

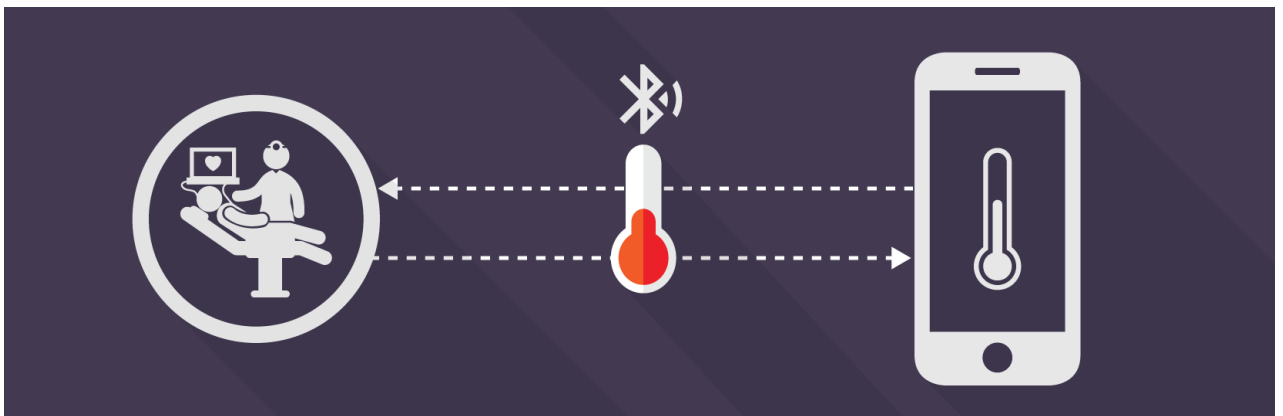
## TUTORIAL 5: Demo thiết bị nhiệt kế y tế với VBLUno

### Nội dung trình bày:

- Giới thiệu các nét chính về nhiệt kế y tế (Health Thermometer – HTM);
- Ví dụ demo thiết bị nhiệt kế y tế với VBLUno.

### 1. Giới thiệu các nét chính về nhiệt kế y tế - HTM

Nhiệt kế y tế là thiết bị có chức năng đo nhiệt độ cơ thể, phục vụ chuẩn đoán sức khỏe và các ứng dụng liên quan đến thể dục thể thao (fitness). Bài viết này nói đến thiết bị nhiệt kế y tế thông minh, đo nhiệt độ cơ thể và gửi thông tin cần thiết đến các thiết bị thông minh (Smartphone, Tablet, PC). Chuẩn truyền thông không dây Bluetooth Low Energy (BLE) thường được sử dụng trong các thiết bị này.



*Hình 1: Hoạt động của thiết bị nhiệt kế y tế qua giao tiếp BLE*

Các nhiệt kế y tế này sẽ hoạt động như GATT (Generic Attribute Profile) Server, trong khi các thiết bị Central sẽ hoạt động như GATT Client. Nhiệt kế sẽ có hai services là Health Thermometer Service và Device Information Service.

Đầu tiên, ứng dụng tại Central tiến hành khám phá (discovery) các services có trên nhiệt kế, và sau đó sẽ tiến hành khám phá các characteristics đi kèm. Với Health Thermometer Service (HTS), các characteristics sau có thể được tích hợp: temperature measurement, temperature type, intermediate temperature and measurement interval characteristics. Trong đó temperature measurement là characteristic quan trọng nhất của service này, giúp cung cấp thông tin về giá trị nhiệt độ đo được của cảm biến.

Chi tiết về HTS và các characteristics tương ứng được trình bày chi tiết tại đường dẫn sau:

[https://www.bluetooth.com/specifications/gatt/viewer?attributeXmlFile=org.bluetooth.service.health\\_thermometer.xml](https://www.bluetooth.com/specifications/gatt/viewer?attributeXmlFile=org.bluetooth.service.health_thermometer.xml)

