USBD HID Keyboard SetReport

NuMicro® 32 位系列微控制器范例代码介绍

文件信息

代码简述	示范如何使用键盘取得设定报告请求的数据
BSP 版本	NUC230/NUC240 Series BSP CMSIS v3.01.002
开发平台	NuTiny-EVB-NUC240 V1.2

The information described in this document is the exclusive intellectual property of Nuvoton Technology Corporation and shall not be reproduced without permission from Nuvoton.

Nuvoton is providing this document only for reference purposes of NuMicro microcontroller based system design. Nuvoton assumes no responsibility for errors or omissions.

All data and specifications are subject to change without notice.

For additional information or questions, please contact: Nuvoton Technology Corporation.

www.nuvoton.com

1 功能介绍

1.1 简介

本范例实现了一个USB键盘且使用一根GPIO来模拟击键，同时，使用者可以从PC取得Set_Report请求的数据。

1.2 原理

USB键盘是一种HID(人机接口装置)类别装置，其使用控制管线或中断管线与HID类别驱动器沟通，如图 1-1所示。在本范例中，控制管线用来接收及响应USB控制与类别数据的请求，用来接收从主机传来的数据；而中断管线则是用来接收装置传来的异步数据。此外，本范例并没有宣告中断输出端点，因此，输出报告可以在使用Set_Report请求下通过控制管线传送给装置。

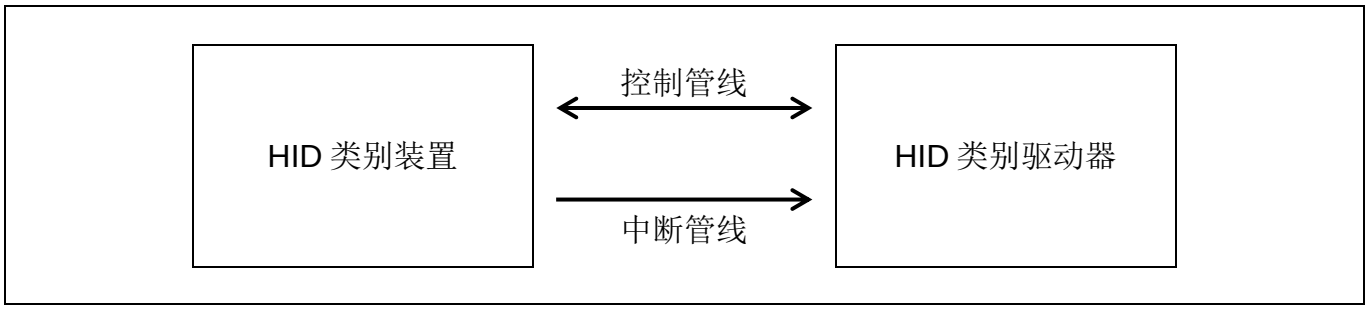

图 1-1 HID 类别沟通图

Set_Report请求是特定类别请求之一，其允许主机传送报告给装置，可能是设定输入、输出或功能控制的状态。这些传输会通过控制管线来完成，因此，它们必需遵照USB规范中定义的控制管线请求格式。Set_Report请求格式栏出如表 1所示。

组成	偏移/大小 (字节)	描述
bmRequestType	0/1	00100001
bRequest	1/1	SET_REPORT (0x09)
wValue	2/2	报告类型与报告标识
wIndex	4/2	界面
wLength	6/2	报告长度

表 1 Set_Report 请求格式

其中，*wValue*字段的高字节指定为报告类型，而报告类型被指定为如表 2所示。因此，在本范例中，当报告类型为0x2时，使用者可以读到从控制输出端点传送来的输出报告。

值	报告类型
0x1	输入

0x2	输出
0x3	功能
0x4-0xFF	保留

表 2 报告类型

1.3 执行结果

以下执行结果会输出于UART鲍率设定为115200的终端机窗口。
一旦GPIO脚位PB15被拉为0时，本范例所实现的USB键盘将会键入Caps Lock。此时，用户可以确认自己键盘上的LED灯是否产生变化。此外，使用者还可以看到其所取得的设定报告请求数据如图 1-2所示。

