Bluetooth Low Energy for starters

Bogdan Vlad

Staff Engineer @ Fitbit

Bluetooth Low Energy for starters

1. Introduction to Bluetooth Low Energy
2. CoreBluetooth.framework
3. Recent BLE improvements in iOS
4. Bluetooth profiles exposed by iOS
5. Useful resources

Introduction to Bluetooth Low Energy

- Bluetooth 4.0 specification in 2010
- Low power consumption
- Small size and low cost
- Targeted to support IoT Applications

Introduction to Bluetooth Low Energy

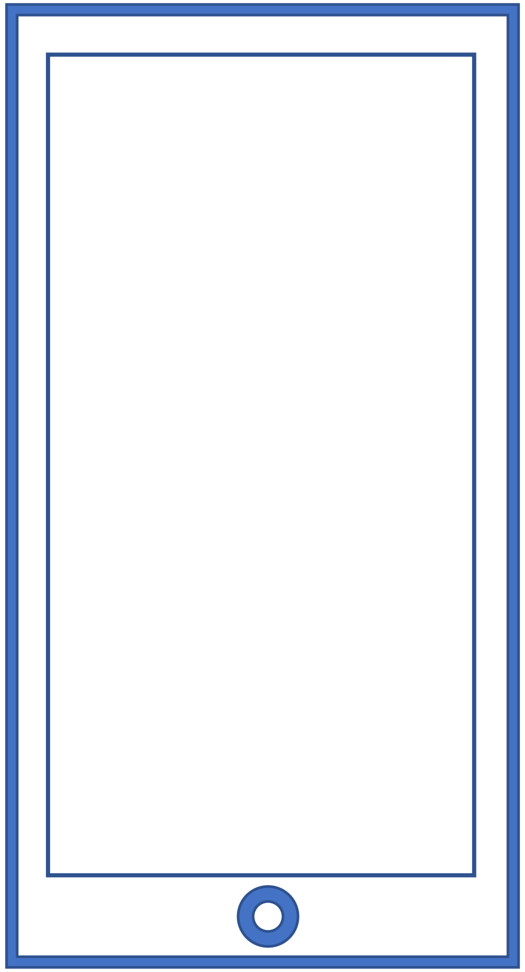
- Bluetooth 4.0 specification in 2010
- Low power consumption
- Small size and low cost
- Targeted to support IoT Applications

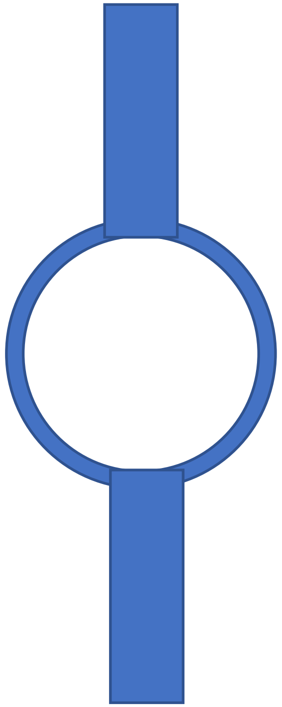
Fun facts:

- First phone with BLE: iPhone 4S (October 2011)
- The first two computers with BLE: MacMini and MacBook Air (July 2011)
- Apple and Nordic Semiconductor join the SIG Board of Directors (2011)

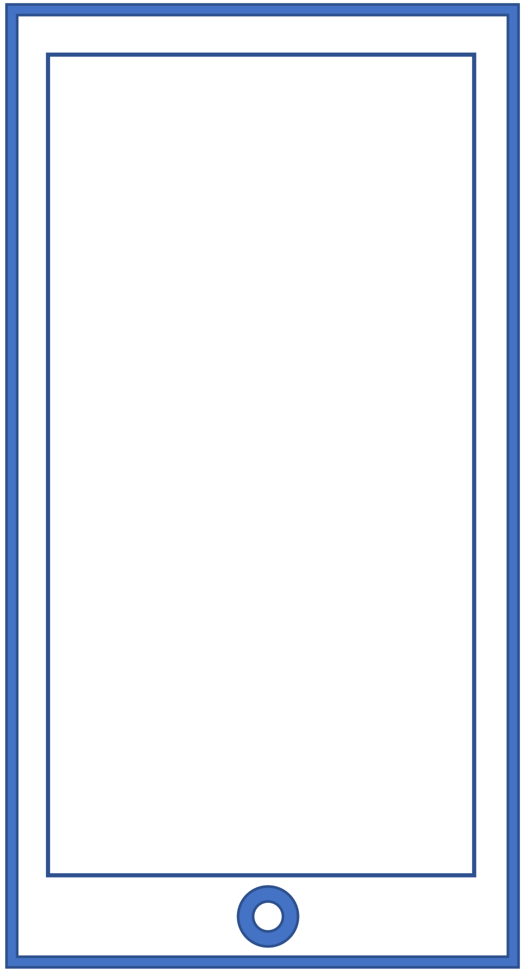
Introduction to Bluetooth Low Energy

- Peripheral
  - the low power device (a fitness tracker or a smartwatch)
  - can be connected to one central device
- Central
  - More processing power needed
  - Decides the connection parameters
  - Can maintain a connection with multiple peripherals

