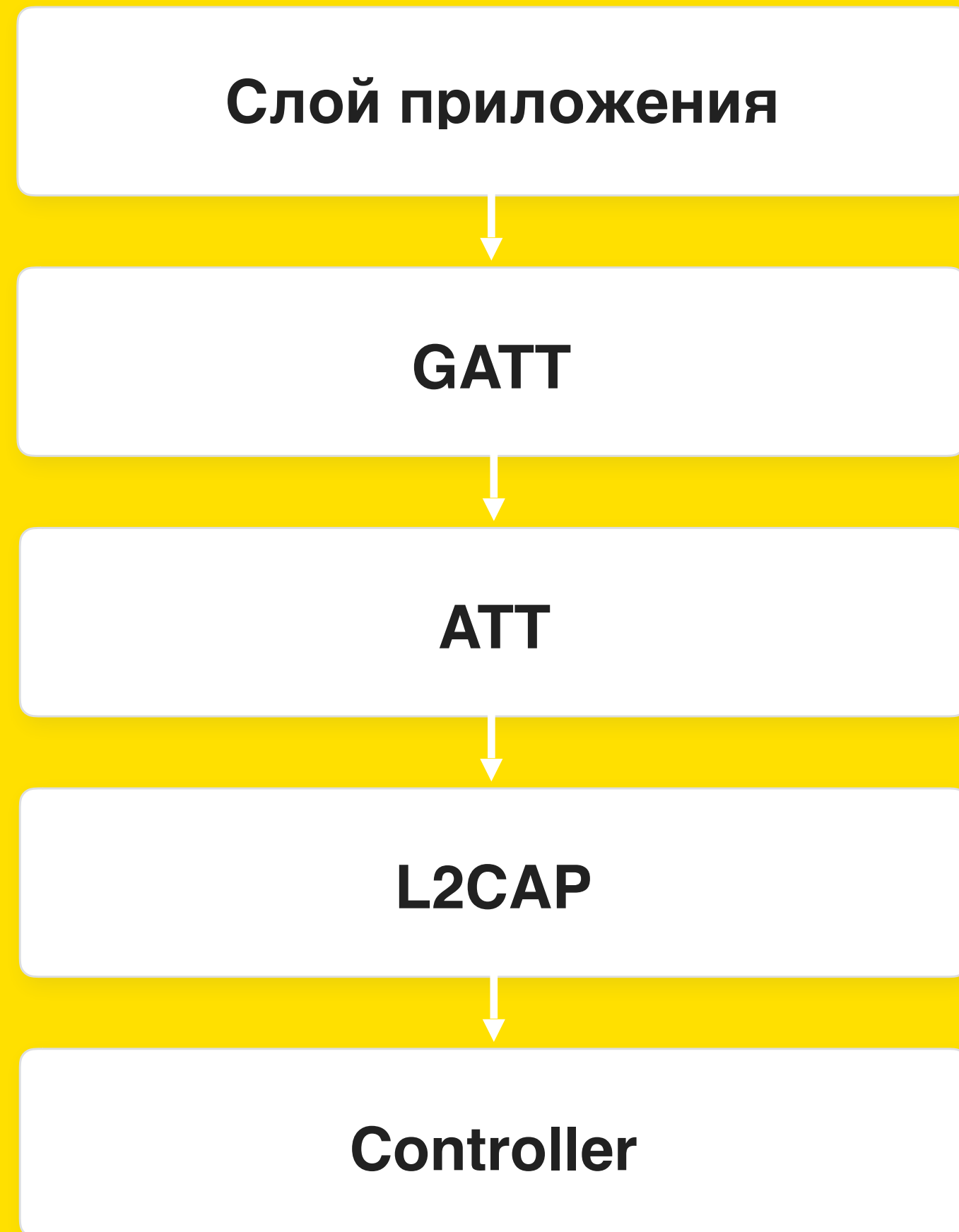
□ ЗАПИСЬ



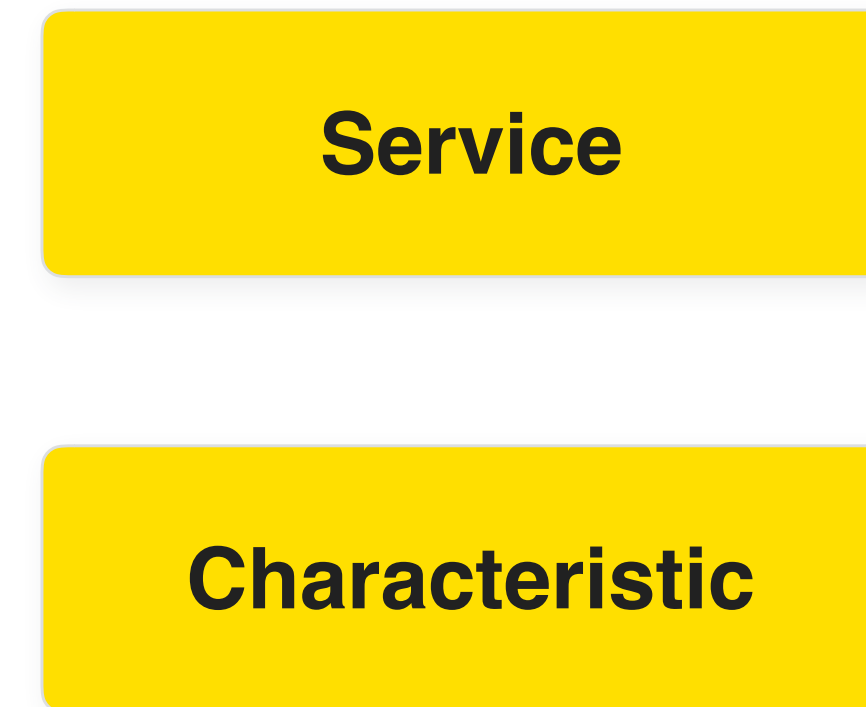
□ ПРОТОКОЛ



□ ПРОТОКОЛ



□ GATT



□ SERVICE

SERVICE

UUID

SERVICE

Characteristic

□ CHARACTERISTIC

□ SERVICE

SERVICE

UUID

SERVICE

Characteristic

□ CHARACTERISTIC

CHARACTERISTIC

UUID

VALUE

DESCRIPTORS

PERMISSIONS

□ УРОВНИ ДОСТУПА

Read

Характеристику можно читать

Write

В характеристику можно записывать

WriteWithoutResponse

В характеристику можно записывать
без отчета о записи

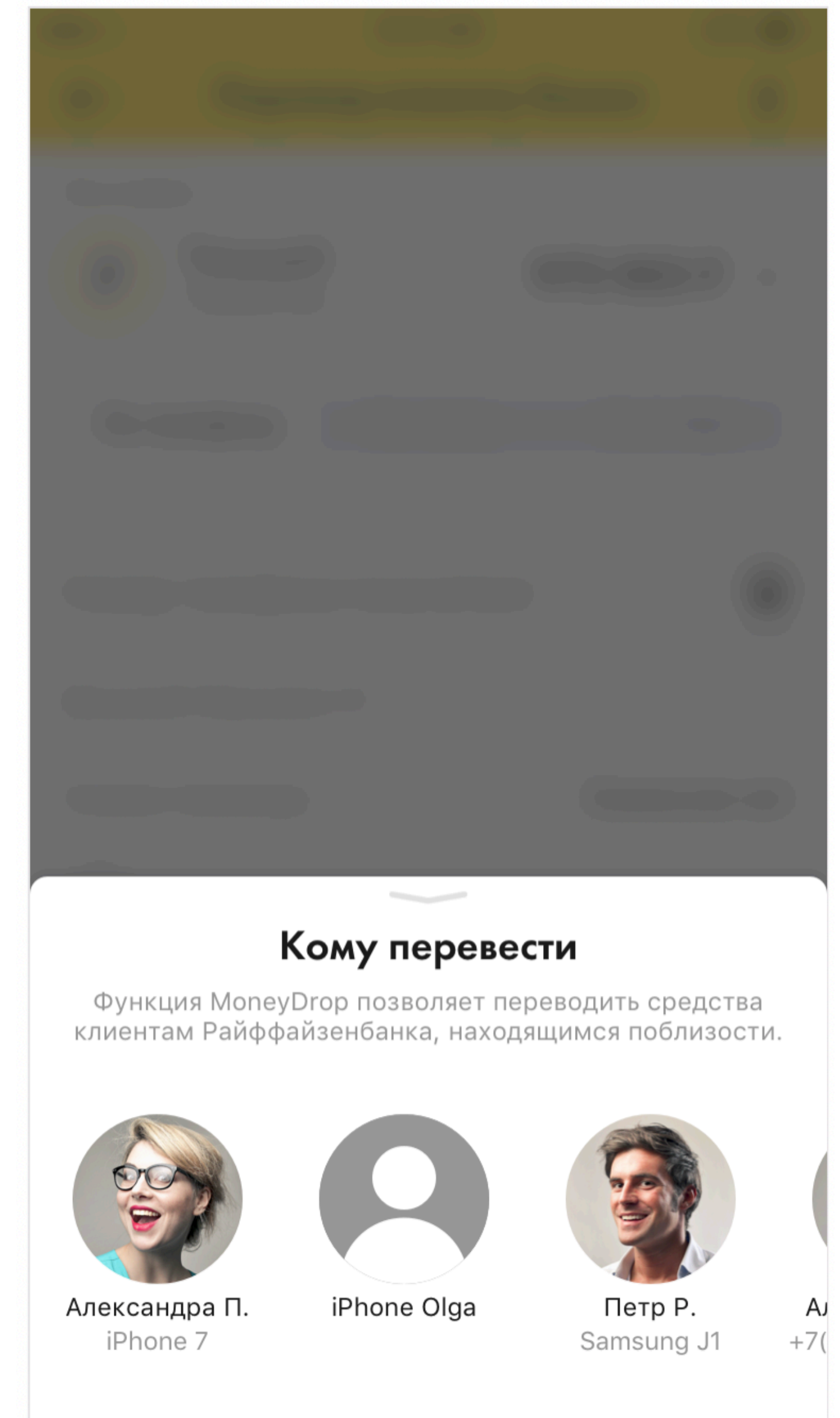
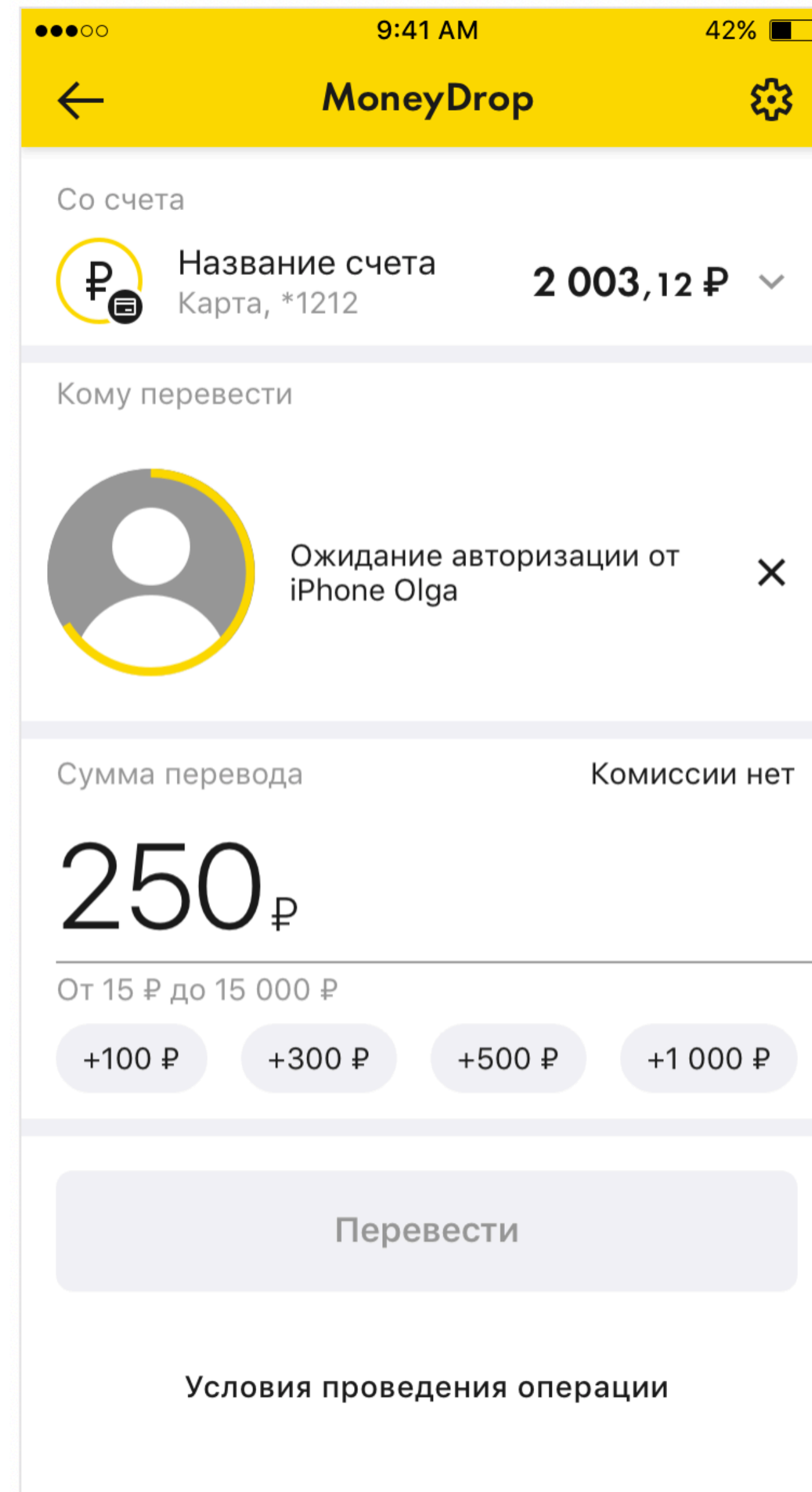
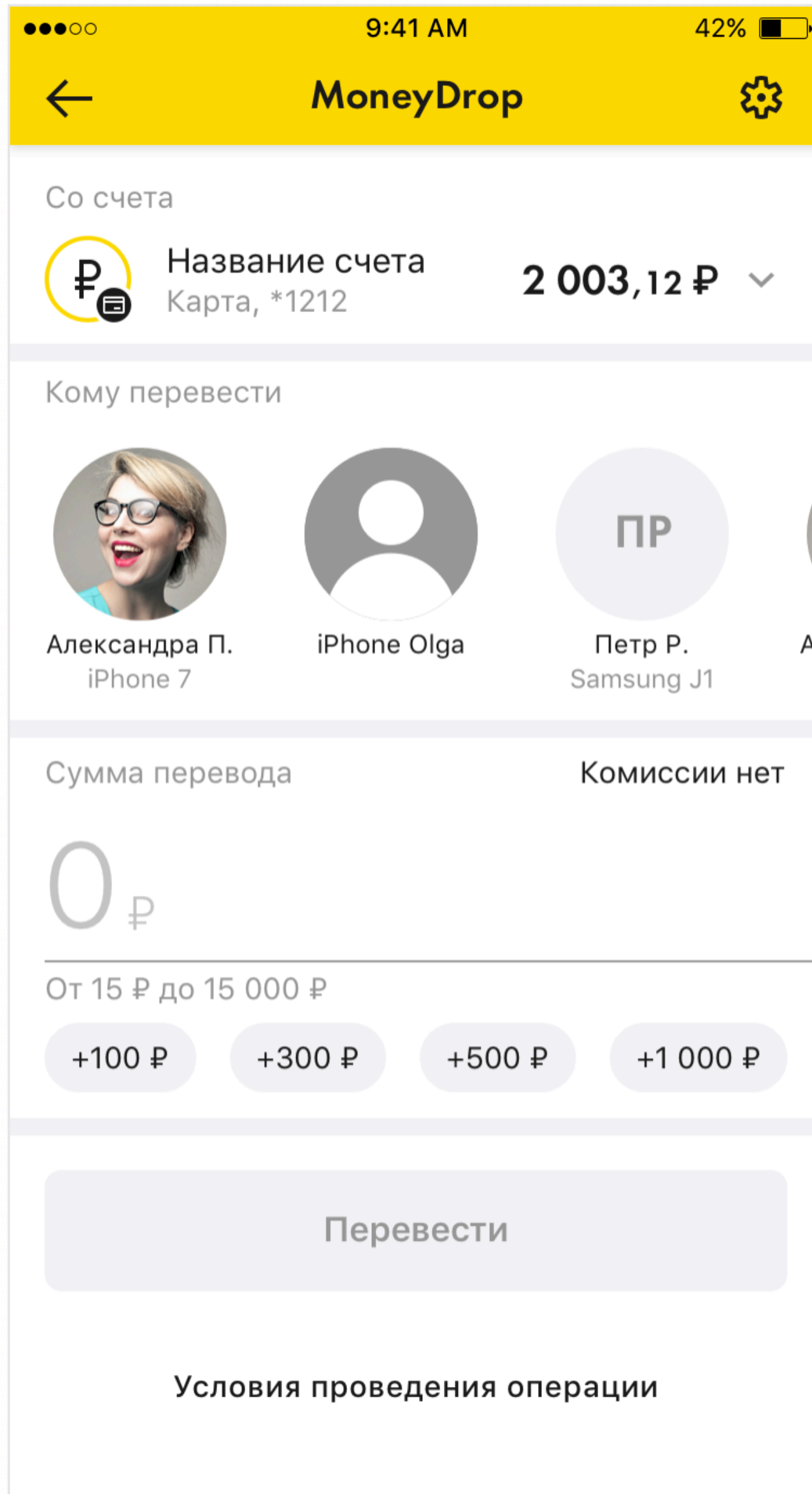
Notify

Можно подписываться об изменениях

Indicate

Можно подписываться об изменениях
без отчета о прочтении

ПОПРОБУЕМ САМИ



□ СХЕМА РАБОТЫ

Отправитель CENTRAL

Поиск

Подключение

Чтение реквизитов получателя

Перевод денег

Отправка сообщения

Получатель PERIPHERAL

Обнаружение

Передача реквизитов

Отображение PUSH

□ КАКИЕ UUID ИСПОЛЬЗОВАТЬ?

```
import CoreBluetooth
```

```
let serviceUUID = CBUUID(string: "e614b814-d9f6-11e8-9f8b-f2801f1b9fd1")
```

```
let userInfoCharacteristicUUID = CBUUID(string: "00d07ce2-d9f7-11e8-9f8b-f2801f1b9fd1")
```

```
let sendInfoCharacteristicUUID = CBUUID(string: "0c74d200-d9f7-11e8-8784-f2801f1b9fd1")
```



.....

