

USBFS HID Code Example

v3.0

Features

- HID implementation
- 3-button mouse

General Description

This code example demonstrates USB HID interface class operation by implementing a 3-button mouse. When the code is run, the mouse cursor moves from the right to the left, and vice-versa.

Development Kit Configuration

The example project runs on the CY8CKIT-046 kit from Cypress Semiconductor. A description of the kit, along with more code examples and ordering information, is at <http://www.cypress.com/go/cy8ckit-046>.

The project requires configuration settings changes to run on other kits from Cypress Semiconductor. Table 1 is the list of the supported kits. To switch from CY8CKIT-046 to any other kit, change the project's device with the help of Device Selector called from the project's context menu.

Table 1. Development Kits vs Parts

Development Kit	Device
CY8CKIT-001	CY8C3866AXI-040/ CY8C5868AXI_LP035
CY8CKIT-046	CY8C4248BZI-L489
CY8CKIT-030	CY8C3866AXI-040
CY8CKIT-050	CY8C5868AXI_LP035

The pins assignment for the supported kits is in Table 2.

Table 2. Pins Assignment

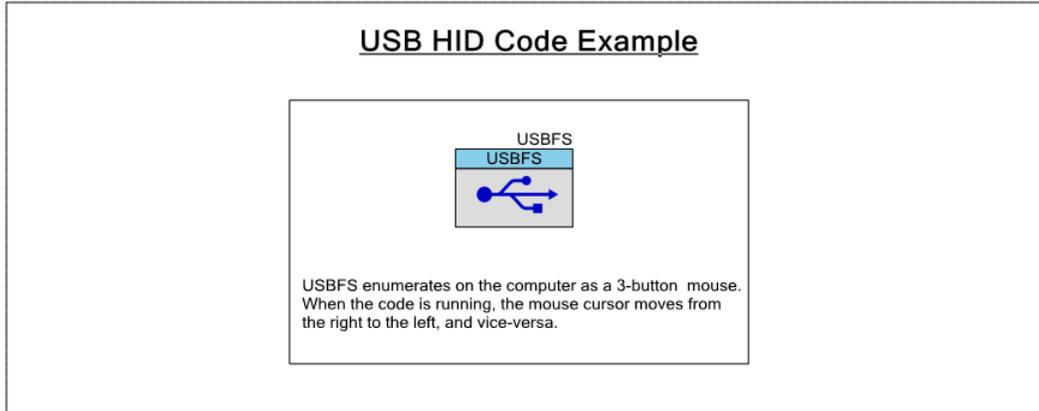
Pin Name	Development Kit			
	CY8CKIT-001	CY8CKIT-046	CY8CKIT-030	CY8CKIT-050
\\USBFS:Dm\\	P15[7]	P13[1]	P15[7]	P15[7]
\\USBFS:Dp\\	P15[6]	P13[0]	P15[6]	P15[6]

Note The project control file handles the pins placement automatically according to a selected PSoC.

Project Configuration

The example project consists of the USBFS component. The project schematic is in [Figure 1](#).

Figure 1. Example Project Design Schematic PSoC 4200



The important USBFS component configuration Tabs are in the figures below.