### Service Characteristics

Overview	Properties																				
<p><b>Name:</b> Temperature Measurement</p> <p><b>Description:</b> This characteristic is used to send a temperature measurement.</p> <p><b>Type:</b> <a href="#">org.bluetooth.characteristic.temperature_measurement</a></p> <p><b>Requirement:</b> Mandatory</p>	<table> <tr> <th>Property</th><th>Requirement</th></tr> <tr> <td>Read</td><td>Excluded</td></tr> <tr> <td>Write</td><td>Excluded</td></tr> <tr> <td>WriteWithoutResponse</td><td>Excluded</td></tr> <tr> <td>SignedWrite</td><td>Excluded</td></tr> <tr> <td>Notify</td><td>Excluded</td></tr> <tr> <td>Indicate</td><td>Mandatory</td></tr> <tr> <td>WritableAuxiliaries</td><td>Excluded</td></tr> <tr> <td>Broadcast</td><td>Excluded</td></tr> <tr> <td>ExtendedProperties</td><td></td></tr> </table>	Property	Requirement	Read	Excluded	Write	Excluded	WriteWithoutResponse	Excluded	SignedWrite	Excluded	Notify	Excluded	Indicate	Mandatory	WritableAuxiliaries	Excluded	Broadcast	Excluded	ExtendedProperties	
Property	Requirement																				
Read	Excluded																				
Write	Excluded																				
WriteWithoutResponse	Excluded																				
SignedWrite	Excluded																				
Notify	Excluded																				
Indicate	Mandatory																				
WritableAuxiliaries	Excluded																				
Broadcast	Excluded																				
ExtendedProperties																					

Hình 2: Thông tin chi tiết về các characteristic của HTS

Hình trên cho thấy kiểu của Temperature Measurement characteristic bắt buộc phải là Indicate. Đây sẽ là thông tin quan trọng khi bạn lập trình.

### ***Kiểu giá trị nhiệt độ trong HTM***

Dữ liệu về sức khỏe cá nhân được truyền nhận theo một chuẩn riêng – IEEE 11073 Personal Health Data (PHD). Nhiệt kế y tế truyền giá trị nhiệt độ dưới dạng số thực với độ chính xác đơn (FLOAT), cụ thể là chuẩn ISO/IEEE 11073-20601. Bạn có thể tham khảo tài liệu chi tiết tại đây:

#### 1. ISO/IEEE 11073 Personal Health Data (PHD) Standards

---

[https://en.wikipedia.org/wiki/ISO/IEEE\\_11073\\_Personal\\_Health\\_Data\\_\(PHD\)\\_Standards](https://en.wikipedia.org/wiki/ISO/IEEE_11073_Personal_Health_Data_(PHD)_Standards)

#### 2. PERSONAL HEALTH DEVICES TRANSCODING WHITE PAPER

[https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwigtPf10IbPAhVKNpQKHUMMAxgQFggbMAA&url=https%3A%2F%2Fwww.bluetooth.org%2FDocMan%2Fhandlers%2FDownloadDoc.aspx%3Fdoc\\_id%3D272346&usg=AFQjCNHyUTOadbOChQMWt6yAp1WG79aYg&sig2=grd30PZDyCBMpcTQsreUXQ](https://www.google.com.vn/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwigtPf10IbPAhVKNpQKHUMMAxgQFggbMAA&url=https%3A%2F%2Fwww.bluetooth.org%2FDocMan%2Fhandlers%2FDownloadDoc.aspx%3Fdoc_id%3D272346&usg=AFQjCNHyUTOadbOChQMWt6yAp1WG79aYg&sig2=grd30PZDyCBMpcTQsreUXQ)

### **2. Ví dụ demo thiết bị nhiệt kế y tế với VBLUno.**

Trong tutorial trước, chúng ta đã cùng tìm hiểu cách sử dụng cảm biến nhiệt độ sẵn có bên trong chip nRF51822 của VBLUno. Trong phần này chúng ta sẽ sử dụng nó để minh họa một thiết bị nhiệt kế y tế (HTM) thông minh.

