



链滴

# iOS 蓝牙开发中 app 作为外设被连接怎么实现?

作者: [kdnkms](#)

原文链接: <https://ld246.com/article/1446703990183>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

在 iOS 应用开发中，蓝牙开发是一个重点，是 iOS 开发人员必须掌握的一个知识。今天小编就为大家分享一篇 iOS 大牛写的有关蓝牙开发中，使用 app 发布一个 peripheral，给其他的 central 接的实现方法。

peripheral 模式的流程

<ol>

<li>

打开 peripheralManager，设置 peripheralManager 的委托

</li>

<li>

创建 characteristics，characteristics 的 description 创建 service，把 characteristics 添加到 service 中，再把 service 添加到 peripheralManager 中

</li>

<li>

开启广播 advertising

</li>

<li>

对 central 的操作进行响应

<ul>

<li>

4.1 读 characteristics 请求

</li>

<li>

4.2 写 characteristics 请求

</li>

<li>

4.4 订阅和取消订阅 characteristics

</li>

</ul>

</li>

</ol>

准备开发环境

1、Xcode

2、开发证书和手机（蓝牙程序需要使用真机调试，使用模拟器也可以调试，但是方法很负，本篇分享不做过多赘述），如果不行可以使用 osx 程序调试

3、蓝牙外设

实现步骤

<ol>

<li>

打开 peripheralManager，设置 peripheralManager 的委托。设置当前 ViewController 实现 BPeripheralManagerDelegate 委托

@interface BePeripheralViewController : UIViewController

</li>

</ol>

初始化 peripheralManager

```
<code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl"> /*
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> 和CBCentralManager类似，蓝牙设备打开需要一定时间，打开成功后会进入委托方法
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> - (void)peripheralManagerDidUpdateState:(CBPeripheralManager *)peripheral;
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl"> 模拟器永远也不
```

得CBPeripheralManagerStatePoweredOn状态

```
</span></span><span class="highlight-line"><span class="highlight-cl"> */
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">peripheralManage
= [[CBPeripheralManager alloc] initWithDelegate:self queue:nil];
</span></span></code></pre>
<ol start="2">
<li>创建 characteristics, characteristics 的 description , 创建 service, 把 characteristics 添加到
service 中, 再把 service 添加到 peripheralManager 中</li>
</ol>
<p>在委托方法 - (void)peripheralManagerDidUpdateState:(CBPeripheralManager *)peripheral
中, 当 peripheral 成功打开后, 才可以配置 service 和 characteristics。这里创建的 service 和 cha
a 对象是 CBMutableCharacteristic 和 CBMutableService。他们的区别就像 NSArray 和 NSMutable
Array 区别类似。我们先创建 characteristics 和 description, description 是 characteristics 的描
, 描述分很多种, 这里不细说了, 常用的就是 CBUUIDCharacteristicUserDescriptionString。</p>
<p>//peripheralManager 状态改变</p>
<ul>
<li>
<p>(void)peripheralManagerDidUpdateState:(CBPeripheralManager *)peripheral{</p>
<p>switch (peripheral.state) {</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">
//在这里判断蓝牙设别的状态 当开启了则可调用 setUp方法(自定义)
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> case CBPeriphera
ManagerStatePoweredOn:
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     NSLog(@"po
wered on");
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     [info setText:[
NSString stringWithFormat:@"设备名%@已经打开, 可以使用center进行连接",LocalNameKey]];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     [self setUp];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     break;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> case CBPeriphera
ManagerStatePoweredOff:
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     NSLog(@"po
wered off");
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     [info setText:
"powered off"];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     break;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     default:
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl">     break;
</span></span></code></pre>
<p>}</p>
</li>
</ul>
```

```

<p></p>
<p>//配置 bluetooth 的</p>
<p>-(void)setup{<br>
//characteristics 字段描述</p>
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl">
CBUUID *CBUUIDCharacteristicUserDescriptionStringUUID = [CBUUID UUIDWithString
CBUUIDCharacteristicUserDescriptionString];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> /*
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 可以通知的Char
characteristic
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> properties: C
CharacteristicPropertyNotify
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> permissions C
AttributePermissionsReadable
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> */
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> CBMutableChar
acteristic *notifyCharacteristic = [[CBMutableCharacteristic alloc] initWithType:[CBUUID UUIDWi
thString:notifyCharacteristicUUID] properties:CBCharacteristicPropertyNotify value:nil permissi
ons:CBAttributePermissionsReadable];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> /*
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> 可读写的charac
teristics
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> properties: C
CharacteristicPropertyWrite | CBCharacteristicPropertyRead
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> permissions C
AttributePermissionsReadable | CBAttributePermissionsWritable
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> */
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> CBMutableChar
acteristic *readwriteCharacteristic = [[CBMutableCharacteristic alloc] initWithType:[CBUUID UI
WithString:readwriteCharacteristicUUID] properties:CBCharacteristicPropertyWrite | CBCharac
teristicPropertyRead value:nil permissions:CBAttributePermissionsReadable | CBAttributePermi
ssionsWritable];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> //设置descripti
n
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> CBMutableDesc
riptor *readwriteCharacteristicDescription1 = [[CBMutableDescriptor alloc] initWithType: CBUU
DCharacteristicUserDescriptionStringUUID value:@"name"];
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
</span> </span> <span class="highlight-line"> <span class="highlight-cl"> [readwriteChar
acteristic setDescriptors:@[readwriteCharacteristicDescription1]];