Figure 2. USBFS Descriptor Root

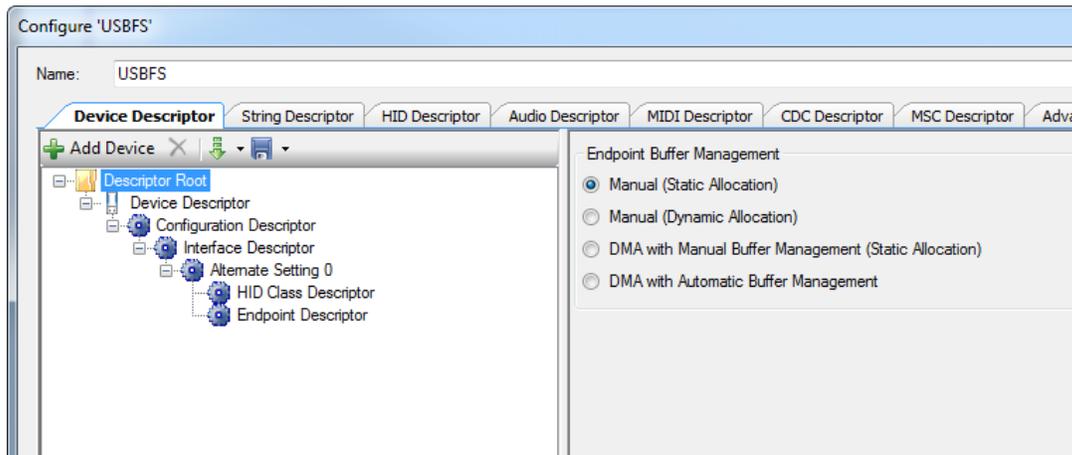


Figure 3. USBFS Device Descriptor

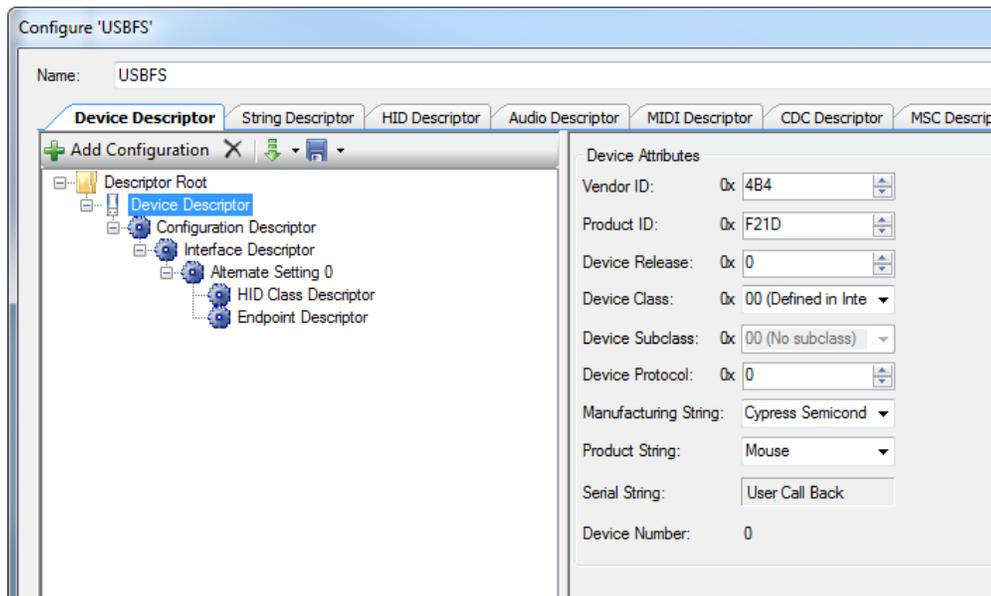


Figure 4. USBFS Configuration Descriptor

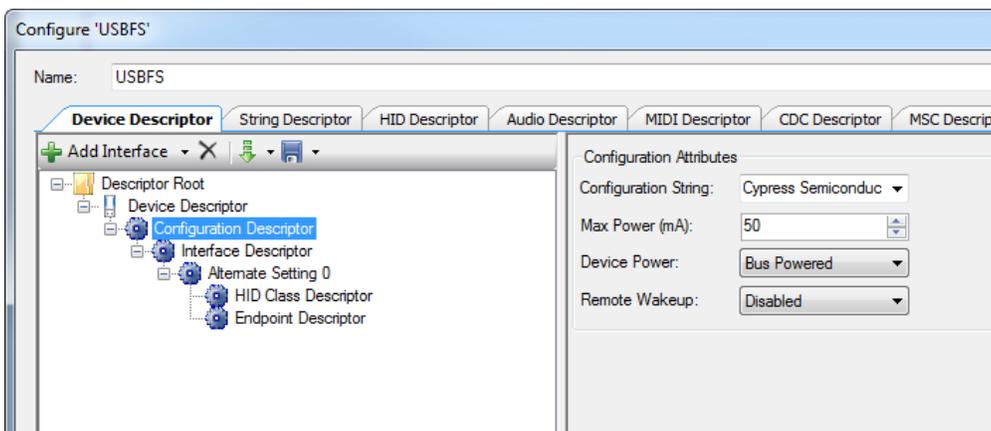