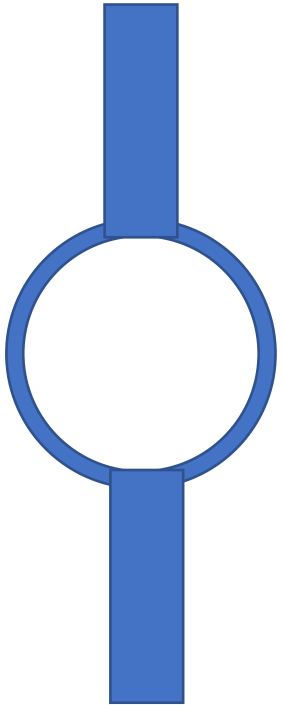


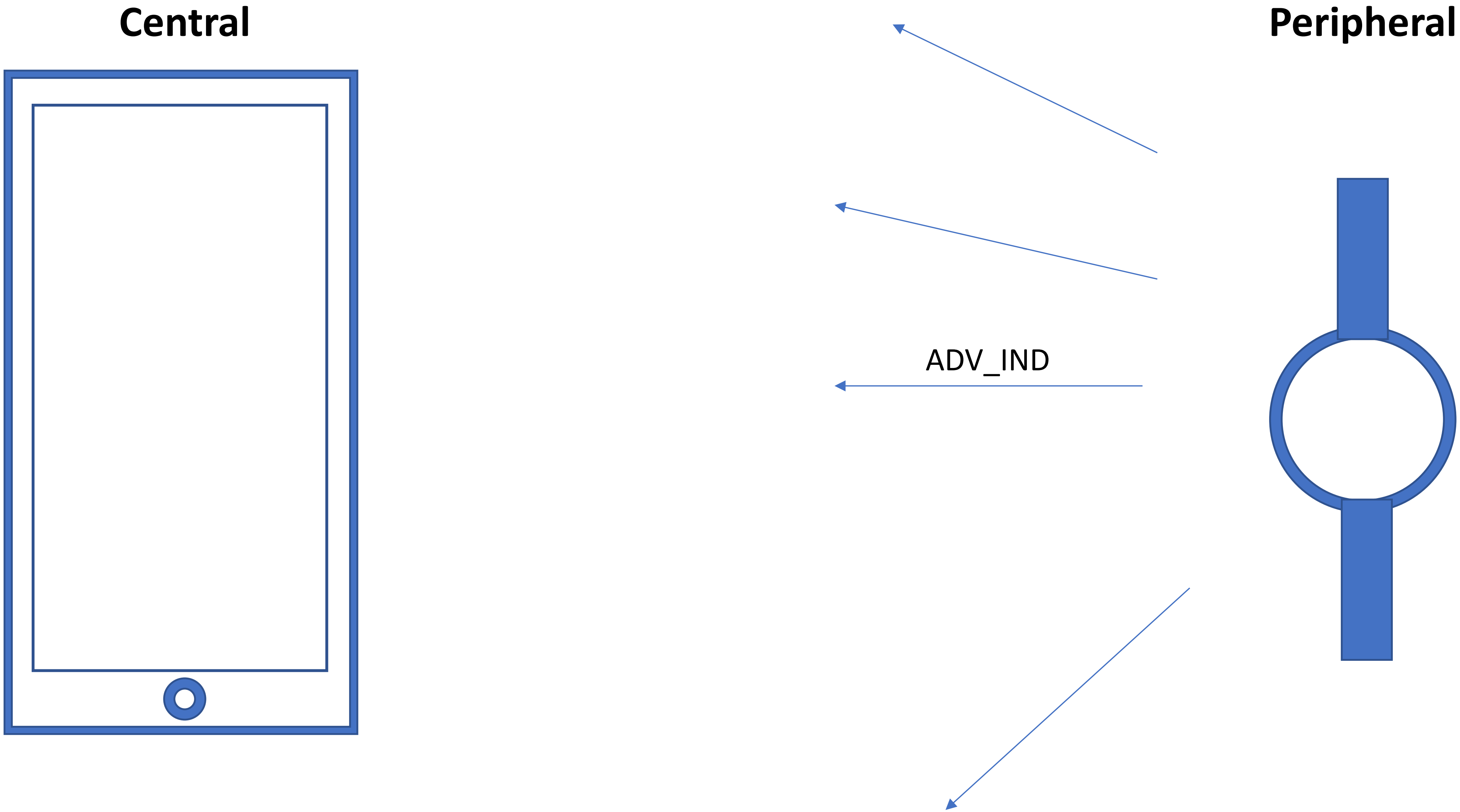


Central



Peripheral

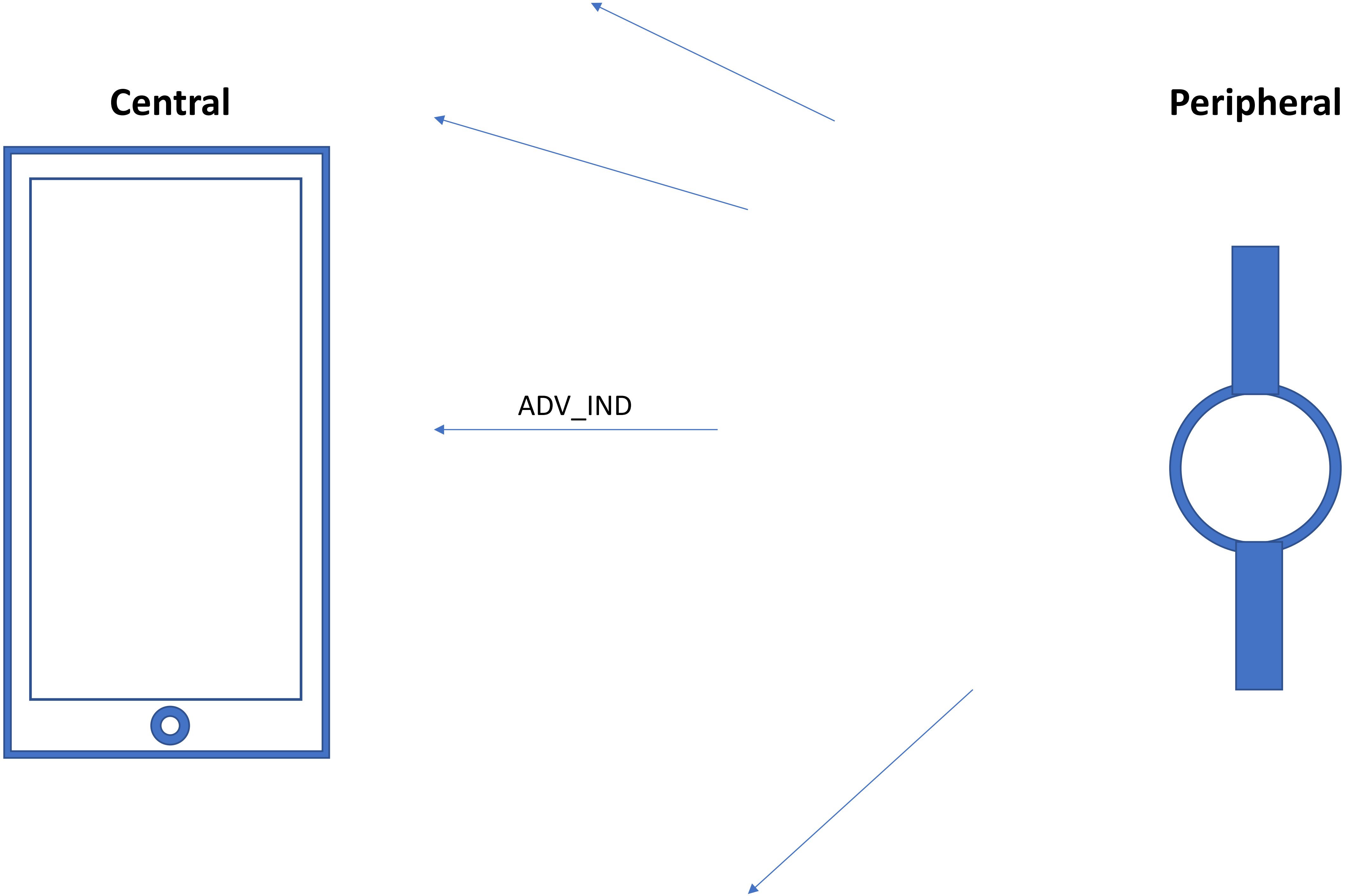




Central

Peripheral

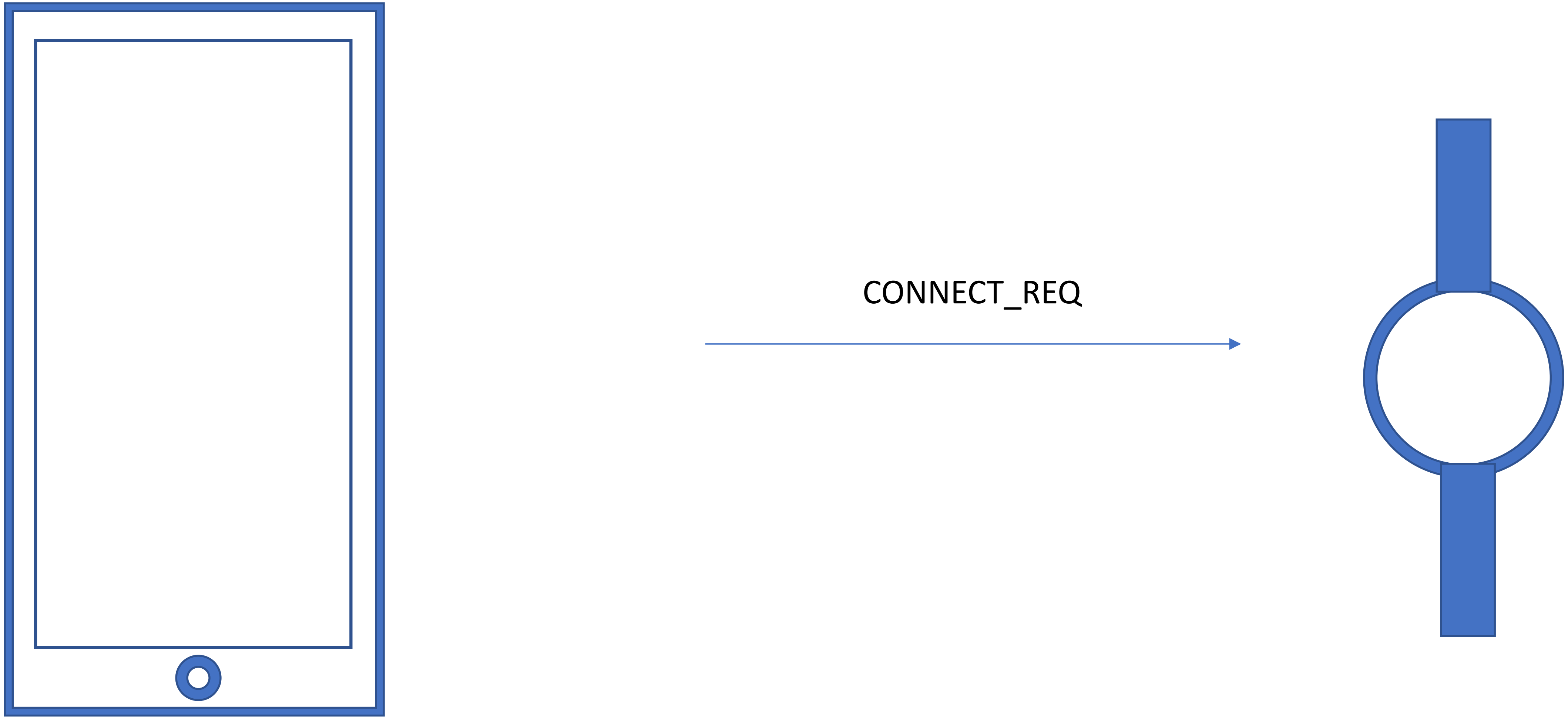