□ КАКИЕ UUID ИСПОЛЬЗОВАТЬ?

```
import CoreBluetooth
```

```
let serviceUUID = CBUUID(string: "e614b814-d9f6-11e8-9f8b-f2801f1b9fd1")
```

```
let userInfoCharacteristicUUID = CBUUID(string: "00d07ce2-d9f7-11e8-9f8b-f2801f1b9fd1")
```

```
let sendInfoCharacteristicUUID = CBUUID(string: "0c74d200-d9f7-11e8-8784-f2801f1b9fd1")
```

Любые UUID кроме:

XXXXXXXX-0000-1000-8000-00805F9B34FB

Или купить у Bluetooth SIG за **2500\$**

<https://www.bluetooth.com/specifications/assigned-numbers/16-bit-uuids-for-members>

□ СОЗДАЕМ МЕНЕДЖЕРОВ

```
peripheralManager = CBPeripheralManager(delegate: self, queue: nil)  
centralManager = CBCentralManager(delegate: self, queue: nil)
```

□ СХЕМА РАБОТЫ

Отправитель CENTRAL

Поиск

Подключение

Чтение реквизитов получателя

Перевод денег

Отправка сообщения

Получатель PERIPHERAL

Обнаружение

Передача реквизитов

Отображение PUSH

□ PERIPHERAL - ОБНАРУЖЕНИЕ

//Создаем сервис

```
let service = CBMutableService(type: serviceUUID, primary: true)
```

□ PERIPHERAL - ОБНАРУЖЕНИЕ

```
//Создаем сервис
```

```
let service = CBMutableService(type: serviceUUID, primary: true)
```

```
//Создаем Characteristic для отправки информации о получателе
```

```
let userInfoCharacteristic = CBMutableCharacteristic(type: userInfoCharacteristicUUID,  
properties: .read, value: clientData, permissions: .readable)
```

```
//Создаем Characteristic для информирования что деньги отправлены
```

```
let sendInfoCharacteristic = CBMutableCharacteristic(type: sendInfoCharacteristicUUID,  
properties: .write, value: clientData, permissions: .writeable)
```


□ PERIPHERAL - ОБНАРУЖЕНИЕ

```
//Создаем сервис
```

```
let service = CBMutableService(type: serviceUUID, primary: true)
```

```
//Создаем Characteristic для отправки информации о получателе
```

```
let userInfoCharacteristic = CBMutableCharacteristic(type: userInfoCharacteristicUUID,  
    properties: .read, value: clientData, permissions: .readable)
```

```
//Создаем Characteristic для информирования что деньги отправлены
```

```
let sendInfoCharacteristic = CBMutableCharacteristic(type: sendInfoCharacteristicUUID,  
    properties: .write, value: clientData, permissions: .writeable)
```

```
//Регистрируем их в сервисе
```

```
service.characteristics = [userInfoCharacteristic, sendInfoCharacteristic]
```

```
//Запускаем manager
```

```
peripheralManager?.add(service)
```

```
peripheralManager?.startAdvertising([CBAdvertisementDataServiceUUIDsKey: [serviceUUID]])
```

□ СХЕМА РАБОТЫ

Отправитель CENTRAL

Поиск

Подключение

Чтение реквизитов получателя

Перевод денег

Отправка сообщения

Получатель PERIPHERAL

Обнаружение

Передача реквизитов

Отображение PUSH

□ CENTRAL - ПОИСК & ПОДКЛЮЧЕНИЕ

```
func centralManagerDidUpdateState(_ central: CBCentralManager) {  
    central.scanForPeripherals(withServices: [serviceUUID], options: nil)  
}
```

□ CENTRAL - ПОИСК & ПОДКЛЮЧЕНИЕ

```
func centralManagerDidUpdateState(_ central: CBCentralManager) {  
    central.scanForPeripherals(withServices: [serviceUUID], options: nil)  
}
```

```
func centralManager(_ central: CBCentralManager, didDisconnectPeripheral  
peripheral: CBPeripheral, error: Error?) {  
    availablePeripherals.append(peripheral)  
    central.connect(peripheral, options: nil)  
}
```

□ CENTRAL - ПОИСК & ПОДКЛЮЧЕНИЕ

```
func centralManagerDidUpdateState(_ central: CBCentralManager) {
    central.scanForPeripherals(withServices: [serviceUUID], options: nil)
}
```

```
func centralManager(_ central: CBCentralManager, didDisconnectPeripheral
peripheral: CBPeripheral, error: Error?) {
    availablePeripherals.append(peripheral)
    central.connect(peripheral, options: nil)
}
```

```
func centralManager(_ central: CBCentralManager, didConnect peripheral:
CBPeripheral) {
    peripheral.delegate = self
    peripheral.discoverServices([serviceUUID])
}
```

□ СХЕМА РАБОТЫ

Отправитель CENTRAL

Поиск

Подключение

Чтение реквизитов получателя

Перевод денег

Отправка сообщения

Получатель PERIPHERAL

Обнаружение

Передача реквизитов

Отображение PUSH

□ CENTRAL - ЧТЕНИЕ РЕКВИЗИТОВ

```
func peripheral(_ peripheral: CBPeripheral, didDiscoverServices error: Error?) {
    for service in peripheral.services ?? [] {
        if service.uuid == serviceUUID {
            //Получим все Characteristics для нашего сервиса
            peripheral.discoverCharacteristics(nil, for: service)
        }
    }
}
```

□ CENTRAL - ЧТЕНИЕ РЕКВИЗИТОВ

```
func peripheral(_ peripheral: CBPeripheral, didDiscoverServices error: Error?) {
    for service in peripheral.services ?? [] {
        if service.uuid == serviceUUID {
            //Получим все Characteristics для нашего сервиса
            peripheral.discoverCharacteristics(nil, for: service)
        }
    }
}
```

```
func peripheral(_ peripheral: CBPeripheral, didDiscoverCharacteristicsFor service: CBService, error: Error?) {
    for characteristics in service.characteristics ?? [] {
        if characteristics.uuid == userInfoCharacteristicUUID {
            peripheral.readValue(for: characteristics)
        }
    }
}
```


□ CENTRAL - ЧТЕНИЕ РЕКВИЗИТОВ

```
func peripheral(_ peripheral: CBPeripheral, didDiscoverServices error: Error?) {
    for service in peripheral.services ?? [] {
        if service.uuid == serviceUUID {
            //Получим все Characteristics для нашего сервиса
            peripheral.discoverCharacteristics(nil, for: service)
        }
    }
}

func peripheral(_ peripheral: CBPeripheral, didDiscoverCharacteristicsFor service: CBService, error: Error?) {
    for characteristics in service.characteristics ?? [] {
        if characteristics.uuid == userInfoCharacteristicUUID {
            peripheral.readValue(for: characteristics)
        }
    }
}

func peripheral(_ peripheral: CBPeripheral, didUpdateValueFor characteristic: CBCharacteristic, error: Error?) {
    print("Данные получателя \(characteristic.value!)")
}
```

□ СХЕМА РАБОТЫ

Отправитель CENTRAL

Поиск

Подключение

Чтение реквизитов получателя

Перевод денег

Отправка сообщения

Получатель PERIPHERAL

Обнаружение

Передача реквизитов

Отображение PUSH

□ ОТПРАВКА & ПОЛУЧЕНИЕ

CENTRAL

```
func sendMoney(data: Data, forPeripheral peripheral: CBPeripheral) {  
    let characteristic = savedSendCharacteristic[peripheral]!  
    peripheral.writeValue(data, for: characteristic, type: .withResponse)  
}
```

□ ОТПРАВКА & ПОЛУЧЕНИЕ

CENTRAL

```
func sendMoney(data: Data, forPeripheral peripheral: CBPeripheral) {  
    let characteristic = savedSendCharacteristic[peripheral]!  
    peripheral.writeValue(data, for: characteristic, type: .withResponse)  
}
```

PERIPHERAL

```
func peripheralManager(_ peripheral: CBPeripheralManager, didReceiveWrite requests: [CBATTRequest]) {  
    for request in requests {  
        if request.characteristic.uuid == sendInfoCharacteristicUUID {  
            print("Деньги отправлены : \(request.value!)")  
            peripheralManager?.respond(to: request, withResult: .success)  
        }  
    }  
}
```