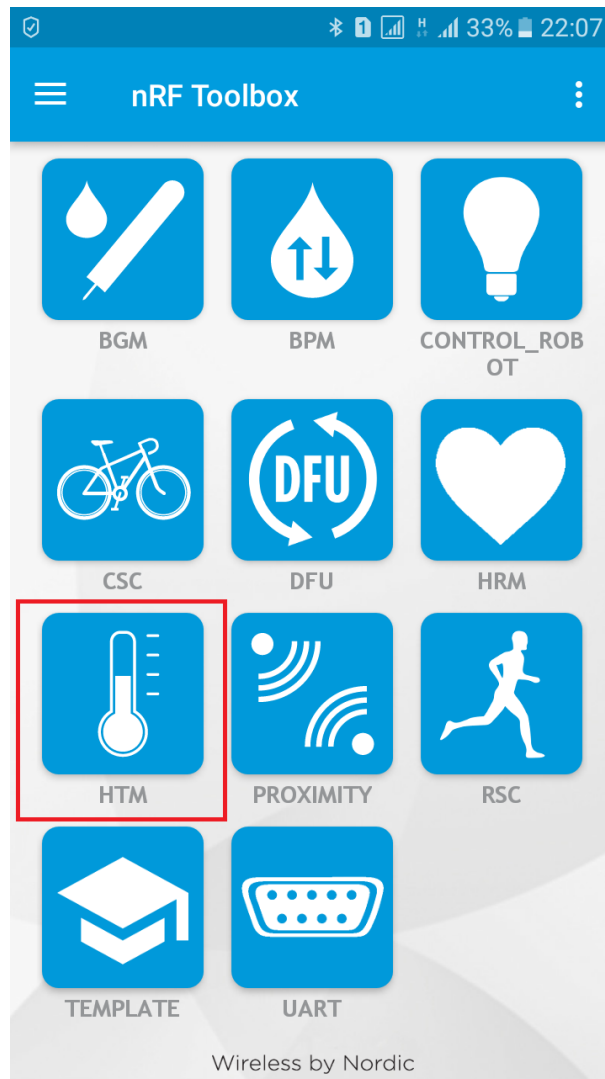
#### **Chạy demo:**

+ Tải mã nguồn sau về máy, tiến hành biên dịch và nạp vào VBLUno.

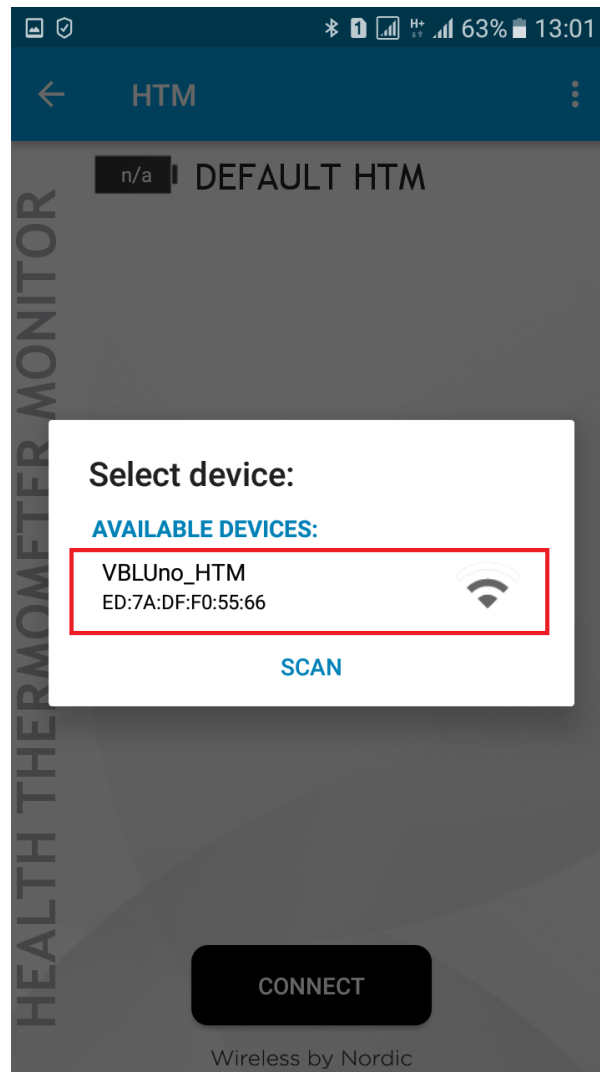
[https://github.com/VNGIoTLab/Arduino\\_VBLUno\\_nRF51822/blob/master/Tutorial/Source/VBLUno\\_Tut5\\_HTM.ino](https://github.com/VNGIoTLab/Arduino_VBLUno_nRF51822/blob/master/Tutorial/Source/VBLUno_Tut5_HTM.ino)

+ Đặt jump J7 về vị trí chạy application và nhấn nút Reset

+ Sử dụng chương trình nRF Toolbox của Nordic để kết nối và nhận dữ liệu từ thiết bị.