ADV_IND



Central

Peripheral

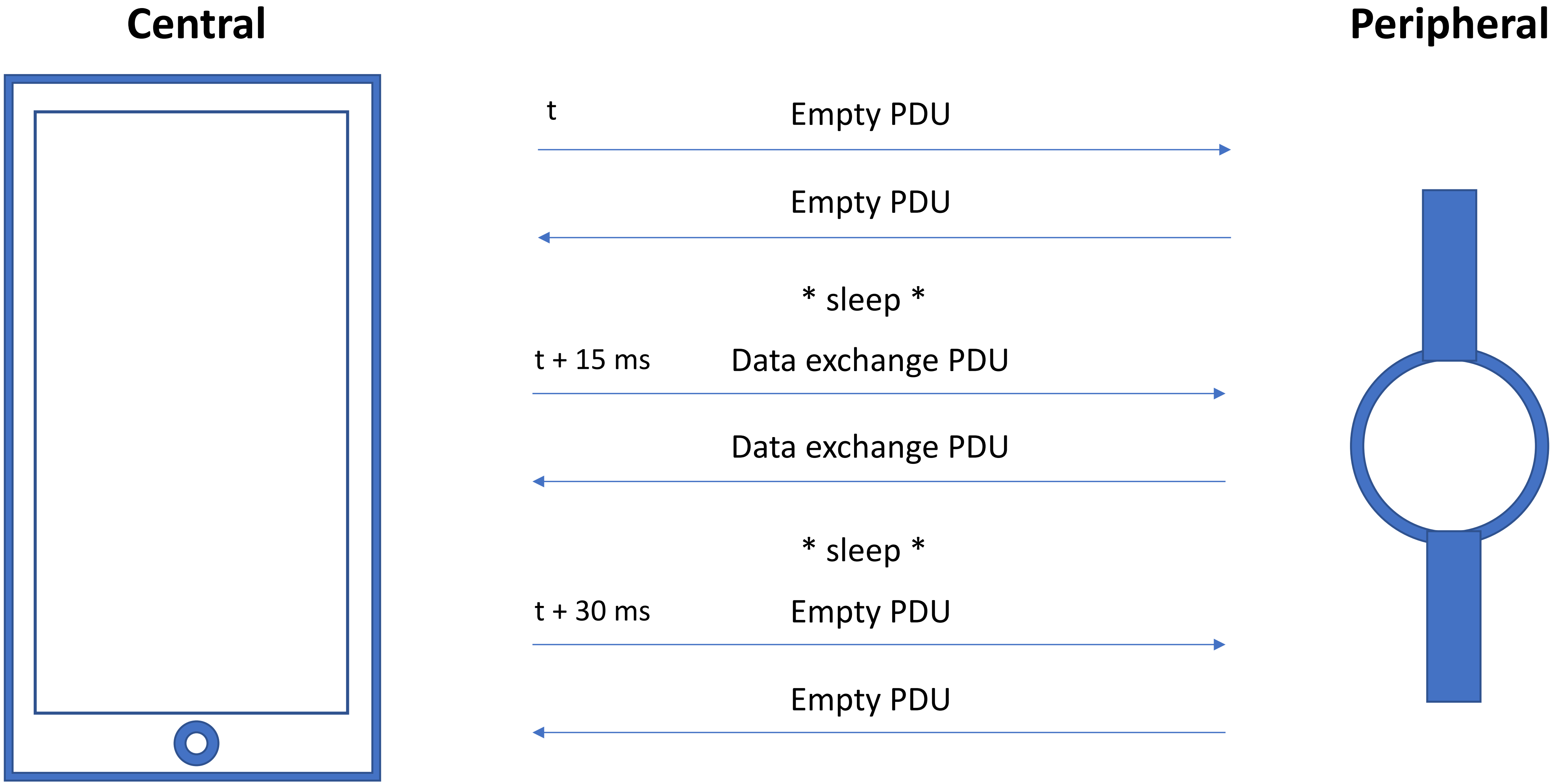
ADV_IND



Central

CONNECT_REQ

Peripheral



Central

Peripheral

t

Empty PDU

Empty PDU

* sleep *