Figure 5. USBFS Interface Descriptor

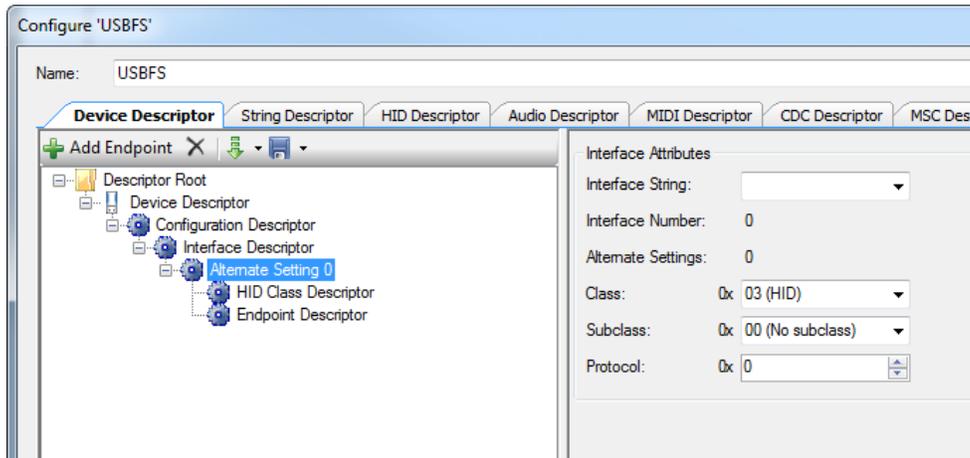


Figure 6. USBFS HID Class Descriptor

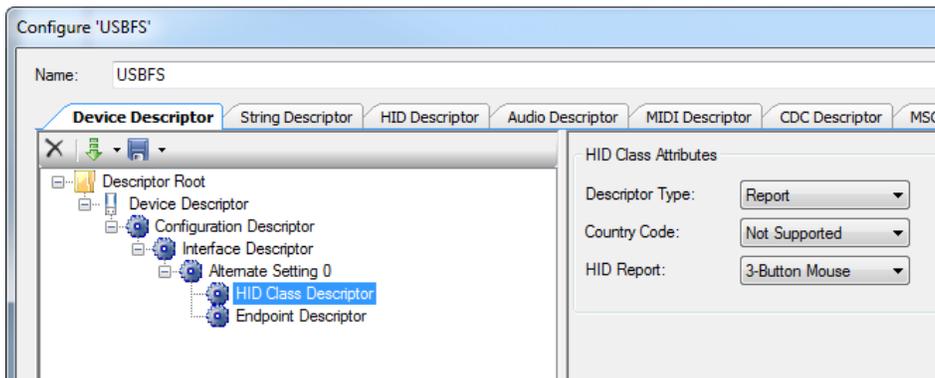
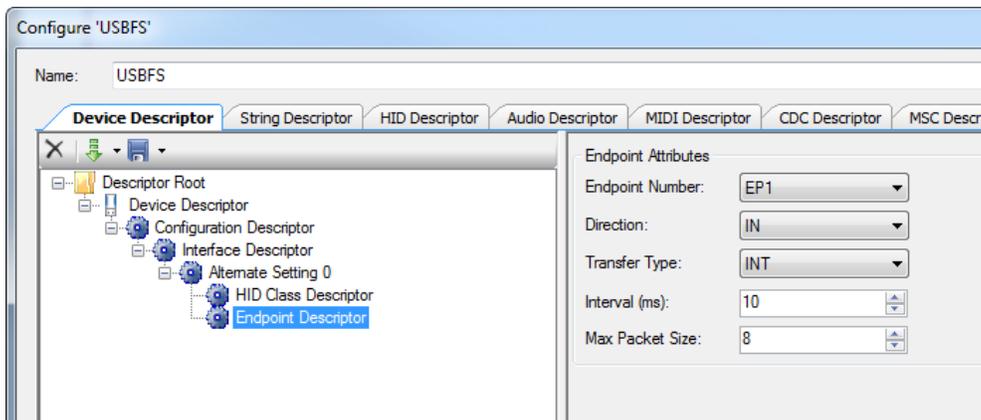