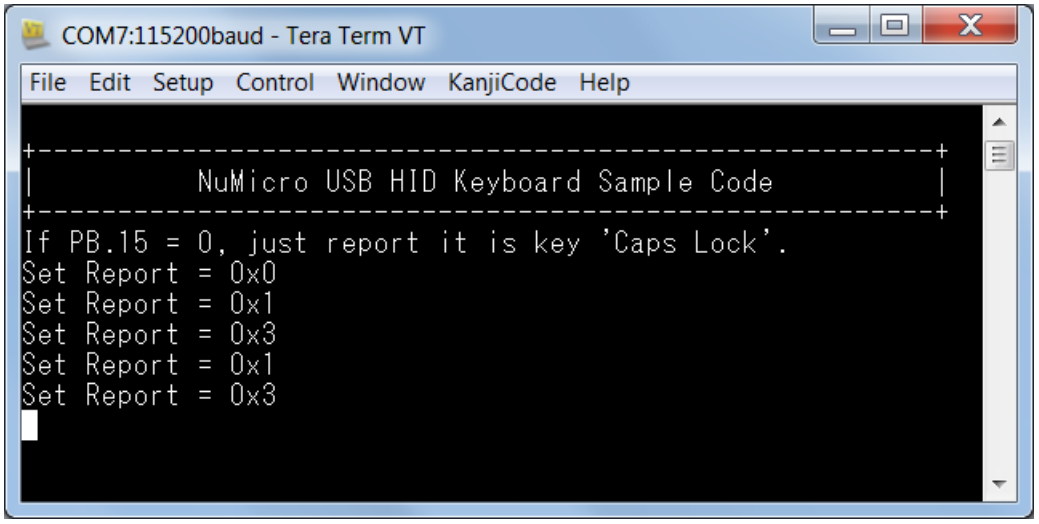

图 1-2 执行结果

2 代码介绍

在主程序中，它会一直轮询 HID_UpdateKbData()函式来仿真 USB 键盘按键的输入；轮询 SetReport_flag 旗标以当 Set_Report 请求发生时将我们所取得的数据讯息显示于 UART：

```
    while (1)
    {
        if (SetReport_flag == 1)
        {
            SetReport_flag = 0;
            printf("Set Report = 0x%X\n\r", RcvBuffer[0]);
        }

        HID_UpdateKbData();
    }
```

当设备接收到 PC 所发出用来求请数据的 IN token 时，端点 2 中断处理程序中的 g_u8EP2Ready 会被设定为 1。如果 g_u8EP2Ready 为 1，HID_UpdateKbData()函式将会执行去设定有效载荷长度及传送数据给 PC。且如果当 PB15 被接至低电位而使 key 为 1 时，buf[2]会被设为 0x39 以模拟键入“Caps Lock”：

```
void HID_UpdateKbData(void)
{
    int32_t i;
    uint8_t *buf;
    uint32_t key = 0xF;
    static uint32_t preKey;

    if (g_u8EP2Ready)
    {
        buf = (uint8_t *)(USBD_BUF_BASE + USBD_GET_EP_BUF_ADDR(EP2));

        /* If GPB15 = 0, just report it is key 'Caps Lock' */
        key = (PB->PIN & (1 << 15)) ? 0 : 1;

        if (key == 0)
        {
            for (i = 0; i < 8; i++)
            {
                buf[i] = 0;
            }
```

```
            if (key != preKey)
            {
                /* Trigger to note key release */
                USBD_SET_PAYLOAD_LEN(EP2, 8);
            }
        }
        else
        {
            preKey = key;
            buf[2] = 0x39; /* Key Caps Lock */
            USBD_SET_PAYLOAD_LEN(EP2, 8);
        }
    }
}
```

在HID_ClassRequest()函式中，当使用者键入Caps Lock时，PC会发出HID特定类别请求中的SET_REPORT请求给装置。此时，设备可以接收到储存在RcvBuffer中击键其新的状态数据，并且设定一个名叫SetReport_flag的旗标为1：

```
void HID_ClassRequest(void)
{
    uint8_t buf[8];

    USBD_GetSetupPacket(buf);

    if (buf[0] & 0x80)   /* request data transfer direction */
    {
        // Device to host
        switch (buf[1])
        {
        ...
        }
    }
    else
    {
        // Host to device
        switch (buf[1])
        {
            case SET_REPORT:
            {
```

```
                if (buf[3] == 2)
                {
                    USBD_PrepareCtrlOut((uint8_t *)&RcvBuffer, 1);

                    /* Request Type = Output */
                    USBD_SET_DATA1(EP1);
                    USBD_SET_PAYLOAD_LEN(EP1, buf[6]);

                    /* Status stage */
                    USBD_PrepareCtrlIn(0, 0);

                    SetReport_flag = 1;
                }
                break;
            }

            ...

            case SET_PROTOCOL:
            {
                USBD_SET_DATA1(EP0);
                USBD_SET_PAYLOAD_LEN(EP0, 0);
                break;
            }
            default:
            {
                // Stall
                /* Setup error, stall the device */
                USBD_SetStall(EP0);
                USBD_SetStall(EP1);
                break;
            }
        }
    }
}
```