□ BACKGROUND

INFO.PLIST

▼ UIBackgroundModes	↕	Array	(1 item)
Item 0	+ -	String	↕ bluetooth-peripheral

□ BACKGROUND

INFO.PLIST

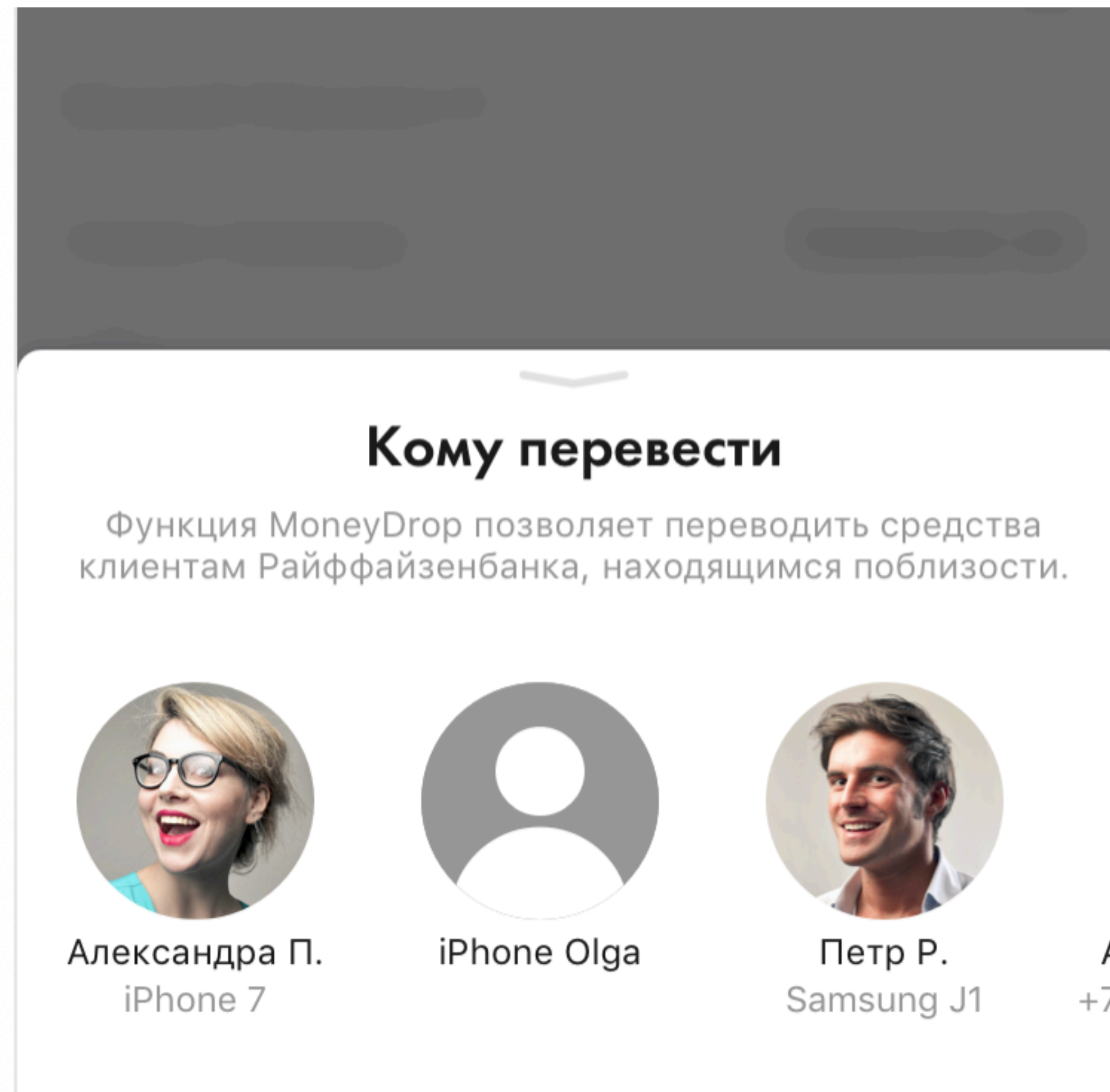
▼ UIBackgroundModes	↕	Array	(1 item)
Item 0	+ -	String	↕ bluetooth-peripheral

MANAGER

```
centralManager = CBCentralManager(delegate: self, queue: nil,
                                   options: [CBPeripheralManagerOptionRestoreIdentifierKey: "restore"])
```

```
func peripheralManager(_ peripheral: CBPeripheralManager, willRestoreState dict: [String : Any]) {
}
```

А где же фото?