t + 15 ms

Data exchange PDU

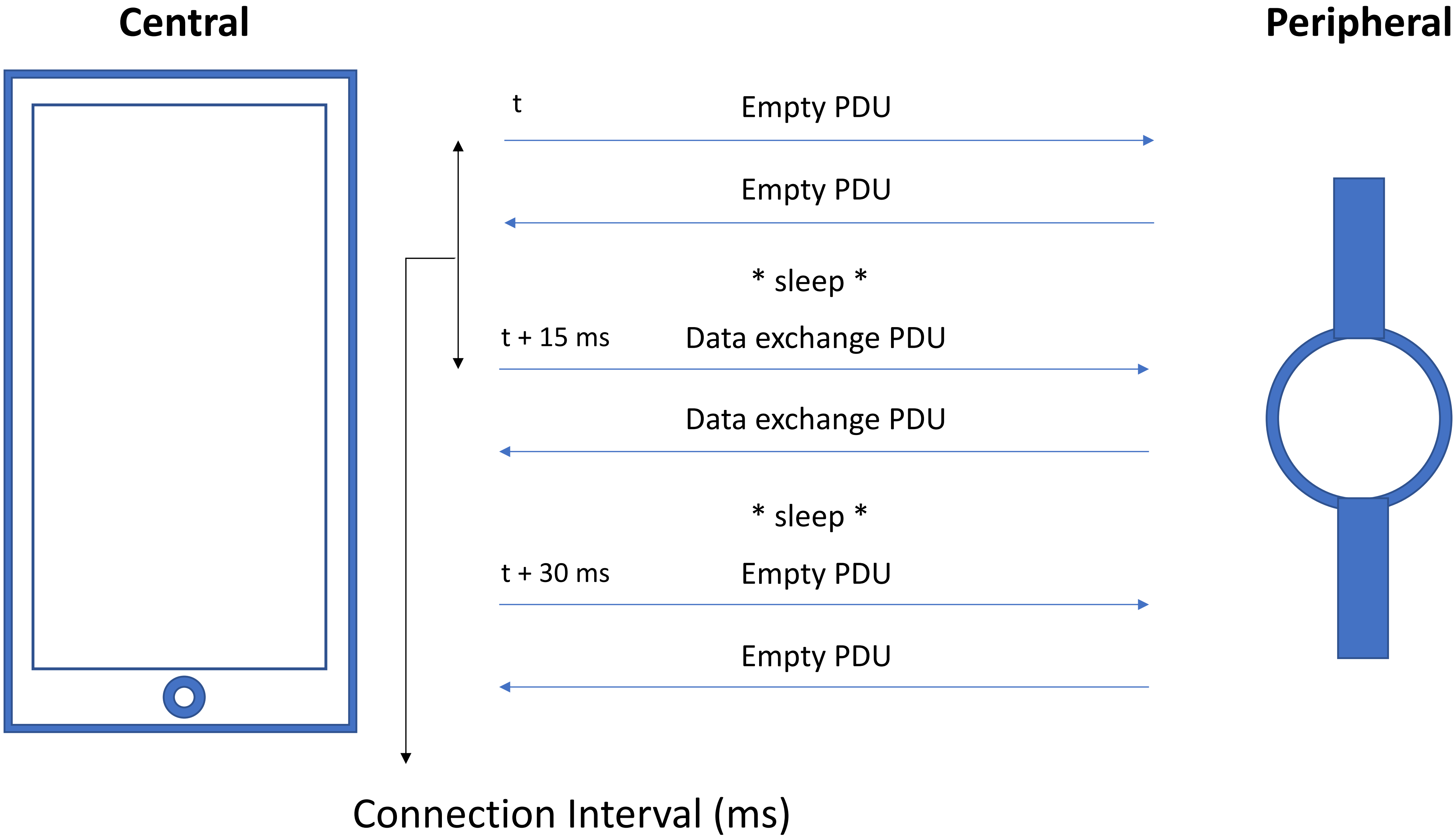
Data exchange PDU

* sleep *

t + 30 ms

Empty PDU

Empty PDU



Central

Peripheral

t

Empty PDU

Empty PDU

* sleep *