3 软件与硬件环境

- 软件环境
  - BSP 版本
    - NUC230/NUC240 Series BSP CMSIS v3.01.002
  - IDE 版本
    - Keil uVision V5.26
- 硬件环境
  - 电路组件
    - NuTiny-EVB-NUC240
  - 示意图
    - 如图 3-1 所示，将 NUC240 USBD 藉由 mini USB 传输线连接至 PC；分别将 NUC240 的引脚：UART0_RX (PB.0)、UART0_TX (PB.1)与 PC 的 UART_TX、UART_RX 进行连接以在 PC 显示执行结果。

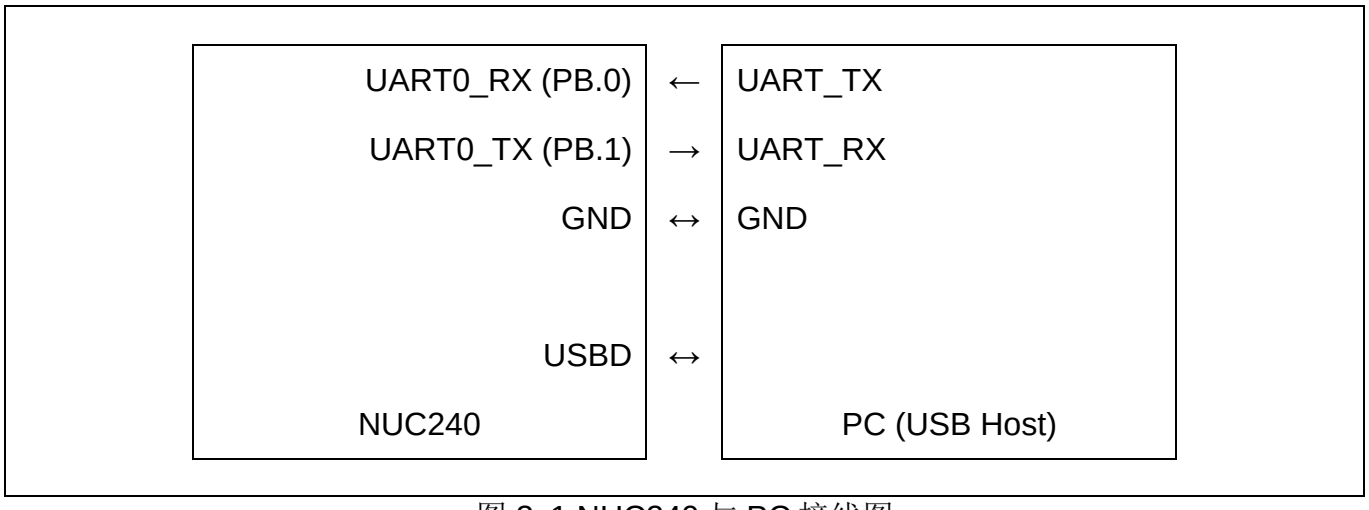

图 3-1 NUC240 与 PC 接线图

4 目录信息

- EC_NUC240_USBD_HID_Keyboard_SetReport_V1.00
  - Library — Sample code header and source files
    - CMSIS — Cortex® Microcontroller Software Interface Standard (CMSIS) by Arm® Corp.
    - Device — CMSIS compliant device header file
    - StdDriver — All peripheral driver header and source files
  - SampleCode
    - ExampleCode — Source file of example code

5 如何执行范例程序

1. 根据目录信息章节进入 ExampleCode 路径中的 KEIL 文件夹，双击 NUC240_USBD_HID_Keyboard_SetReport.uvproj。
2. 进入编译模式接口
   a. 编译
   b. 下载代码至内存
   c. 进入 / 离开除错模式
3. 进入除错模式接口
   a. 执行代码

6 修订纪录

Date	Revision	Description
Aug. 26, 2019	1.00	1. 初始发布.

Important Notice

Nuvoton Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, "Insecure Usage".

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer's risk, and in the event that third parties lay claims to Nuvoton as a result of customer's Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Nuvoton.

Please note that all data and specifications are subject to change without notice.
All the trademarks of products and companies mentioned in this datasheet belong to their respective owners.