```

```

</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> /*
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> 只读的Characteristic
istic
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> properties: C
CharacteristicPropertyRead
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> permissions C
AttributePermissionsReadable
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> */
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> CBMutableChar
acteristic *readCharacteristic = [[CBMutableCharacteristic alloc] initWithType:[CBUUID UUIDWith
String:readCharacteristicUUID] properties:CBCharacteristicPropertyRead value:nil permissions
CBAttributePermissionsReadable];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> //service1初始
并加入两个characteristics
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> CBMutableServi
e *service1 = [[CBMutableService alloc] initWithType:[CBUUID UUIDWithString:ServiceUUID1]
rimary:YES];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [service1 setCha
acteristics:@[notiyCharacteristic,readwriteCharacteristic]];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> //service2初始
并加入一个characteristics
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> CBMutableServi
e *service2 = [[CBMutableService alloc] initWithType:[CBUUID UUIDWithString:ServiceUUID2]
rimary:YES];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [service2 setCha
acteristics:@[readCharacteristic]];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> //添加后就会调
代理的- (void)peripheralManager:(CBPeripheralManager *)peripheral didAddService:(CBService
*)service error:(NSError *)error
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [peripheralMan
ger addService:service1];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [peripheralMan
ger addService:service2];
</span></span></code></pre>
<p></p>
<ol start="3">
<li>开启广播 advertising</li>
</ol>
<p>//perihpheral 添加了 service</p>

```

```

<ul>
<li>
<p>(void)peripheralManager:(CBPeripheralManager *)peripheral didAddService:(CBService *)
ervice error:(NSError *)error{</p>
<p>if (error == nil) {</p>
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight
cl"> serviceNum++;
</span></span></code> </pre>
<p>}</p>
<p>//因为我们添加了 2 个服务，所以想两次都添加完成后才去发送广播</p>
<p>if (serviceNum==2) {</p>
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight
cl"> //添加服务后可以在此向外界发出通告 调用完这个方法后会调用代理的
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl"> //(void)periphera
ManagerDidStartAdvertising:(CBPeripheralManager *)peripheral error:(NSError *)error
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl"> [peripheralMana
er startAdvertising:@{
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl">
    CBAvertisementDataServiceUUIDsKey : @[CBUUID UUIDWithString:ServiceUUID1],[CB
UID UUIDWithString:ServiceUUID2]],
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl">
    CBAvertisementDataLocalNameKey : LocalNameKey
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl">
    }
</span></span><span class="highlight-line"> <span class="highlight-cl">
</span></span><span class="highlight-line"> <span class="highlight-cl"> ];
</span></span></code> </pre>
<p>}</p>
</li>
</ul>
<p></p>
<p>//peripheral 开始发送 advertising</p>
<ul>
<li>
<p>(void)peripheralManagerDidStartAdvertising:(CBPeripheralManager *)peripheral error:(N
Error *)error{</p>
<p>NSLog(@"in peripheralManagerDidStartAdvertisiong");</p>
</li>
</ul>
<p></p>
<ol start="4">
<li>对 central 的操作进行响应</li>
</ol>
<ul>
<li>
<p>4.1 读 characteristics 请求</p>
</li>
<li>
<p>4.2 写 characteristics 请求</p>