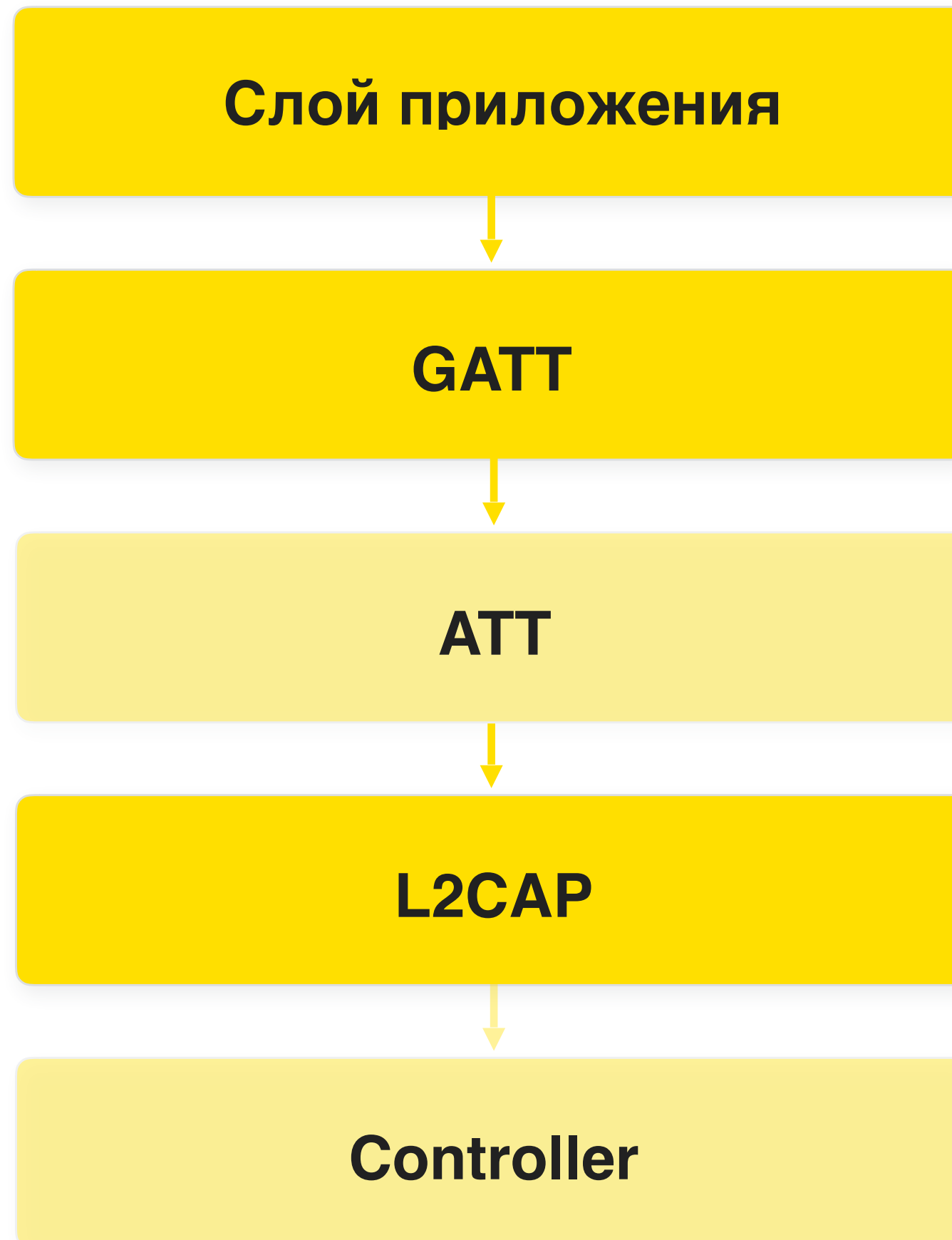


**СПУСТИМСЯ
ГЛУБЖЕ**

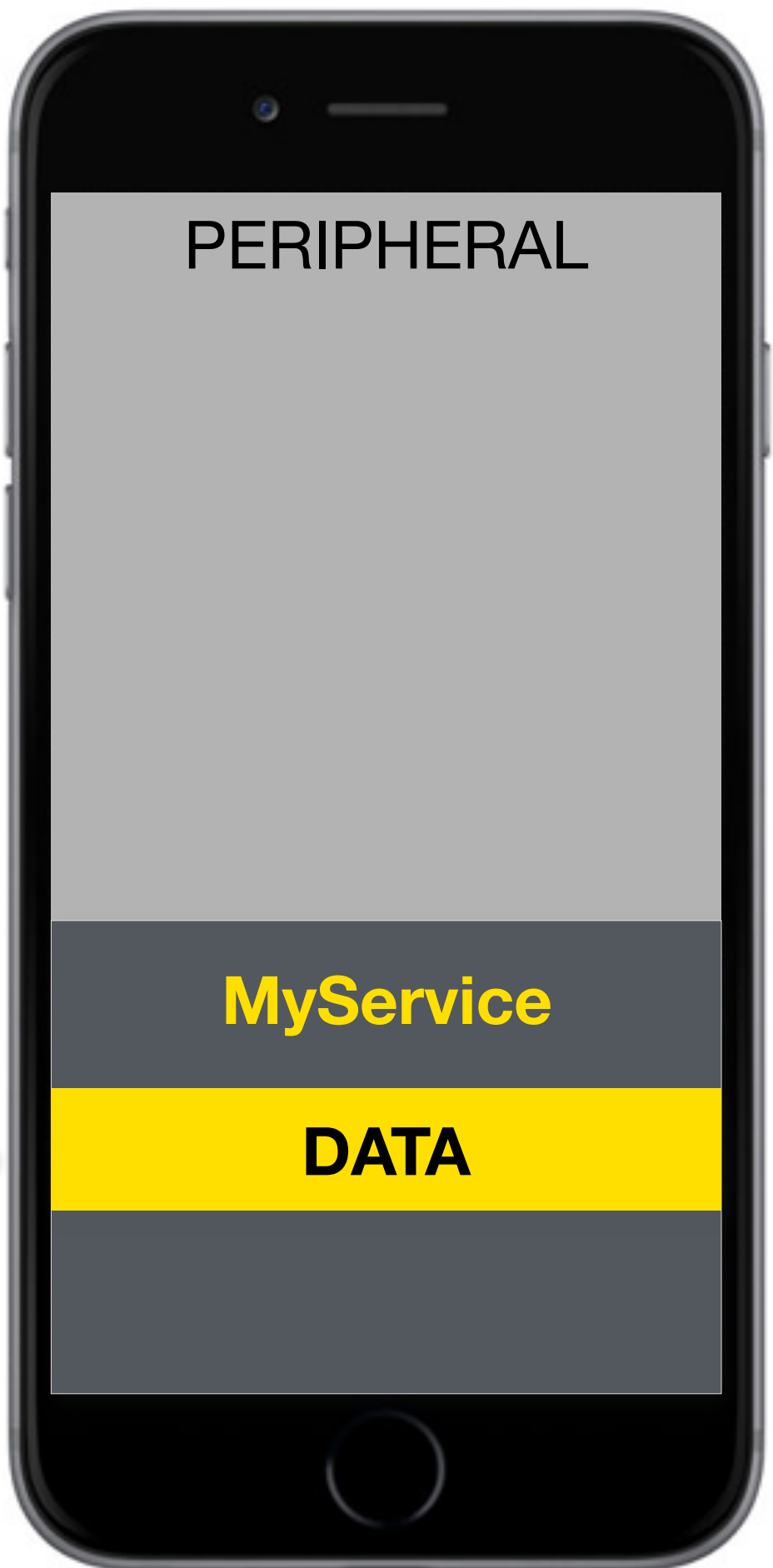
□ СПУСТИМСЯ ГЛУБЖЕ



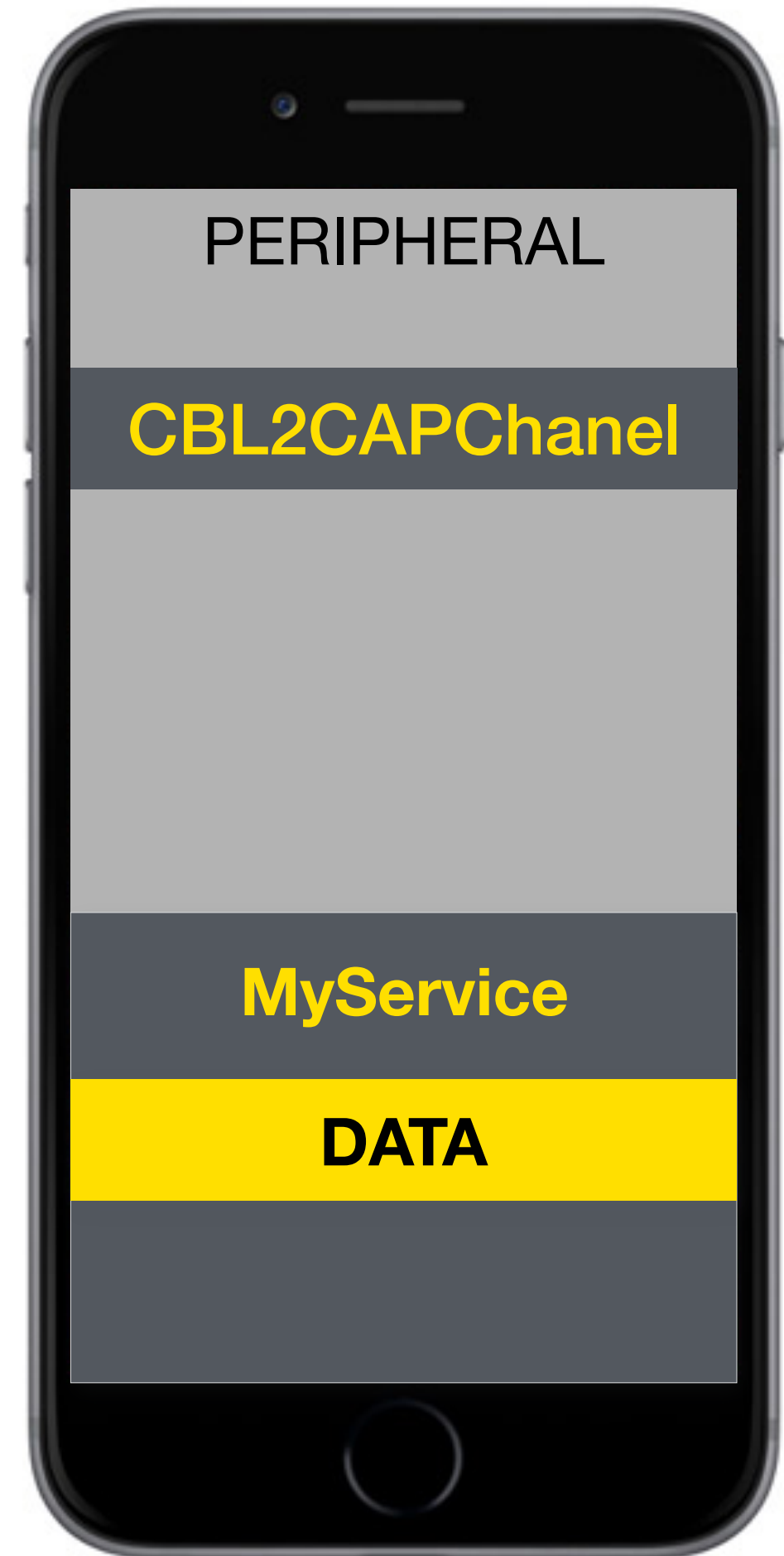
□ СПУСТИМСЯ ГЛУБЖЕ



□ L2CAP



□ L2CAP



□ L2CAP



□ L2CAP



□ L2CAP - PERIPHERAL

Создаем канал

```
peripheralManager?.publishL2CAPChannel(withEncryption: true)
```

□ L2CAP - PERIPHERAL

Создаем канал

```
peripheralManager?.publishL2CAPChannel(withEncryption: true)
```

Отправляем номер канала (порт)

```
func peripheralManager(_ peripheral: CBPeripheralManager, didPublishL2CAPChannel PSM: CBL2CAPPSM, error: Error?) {  
    //отправить channel?.psm to central посредством своей characteristic  
}
```


□ L2CAP - PERIPHERAL

Создаем канал

```
peripheralManager?.publishL2CAPChannel(withEncryption: true)
```

Отправляем номер канала (порт)

```
func peripheralManager(_ peripheral: CBPeripheralManager, didPublishL2CAPChannel PSM: CBL2CAPPSM, error: Error?) {  
    //отправить channel?.psm to central посредством своей characteristic  
}
```

После успешного соединения

```
func peripheralManager(_ peripheral: CBPeripheralManager, didOpen channel: CBL2CAPChannel?, error: Error?) {  
    channel?.outputStream  
    channel?.inputStream  
}
```

□ L2CAP - CENTRAL

Подключаемся

```
availablePeripherals[0].openL2CAPChannel(psm)
```

□ L2CAP - CENTRAL

Подключаемся

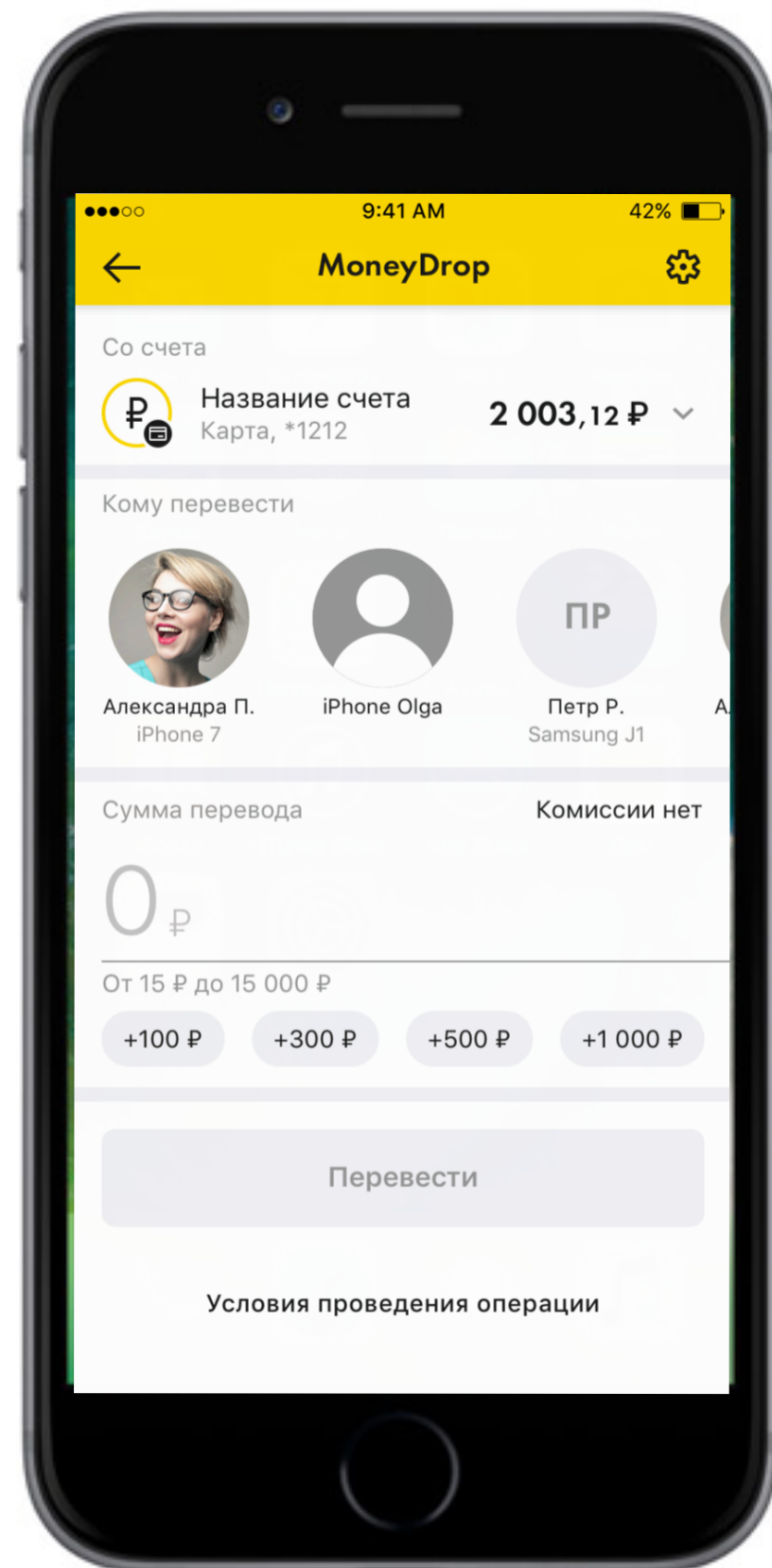
```
availablePeripherals[0].openL2CAPChannel(psm)
```