t + 15 ms

Data exchange PDU

Data exchange PDU

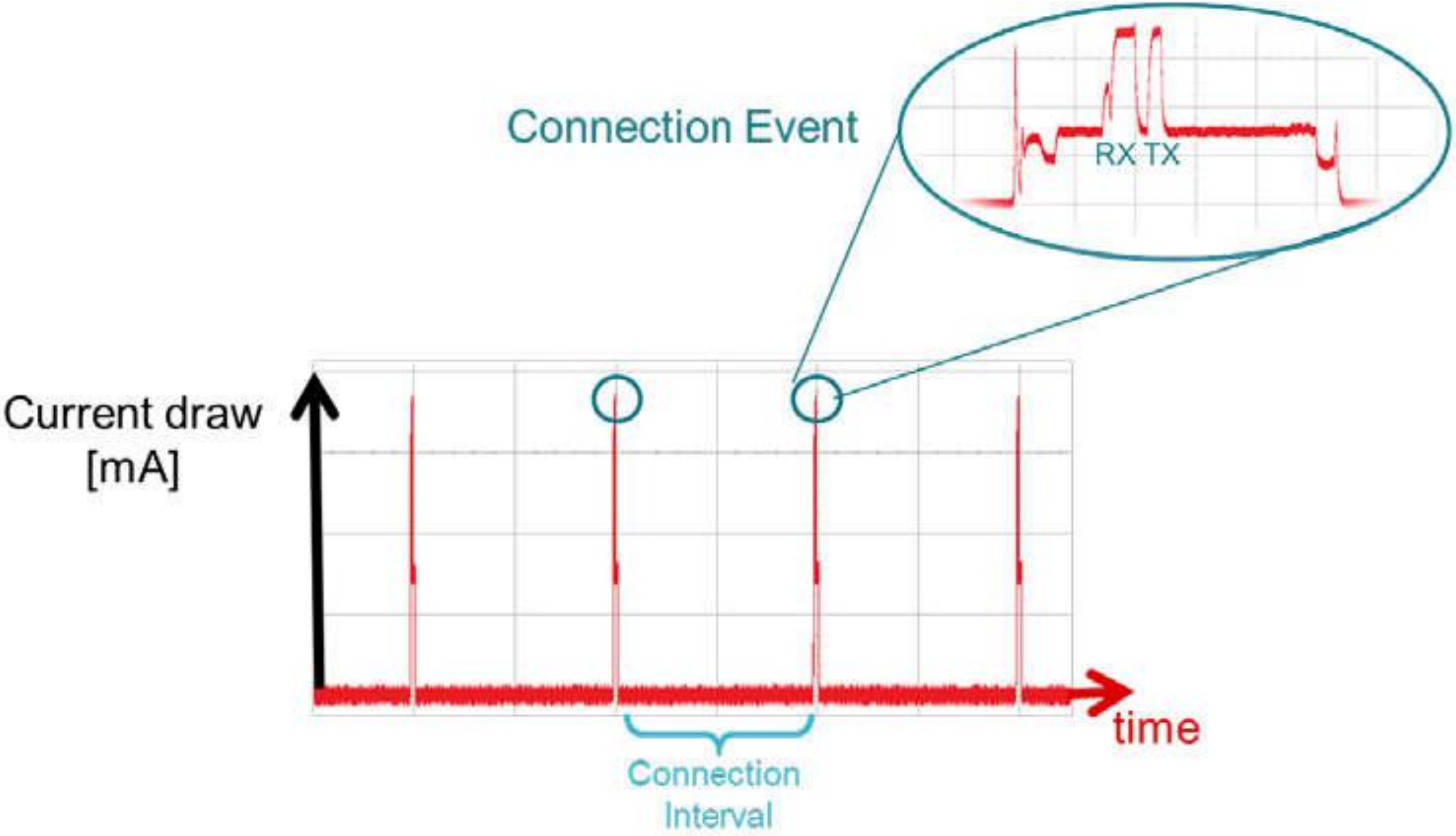
* sleep *

t + 30 ms

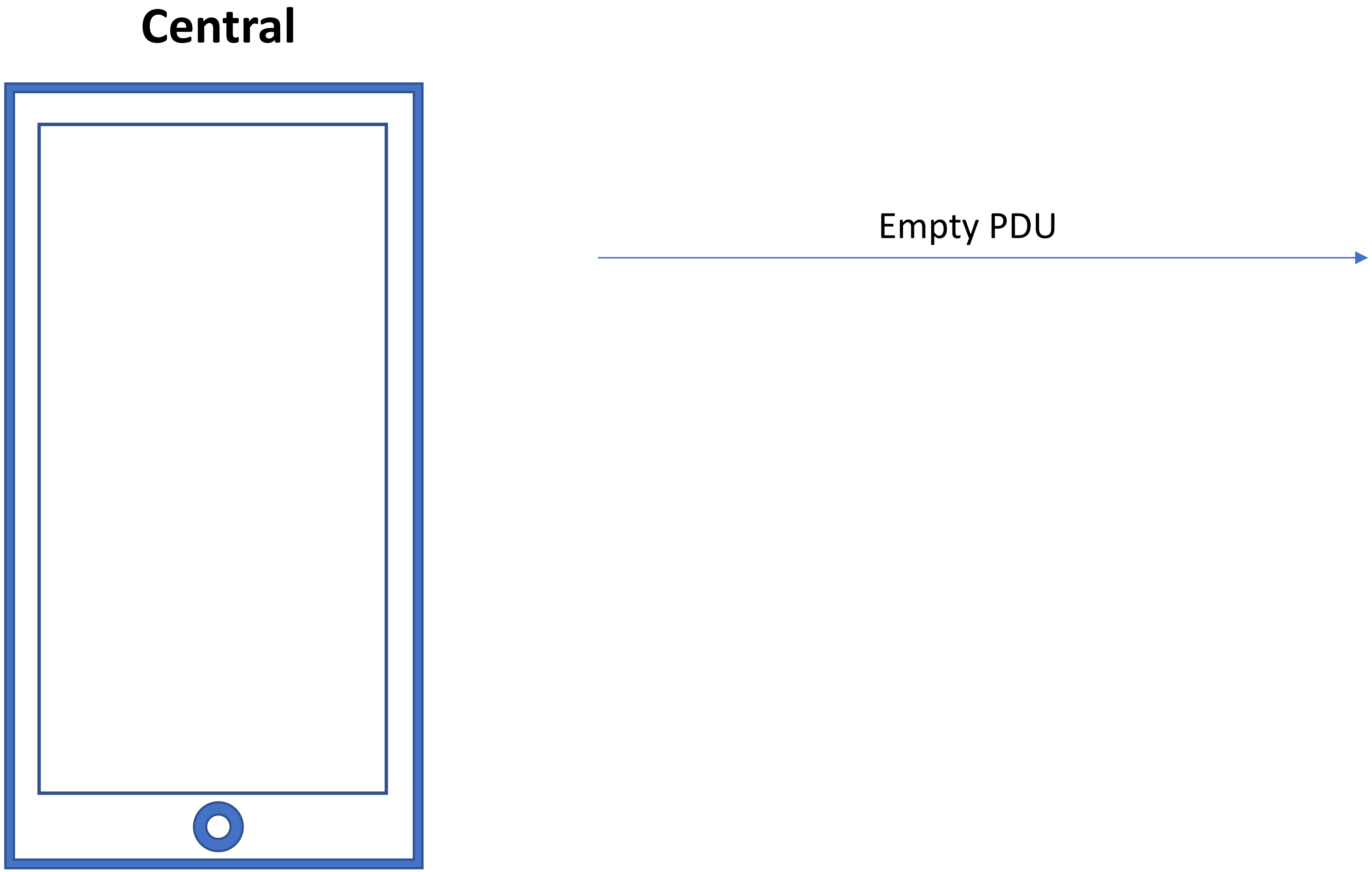
Empty PDU

Empty PDU

Connection Interval (ms)