Figure 7. USBFS EP Descriptor



Project Description

The main firmware routine configures the USBFS component for operation and starts it. The code waits for the USBFS device enumeration. The project enumerates in the PC as a 3-button mouse. When the code is running, the mouse cursor moves from the right to the left, and vice-versa.

Example Project Execution Flow

To execute the USBFS component code example you need the following equipment:

- PSoC 3/4200AL/5LP Kit (CY8CKIT-001/030/046/050)

Follow the procedure below:

1. Configure the development kit to operate as bus-powered (the power is supplied from USB connector VBUS pad). This power scheme supposes that the device operates from 5V.

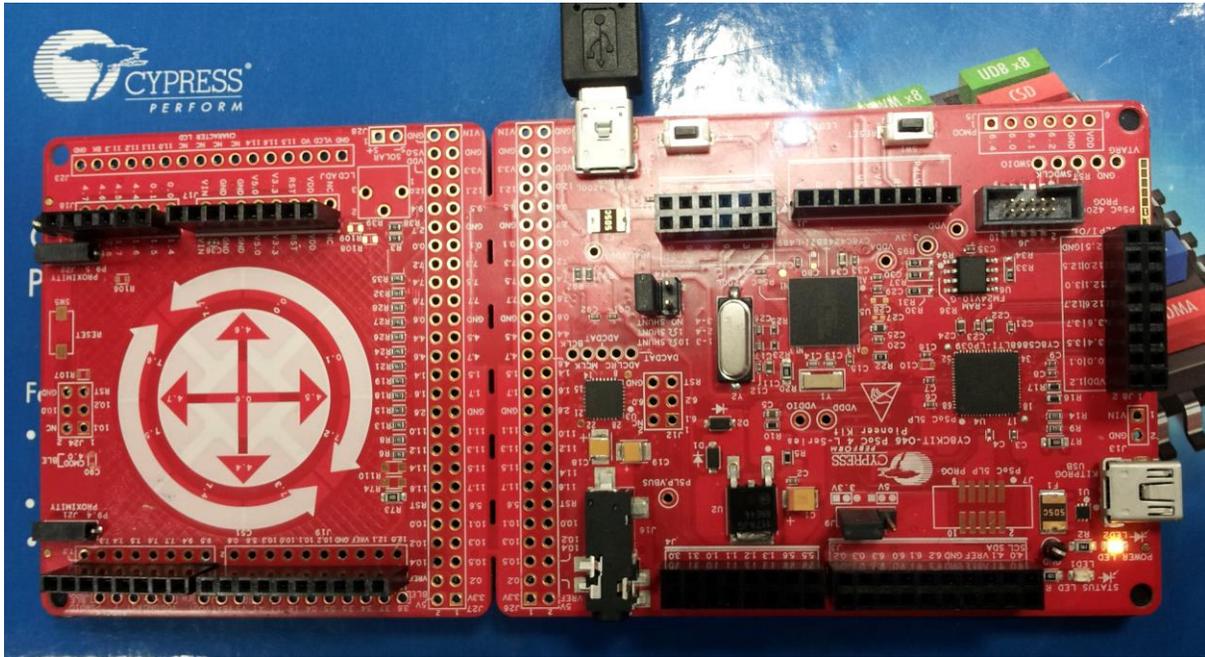
Note The CY8CKIT-046 kit supports only the bus power mode.

2. Build the project and program the hex file into the target device.
3. Connect the computer USB cable to the development kit.
4. Power cycle the device.

Expected Results

You should see that the device is recognized as a 3-button mouse. The mouse is continuously moving back and forth horizontally.

Figure 8. PSoC 4200L Pioneer Kit



© Cypress Semiconductor Corporation, 2015. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.