Hình 3: Giao diện chương trình nRF Toolbox



Hình 4: Ứng dụng tìm thấy thiết bị HTM



Hình 5: Thiết bị HTM đo nhiệt độ, tạo giả mức pin và gửi lên Central

**Giải thích một số điểm cơ bản:**

+ Phần khai báo các services và characteristics của thiết bị HTM (VBLUno)

Trong ví dụ này có cài đặt 2 services:

- Health Thermometer Services: Cung cấp giá trị về nhiệt độ đo được.
- Battery Level Services: Cung cấp thông tin về mức pin còn lại của thiết bị

```

/* Health Thermometer Service */
uint8_t      thermTempPayload[5] = { 0, 0, 0, 0, 0 };
GattCharacteristic thermTemp
(GattCharacteristic::UUID_TEMPERATURE_MEASUREMENT_CHAR, thermTempPayload,
    5, 5, GattCharacteristic::BLE_GATT_CHAR_PROPERTIES_INDICATE);
//Indicate
GattCharacteristic *thermChar[] = {&thermTemp};
GattService      thermService (GattService::UUID_HEALTH_THERMOMETER_SERVICE,
    thermChar, sizeof(thermChar)/sizeof(GattCharacteristic *));

/* Battery Level Service */
uint8_t      batt = 100; /* Battery level */
uint8_t      read_batt = 0; /* Variable to hold battery level reads */
GattCharacteristic battLevel ( GattCharacteristic::UUID_BATTERY_LEVEL_CHAR, (uint8_t
*)&batt, 1, 1,
    GattCharacteristic::BLE_GATT_CHAR_PROPERTIES_NOTIFY |
    GattCharacteristic::BLE_GATT_CHAR_PROPERTIES_READ);
//Notify or Read
GattCharacteristic *battChar[] = {&battLevel};
GattService      battService ( GattService::UUID_BATTERY_SERVICE, battChar,
    sizeof(battChar)/sizeof(GattCharacteristic *));

/* Advertising data and parameters */
GapAdvertisingData advData;
GapAdvertisingData scanResponse;
GapAdvertisingParams advParams (
    GapAdvertisingParams::ADV_CONNECTABLE_UNDIRECTED );
uint16_t      uuid16_list[] = {GattService::UUID_HEALTH_THERMOMETER_SERVICE,
    GattService::UUID_BATTERY_SERVICE};
    
```

### + Cập nhật các giá trị từ thiết bị lên Central

Trong ví dụ này sử dụng ticker có chu kỳ 1 giây để gửi thông tin về giá trị nhiệt độ đo được từ cảm biến và mức pin đến thiết bị Central

```

void periodicCallback() {
    if (ble.getGapState().connected) {
        //blink led
        led_val=!led_val;
        digitalWrite(LED, led_val);

        //Thermometer value
        uint32_t temp_ieee11073 = quick_ieee11073_from_float(read_temp_value());
        memcpy(thermTempPayload+1, &temp_ieee11073, 4);
    }
}
    
```

```
ble.updateCharacteristicValue(thermTemp.getValueAttribute().getHandle(),
thermTempPayload, sizeof(thermTempPayload));

//Batt level value
/* Decrement the battery level. */
batt <=50 ? batt=100 : batt--;
ble.updateCharacteristicValue(battLevel.getValueAttribute().getHandle(),
(uint8_t*)&batt, sizeof(batt));
}
}
```

Các phần còn lại cơ bản giống trong các bài hướng dẫn trước, các bạn có thể dễ dàng tự tìm hiểu.

*Có thể thấy, công nghệ BLE đã và đang được ứng dụng rất rộng rãi trong đời sống hàng ngày. Mong rằng, với bo mạch VBLUno, các bạn sẽ tiếp cận và làm chủ công nghệ BLE một cách dễ dàng và nhanh chóng nhất, tiếp tục tạo ra thật nhiều thiết bị hữu dụng hơn nữa.*