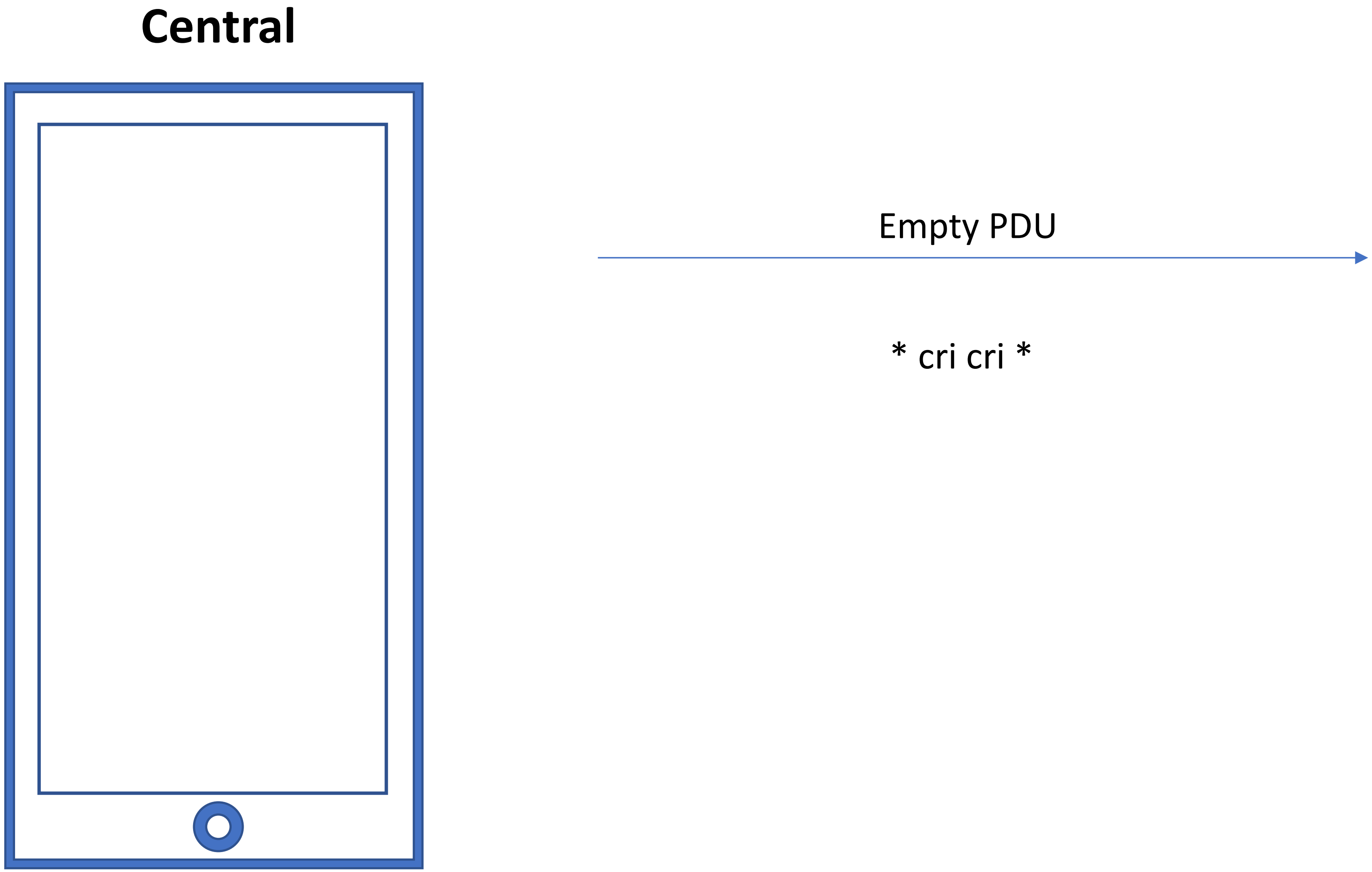


Source: http://dev.ti.com



Central

Empty PDU

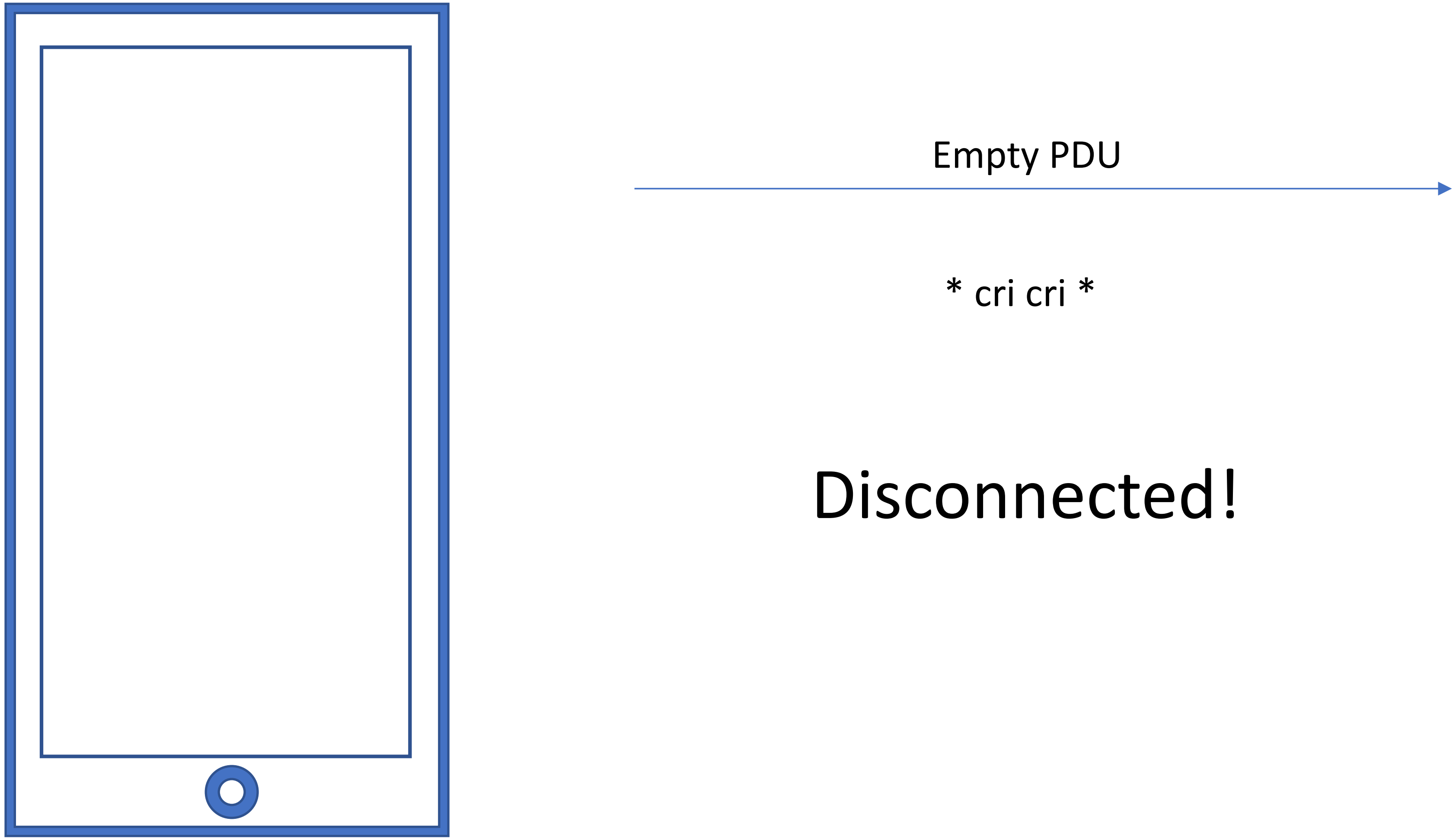


Central

Empty PDU

* cri cri *

Central



Empty PDU

* cri cri *

Disconnected!