После успешного соединения

```
func peripheral(_ peripheral: CBPeripheral, didOpen channel: CBL2CAPChannel?, error: Error?) {  
    channel?.outputStream  
    channel?.inputStream  
}
```

ПОДВОДНЫЕ КАМНИ

□ BACKGROUND



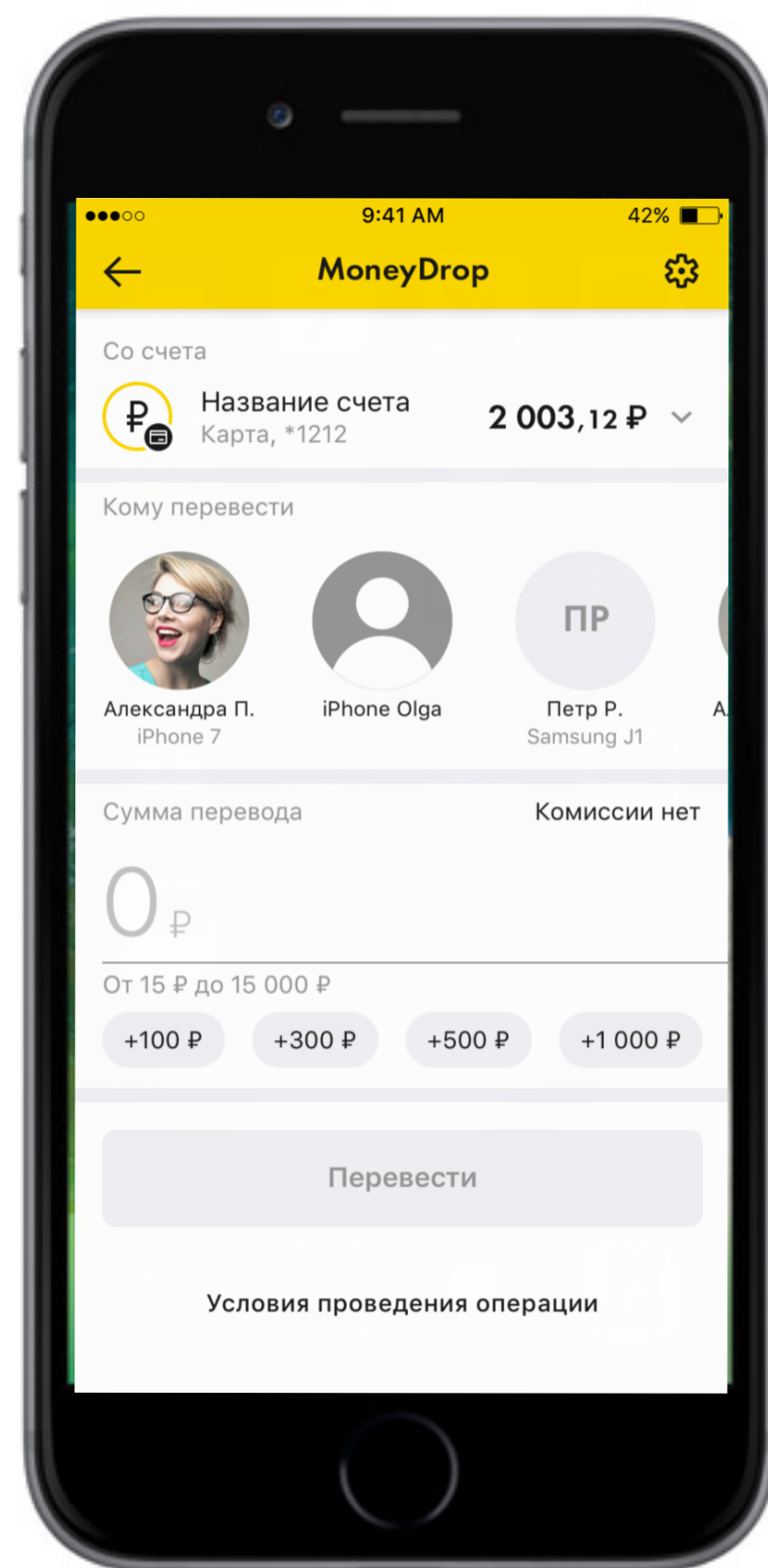
□ BACKGROUND



□ BACKGROUND



□ BACKGROUND



HEADER

DEVICE NAME - 10 BYTES

DATA - 10-18 BYTES

OVERFLOW DATA

□ BACKGROUND



HEADER

DEVICE NAME - 0 BYTES

DATA - 0 BYTES

OVERFLOW DATA

□ UNKNOWN ERROR

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

□ UNKNOWN ERROR

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

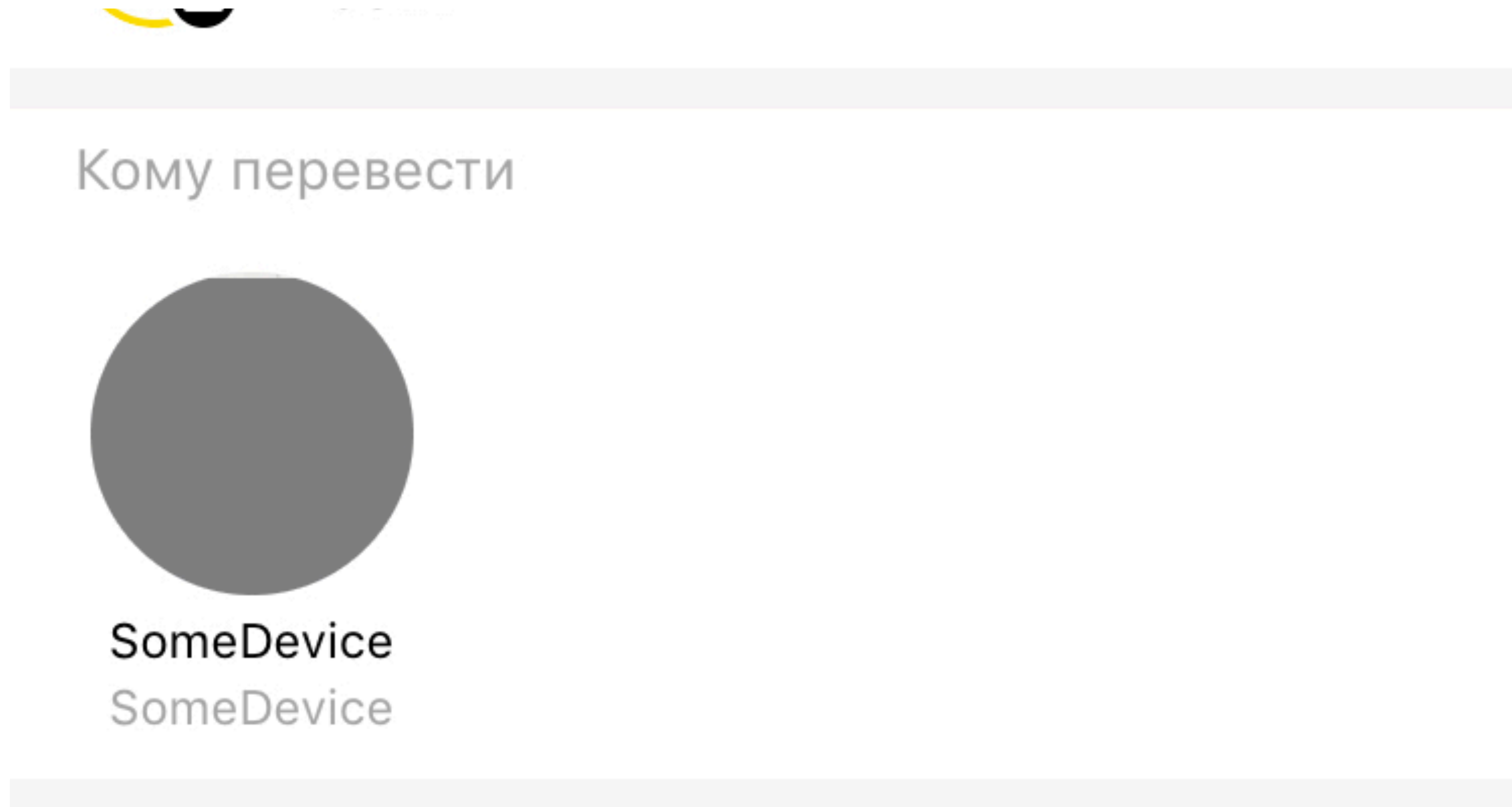
CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