```

</li>

<li>

<p>4.3 订阅和取消订阅 characteristics</p>

</li>

</ul>

<p>//订阅 characteristics</p>

<p>-(void)peripheralManager:(CBPeripheralManager \*)peripheral central:(CBCentral \*)central didSubscribeToCharacteristic:(CBCharacteristic \*)characteristic{</p>

```
<code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">NSLog(@"订阅了 %@的数据",characteristic.UUID);
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">//每秒执行一次给设备发送一个当前时间的秒数
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">timer = [NSTimer scheduledTimerWithTimeInterval:1 target:self selector:@selector(sendData:) userInfo:characteristic repeats:YES];
```

```
</span></span></code></pre>
```

<p></p>

<p>//取消订阅 characteristics</p>

<p>-(void)peripheralManager:(CBPeripheralManager \*)peripheral central:(CBCentral \*)central didUnsubscribeFromCharacteristic:(CBCharacteristic \*)characteristic{</p>

```
<code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">NSLog(@"取消订阅 %@的数据",characteristic.UUID);
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">//取消回应
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">[timer invalidate];
```

```
</span></span></code></pre>
```

<p></p>

<p>//发送数据，发送当前时间的秒数</p>

<p>-(BOOL)sendData:(NSTimer \*)t {</p>

```
<code class="highlight-chroma"><span class="highlight-line"><span class="highlight-cl">CBMutableCharacteristic *characteristic = t.userInfo;
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">NSDateFormatter dft = [[NSDateFormatter alloc] init];
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">[dft setDateForma:@ "ss"];
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">NSLog(@"%@",[dft stringFromDate:[NSDate date]]);
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">//执行回应Central知数据
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">return [peripheralManager updateValue:[dft stringFromDate:[NSDate date]] dataUsingEncoding:NSUTF8StringEncoding] forCharacteristic:(CBMutableCharacteristic *)characteristic
```

```
</span></span><span class="highlight-line"><span class="highlight-cl">onSubscribedCentals:nil];
```

```
</span></span></code></pre>
```

<p><br>

```

//读 characteristics 请求</p>
<ul>
<li>
<p>(void)peripheralManager:(CBPeripheralManager *)peripheral didReceiveReadRequest:(CB
TTRequest *)request{</p>
<p>NSLog(@"didReceiveReadRequest");</p>
<p>//判断是否有读数据的权限</p>
<p>if (request.characteristic.properties & CBCharacteristicPropertyRead) {</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> NSData *data = request.characteristic.value;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [request setValue
data];
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> //对请求作出成
响应
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [peripheralMana
er respondToRequest:request withResult:CBATTErrorsuccess];
</span></span></code></pre>
<p>}else{</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> [peripheralManager respondToRequest:request withResult:CBATTErrortWriteNotPermitte
];
</span></span></code></pre>
<p>}</p>
</li>
</ul>
<p></p>
<p>//写 characteristics 请求</p>
<ul>
<li>
<p>(void)peripheralManager:(CBPeripheralManager *)peripheral didReceiveWriteRequests:(N
Array *)requests{</p>
<p>NSLog(@"didReceiveWriteRequests");</p>
<p>CBATTRequest *request = requests[0];</p>
<p>//判断是否有写数据的权限</p>
<p>if (request.characteristic.properties & CBCharacteristicPropertyWrite) {</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> //需要转换成CBMutableCharacteristic对象才能进行写值
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> CBMutableChara
cteristic *c =(CBMutableCharacteristic *)request.characteristic;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> c.value = request
value;
</span></span><span class="highlight-line"><span class="highlight-cl">
</span></span><span class="highlight-line"><span class="highlight-cl"> [peripheralMana
er respondToRequest:request withResult:CBATTErrorsuccess];
</span></span></code></pre>
<p>}else{</p>
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight
cl"> [peripheralManager respondToRequest:request withResult:CBATTErrortWriteNotPermitte
];
</span></span></code></pre>

```



```
</span></span></code></pre>
```

```
<p></p>
```

```
</li>
```

```
</ul>
```

```
<p></p>
```

以上就是 iOS 蓝牙开发中，app 作为外设被连接的具体实现方法，希望对大家开发这项功能有帮助吧。

相关文章: 《DASH 中查询 iOS 文档的方法》 [https://ld246.com/forward?goto=http%3A%2F%2Fwww.maiziedu.com%2Fgroup%2Farticle/5618/](https://ld246.com/forward?goto=http%3A%2F%2Fwww.maiziedu.com%2Fgroup%2Farticle%2F5618%2F)