Introduction Bluetooth Low Energy

- Two roles: Central and Peripheral
- Connection interval

Introduction Bluetooth Low Energy

- GATT (Generic Attribute Profile)
- Service, Characteristic
- GATT Server / GATT Client

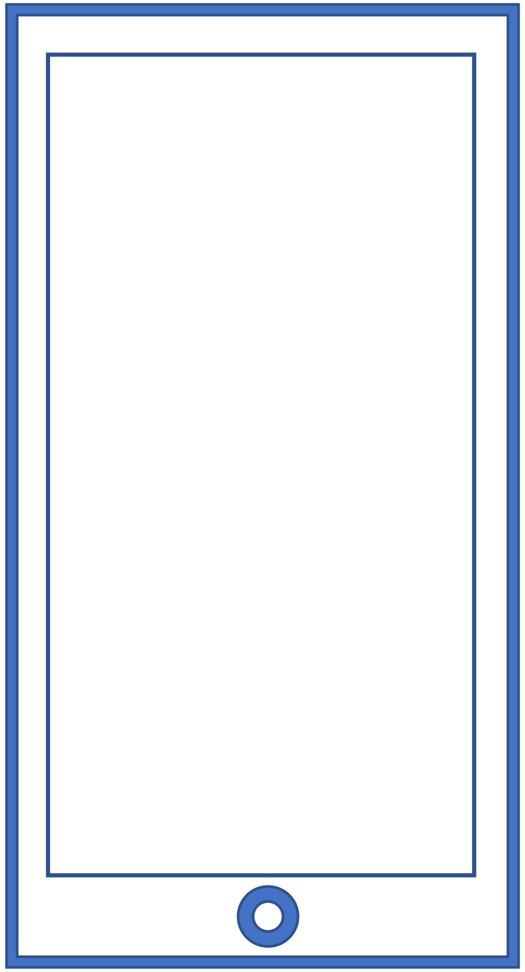
```
struct Service {
        let uuid: UUID
        let characteristics: [Characterisic]
}

struct Characteristic {
        let uuid: UUID
        let value: Data
}
```

Introduction Bluetooth Low Energy

- Example:
- GATT Server
  - Accelerometer Service
    - X-axis Characteristic
    - Y-axis Characteristic
    - Z-axis Characteristic
  - Outside Info Service
    - Temperature Characteristic
    - Humidity Characteristic

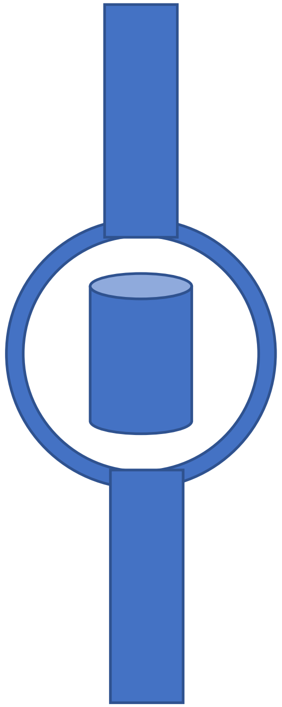
Central

GATT Client



Peripheral

GATT Server

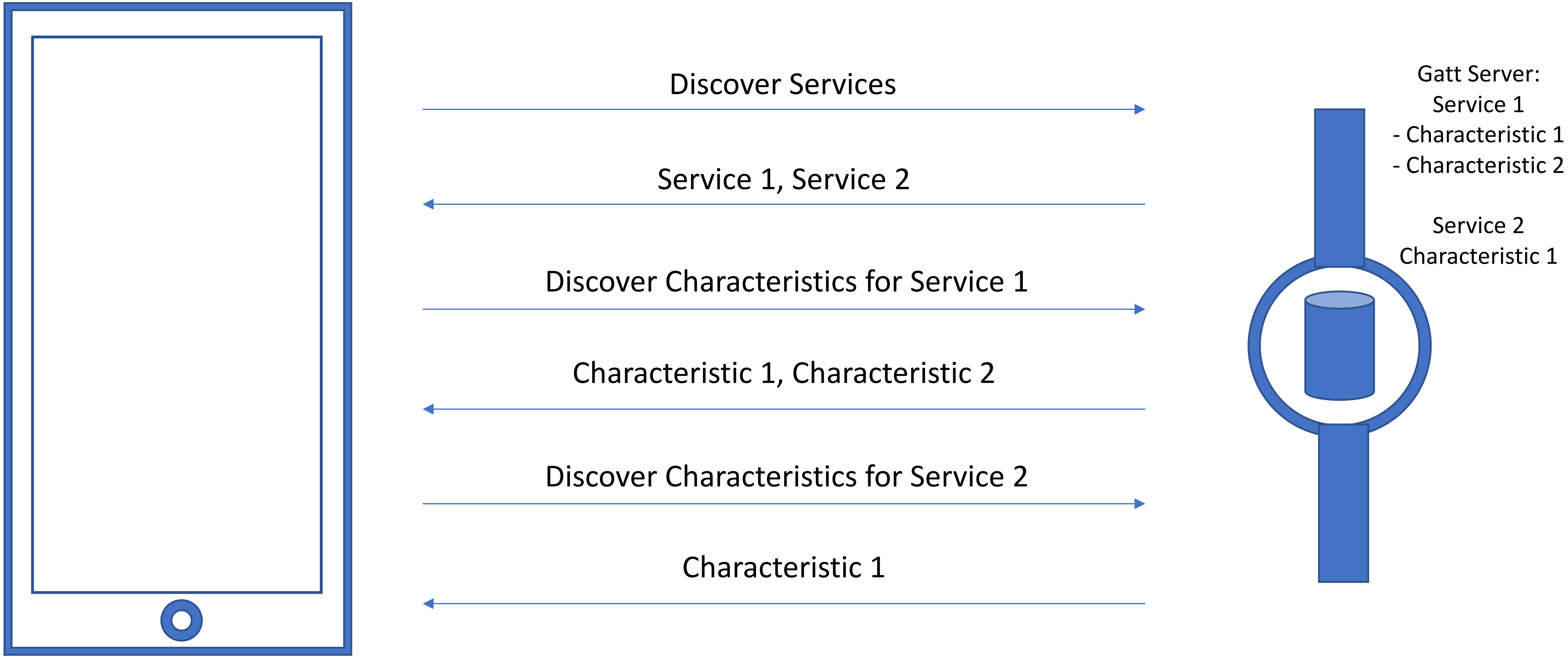


Central

GATT Client

Peripheral

GATT Server



Discover Services

Service 1, Service 2

Discover Characteristics for Service 1

Characteristic 1, Characteristic 2

Discover Characteristics for Service 2

Characteristic 1

Gatt Server:
Service 1
- Characteristic 1
- Characteristic 2

Service 2
Characteristic 1

Introduction Bluetooth Low Energy

- GATT characteristic properties:
  - Read
  - Write without response
  - Write
  - Notify
  - Indicate
  - …