CoreBluetooth[WARNING] Unknown error: 124

.write != .writeWithoutResponse

□ UNKNOWN ERROR



□ UNKNOWN ERROR

CoreBluetooth[WARNING] Unknown error: 722

CoreBluetooth[WARNING] Unknown error: 249

CoreBluetooth[WARNING] Unknown error: 312

□ DEBUG

Apple Bluetooth Explorer

APPLE BLUETOOTH EXPLORER

Bluetooth Explorer Edit **Devices** Tools Utilities Window

- Status ⌘⇧S
- Connection List
- Connection Monitor ⌘⇧C
- Connection Quality Monitor ⌘⇧R
- Get Device Information... ⌘⇧I
- Device Discovery
 - Show Device Discovery ⌘D
 - Clear Device Discovery List
- Low Energy Devices ⌘⇧L
- Near By Devices ⌘⇧B

Low Energy Devices

Start Scanning Disconnect Clear CBCentralManager state: Powered On

Device	RSSI	Addr Type
d0-2b-20-85-48-63	-45	Public
26-cc-a6-8b-ec-e8	-86	Random
5a-80-bf-24-08-e0	-83	Random
c8-d0-83-ba-a6-5e	-86	Public
ec-ad-b8-3f-a7-0c	-46	Public
72-18-90-68-28-46	-93	Random

UUID	Value	Properties
▼ Device Information		
Manufacturer Name String	0x4170706c6520496...	Read
Model Number String	0x5761746368322c34	Read
▼ D0611E78-BBB4-4591-A5F...		
Continuity Characteristic	Value	Write,Notify
▼ Nearby Service		
Nearby Characteristic	Value	Write,Notify

```
Manufacturer Data : <4c001005 0118d1d7 54>
kCBAAdvDataChannel : 37
kCBAAdvDataIsConnectable : 0
kCBAAdvDataAppleMfgData : {
  kCBScanOptionAppleFilterPuckType = 16;
}
```

Total near by devices : 6

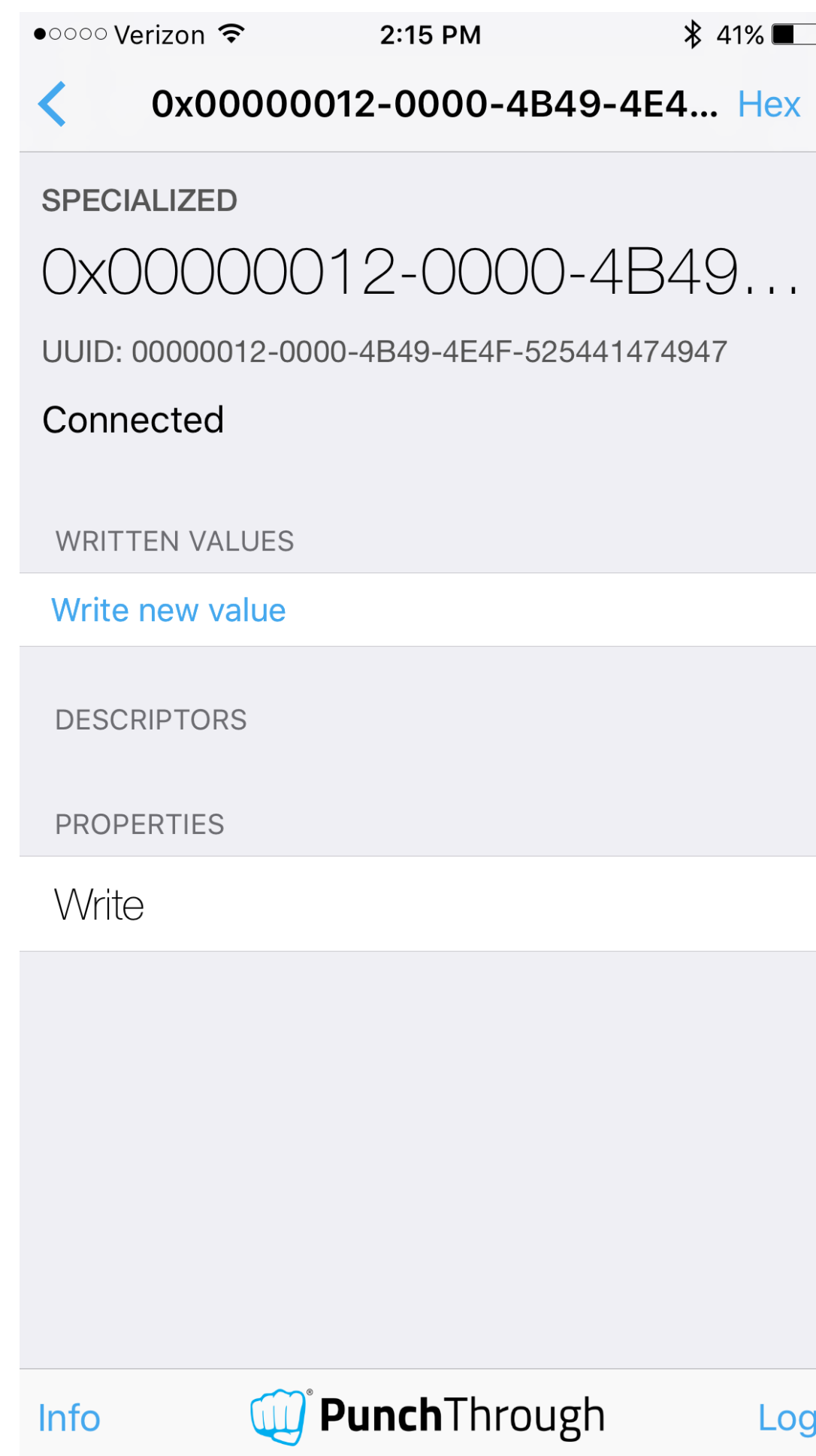
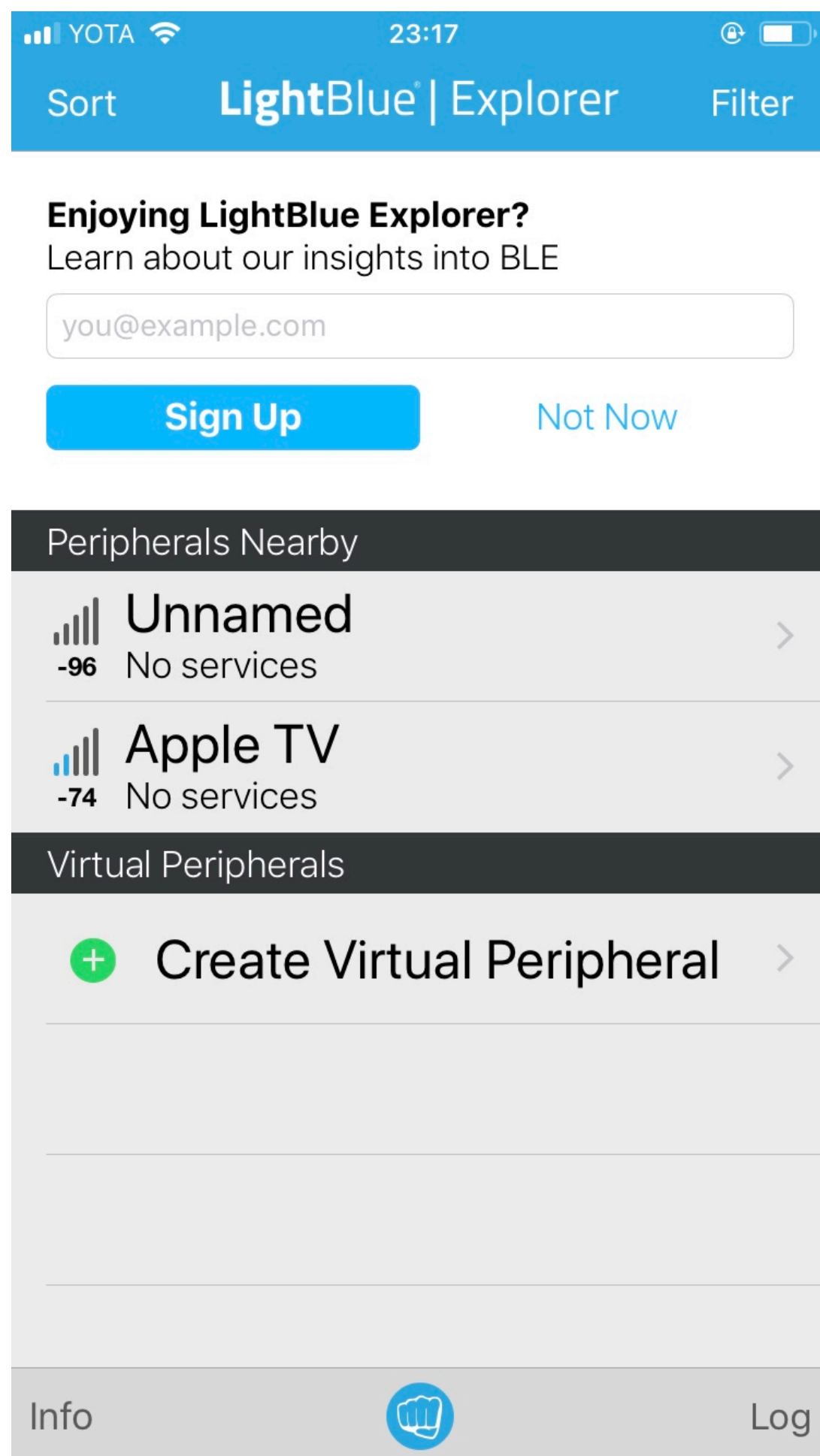
UTF8 Write Read Reg Notify

□ **DEBUG**

Apple Bluetooth Explorer

LightBlue Explorer

LIGHTBLUE EXPLORER



□ DEBUG

Apple Bluetooth Explorer

LightBlue Explorer

WireShark

WIRESHARK

File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help

Filter: **btle** Expression... Clear Apply Save

No.	Time	Source	Destination	Protocol	Length	Info
1338	245.312104	Slave	Master	LE LL	60	ADV_IND
1339	245.315097	Slave	Master	LE LL	60	ADV_IND

Frame 1338: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0

- Nordic BLE sniffer meta
- Bluetooth Low Energy Link Layer**
 - Access Address: 0x8e89bed6
 - Packet Header: 0x2240 (PDU Type: ADV_IND, TXAdd=false, RxAdd=false)
 - Advertising Address: e4:c6:c7:31:95:11 (e4:c6:c7:31:95:11)
 - Advertising Data**
 - Appearance: Generic Tag
 - Length: 3
 - Type: Appearance (0x19)
 - Appearance: Generic Tag (0x0200)
 - Flags
 - Length: 2
 - Type: Flags (0x01)
 - 000. = Reserved: 0x00
 - ...0 ... = Simultaneous LE and BR/EDR to Same Device Capable (Host): false (0x00)
 - ... 0... = Simultaneous LE and BR/EDR to Same Device Capable (Controller): false (0x00)
 -1.. = BR/EDR Not Supported: true (0x01)
 -1. = LE General Discoverable Mode: true (0x01)
 -0 = LE Limited Discoverable Mode: false (0x00)
 - Tx Power Level
 - Length: 2
 - Type: Tx Power Level (0x0a)
 - Power Level (dBm): 0
 - 128-bit Service Class UUIDs
 - Length: 17
 - Type: 128-bit service class UUIDs (0x07)
 - Custom UUID: 9ecadc240ee5a9e093f3a3b50100406e
 - CRC: 0xdf2f9f

```
0000 11 06 35 01 7b 75 06 0a 01 25 2b 00 00 5e 53 08 ..5.{u.. .%+..^S.
0010 00 d6 be 89 8e 40 22 11 95 31 c7 c6 e4 03 19 00 .....@". .1...
0020 02 02 01 06 02 0a 00 11 07 9e ca dc 24 0e e5 a9 .....$..
0030 e0 93 f3 a3 b5 01 00 40 6e fb f4 f9 .....@ n...
```

□ DEBUG

Apple Bluetooth Explorer

LightBlue Explorer

WireShark

□ **DEBUG**

Apple Bluetooth Explorer

LightBlue Explorer

WireShark

PacketLogger

PACKETLOGGER

Untitled [Live]
Unknown... ▾
Unfiltere... ▾
Off ▾
Q Search

Stop Clear
Priorities Throughput
Device Filter
ACL Filter
Syslog Level Filter
Find Filter

Decode Packets
50495 total (8 Err / 364 HCI / 0 ACL / 0 SCO / 124 Misc)
Δ 0.524 s

Time	Type	Handle	Addr	Decoded Packet
Dec 04 18:15:50.681	HCI Command		...e Watch - EVGENY	▶ [2011] LE Add Device To White List - 0 - EC:AD:B8:3F:A7:0C
Dec 04 18:15:50.805	HCI Event	0x0043	...e Watch - EVGENY	▶ LE Meta Event - LE Connection Complete - Master - Public - EC:AD:B8:3F:A7:0C - Conn Interval: 15 ms
Dec 04 18:15:50.818	HCI Command		...e Watch - EVGENY	▶ [2012] LE Remove Device From White List - 0 - EC:AD:B8:3F:A7:0C
Dec 04 18:15:50.826	HCI Command	0x0043	...e Watch - EVGENY	▶ [2019] LE Start Encryption - Connection Handle: 0x0043
Dec 04 18:15:51.056	HCI Event	0x0043	...e Watch - EVGENY	▶ Encryption Change Complete - Encryption Enabled
Dec 04 18:15:51.057	ATT Send	0x0043	...e Watch - EVGENY	▶ Exchange MTU Request - MTU: 515
Dec 04 18:15:51.086	ATT Receive	0x0043	...e Watch - EVGENY	▶ Handle Value Indication
Dec 04 18:15:51.095	ATT Send	0x0043	...e Watch - EVGENY	▶ Handle Value Confirmation
Dec 04 18:15:51.146	ATT Receive	0x0043	...e Watch - EVGENY	▶ Exchange MTU Response - MTU: 515
Dec 04 18:15:51.146	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x0001 - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.148	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x0001 - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.148	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Group Type Response
Dec 04 18:15:51.191	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Group Type Response
Dec 04 18:15:51.191	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x000F - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.206	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x0015 - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.206	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Group Type Response
Dec 04 18:15:51.222	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Group Type Response
Dec 04 18:15:51.222	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x0019 - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.236	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Group Type Request - Start Handle: 0x0028 - End Handle: 0xFFFF - UUID: GATT Primary Service
Dec 04 18:15:51.237	ATT Send	0x0043	...e Watch - EVGENY	▶ Error Response - Attribute Handle: 0x0028 - Error Code: 0x0A
Dec 04 18:15:51.251	ATT Receive	0x0043	...e Watch - EVGENY	▶ Error Response - Attribute Handle: 0x0019 - Error Code: 0x0A
Dec 04 18:15:51.266	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Type Request - Start Handle: 0x0006 - End Handle: 0x0009 - UUID: GATT Characteristic Declaration
Dec 04 18:15:51.277	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Type Request - Start Handle: 0x0001 - End Handle: 0x0005 - UUID: GATT Characteristic Declaration
Dec 04 18:15:51.278	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Type Response
Dec 04 18:15:51.311	ATT Receive	0x0043	...e Watch - EVGENY	▶ Read By Type Response
Dec 04 18:15:51.311	ATT Send	0x0043	...e Watch - EVGENY	▶ Read By Type Request - Start Handle: 0x0006 - End Handle: 0x0009 - UUID: GATT Characteristic Declaration
Dec 04 18:15:51.311	ATT Receive	0x0043	...e Watch - EVGENY	▶ Find Information Request - Handle: 0x0009 - Service Changed

Packet Types

- ACLDecoder:Error 10
- ATTDecoder 5
- ATTDecoder:Error 10
- Config 3
- Error 10**
- HCI Command:Change Connection Packet Type 6
- HCI Command:Change Local Name 5
- HCI Command:Disconnect 9
- HCI Command:LE Add Device To White List 9
- HCI Command:LE Clear White List 5
- HCI Command:LE Connection Update 5
- HCI Command:LE Create Connection 7
- HCI Command:LE Create Connection Cancel 5
- HCI Command:LE Long Term Key Request Reply 5
- HCI Command:LE Read Buffer Size 5
- HCI Command:LE Read Local Supported Features 5
- HCI Command:LE Read Suggested Default Data L... 5
- HCI Command:LE Remove Device From White List 7
- HCI Command:LE Set Event Mask 6
- HCI Command:LE Set Random Address 6
- HCI Command:LE Set Scan Enable 6
- HCI Command:LE Set Scan Parameters 6
- HCI Command:LE Start Encryption 7
- HCI Command:LE Write Suggested Default Data L... 5
- HCI Command:Read Buffer Size 5
- HCI Command:Read Class of Device 5
- HCI Command:Read Device Address 5
- HCI Command:Read Local Extended Features 5

Priority

□ DEBUG

ATT Send	0x0042	iPhone 8	▶ Write Response
L2CAP Receive	0x0042	iPhone 8	▶ LE Credit Based Connection Request - (PSM: 0x00C1)
L2CAP Send	0x0042	iPhone 8	▶ LE Credit Based Connection Response
L2CAP Send	0x0042	iPhone 8	▶ LE Flow Control Credit
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x04E3 (1251) [E3 8A 8A 8A]
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x0002 (02) [A2 8A]
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x04E3 (1251) [E3 8A 8A 8A]
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x0002 (02) [E5 4D]
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x04E3 (1251) [E3 8A 8A 8A]
ATT Receive	0x0042	iPhone 8	▶ Write Response
ATT Receive	0x0042	iPhone 8	▶ Write Request - Handle:0x0032 - Value: 0200
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x0002 (02) [9B FE]
L2CAP Send	0x0042	iPhone 8	▶ Channel ID: 0x004C Length: 0x04E3 (1251) [E3 8A 8A 8A]
L2CAP Receive	0x0042	iPhone 8	▶ Disconnection Request - (PSM: 0x00C1)

□ ПОЛЕЗНЫЕ ССЫЛКИ

WWDC/CoreBluetooth

<https://developer.apple.com/videos/play/wwdc2017/712/>

WWDC 2012 Session 703 Core Bluetooth 101

WWDC 2012 Session 705 Advanced Core Bluetooth

Bluetooth

<https://www.bluetooth.com/specifications/gatt>

Getting Started with Bluetooth Low Energy O'Reilly

YouTube

Arrow Electronics -> Bluetooth Low Energy Series

Q/A

Антропов Евгений

e@antropov.it

<http://t.me/eantropov>