Introduction Bluetooth Low Energy

- GATT protocol
  - Profile
  - Service
  - Characteristic
  - GATT Client
  - GATT Server

- Heart Rate Sensor Profile
  - Heart rate measurement service
    - Heart Rate Measurement Characteristic
    - Body Sensor Location Characteristic
    - Heart Rate Control Point Characteristic
  - Device Information Service
    - Device Information Service Characteristic

CoreBluetooth.framework

- Core Bluetooth
  - CBCentralManager – master + GATT Client
  - CBPeripheralManager – peripheral + GATT Server

CBCentralManager

- CBCentralManagerDelegate
  - did update state
  - did discover peripheral
  - did connect to peripheral
  - did disconnect from peripheral
- CBPeripheral
  - connect
  - discover services
  - discover characteristics
  - read
  - write
- CBPeripheralDelegate
  - Did discover services
  - Did discover characteristics
  - Did update value for characteristic

CBPeripheralManager

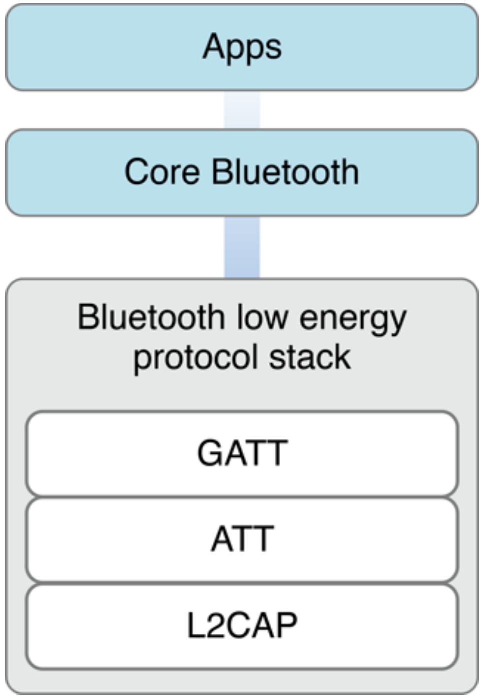
- CBPeripheralManagerDelegate
  - Did update state
  - Did add service
  - Did start advertising
  - Did subscribe / unsubscribe to characteristic
  - Did receive read
  - Did receive write
- CBCentral

Core Bluetooth Background Modes

- Two modes:
  - bluetooth-central
  - bluetooth-peripheral
- Needs to be declared in the plist
- App is woken up in background on Bluetooth events

Recent BLE Improvements in iOS

- Before iOS 11:
  - only GATT
- From iOS 11:
  - we have access to L2CAP



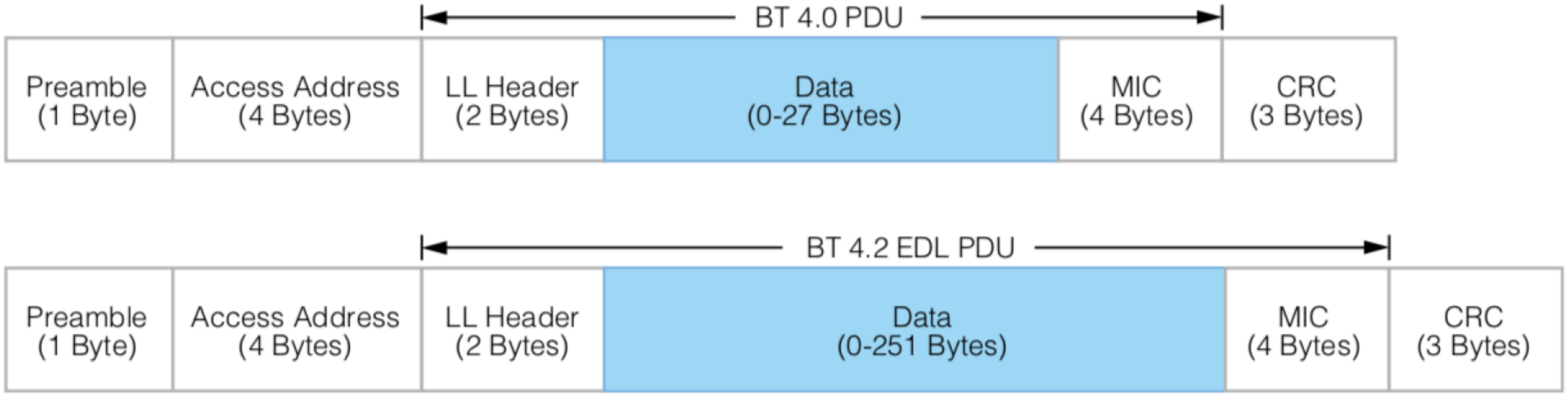
Recent BLE Improvements in iOS

- write without response = no ACK
- no feedback from Core Bluetooth API when buffers are full

- Solution in iOS 11:
  - CBPeripheral canSendWriteWithoutResponse API

Recent BLE Improvements in iOS

- Data length extension
- Enhancement introduced in the *Bluetooth 4.2* specification
- Increases the maximum data length from 27 to 251
- Improves throughput
- The remote accessory must support DLE
- Supported on iPhone 7 and iPhone 7 Plus devices



BT 4.0 PDU

Preamble (1 Byte)	Access Address (4 Bytes)	LL Header (2 Bytes)	Data (0-27 Bytes)	MIC (4 Bytes)	CRC (3 Bytes)

BT 4.2 EDL PDU

Preamble (1 Byte)	Access Address (4 Bytes)	LL Header (2 Bytes)	Data (0-251 Bytes)	MIC (4 Bytes)	CRC (3 Bytes)

ANCS – Apple Notification Center Service

- GATT Service hosted by the iOS
- Responsible for sending notifications that are generated on iOS devices.
- Characteristics
  - Notification Source (notifiable)
  - Control Point (writeable without response)
  - Data Source (notifiable)
- No code needed in the iOS App.
- Impossible to discover the service from an iOS / macOS app.

Other standard BLE profiles

- Time Profile
    - Can be used to get information about phone’s time
        - Timezone
        - DST
        - Time
- Battery Profile

Useful resources

- Apple Bluetooth Design Guidelines
  - useful information about what connection parameters iOS devices can accept
- ANCS Protocol
- Core Bluetooth documentation
- Bluetooth LE Advertising and Connection Parameters for a stable connection
- What's new in Core Bluetooth – WWDC 2017

Thank you!