CE195394 – USB HID Mouse with PSoC 3/ PSoC 5LP

Objective

This code example implements a standard USB HID mouse with a single button. The mouse cursor will move in the shape of a box on the screen.

Overview

This project demonstrates the use of the USBFS Component to implement a HID mouse. Using the standard HID mouse descriptor, the PSoC® device enumerates as a mouse on the PC. When the enumeration is complete, PSoC sends the data about the relative movement of the mouse to the PC. A single button is also implemented in the project to emulate the left button, or button 1, on a standard mouse. You can hold down the button on the kit and watch the cursor highlight text or select items on a desktop while it draws the box.

Requirements

Tool: PSoC Creator™ 4.2

Programming Language: C (Arm® GCC 5.4.1)

Associated Parts: PSoC 3 and PSoC 5LP parts

Related Hardware: CY8CKIT-059, CY8CKIT-001, CY8CKIT-050, CY8CKIT-030

Hardware Setup

Plug in a USB cable to the USB connector on your kit.

This code example is targeted at the CY8CKIT-059 PSoC 5LP Prototyping Kit. If you are using this kit, then no external hardware connections are needed.

If you are not using CY8CKIT-059, you may also need to target a different PSoC device. To do so, right-click the project in **Workspace Explorer** and select **Device Selector**. Select the appropriate PSoC device for your hardware platform.

Software Setup

There is no special software setup to use this project. HID drivers come standard on most operating systems, a benefit of creating a HID device.

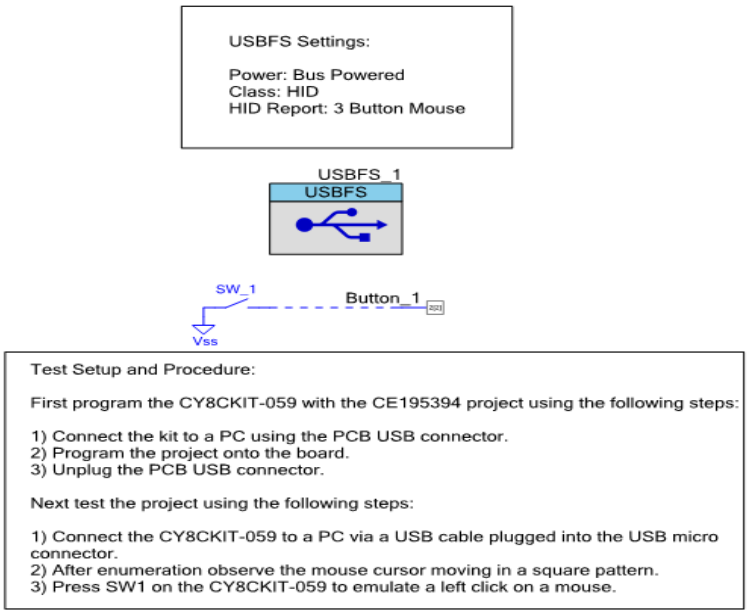
Operation

Program your kit with the code example and plug in a USB cable from your PC to the USB connection on the kit (not the programming connection). Press the associated button to emulate a left click on the mouse.

Design and Implementation

Figure 1 shows the PSoC Creator™ schematic for the code example.

Figure 1. USB HID Mouse Code Example Schematic



The code example uses the USBFS Component to implement the HID mouse and a digital input pin configured in resistive pull-up mode to implement the mouse button.

The firmware for the code example is implemented in *main.c*. The firmware performs the following functions:

- Enumerates the PSoC device as a HID mouse.
- Sends a packet of data with the mouse movement and button status to the PC.
- Waits for the last packet to be acknowledged by the PC.
- Updates the position data and button data after a brief delay.
- Loads the new mouse data into the USB end to be sent to the PC.

The design can be extended to implement a functional mouse by adding real user input to load the X and Y position data with. This can be a joystick or a trackball fed into the ADC.

Components and Settings

Table 1 lists the PSoC Creator Components used in this example, as well as the hardware resources used by each.

Table 1. List of PSoC Creator Components

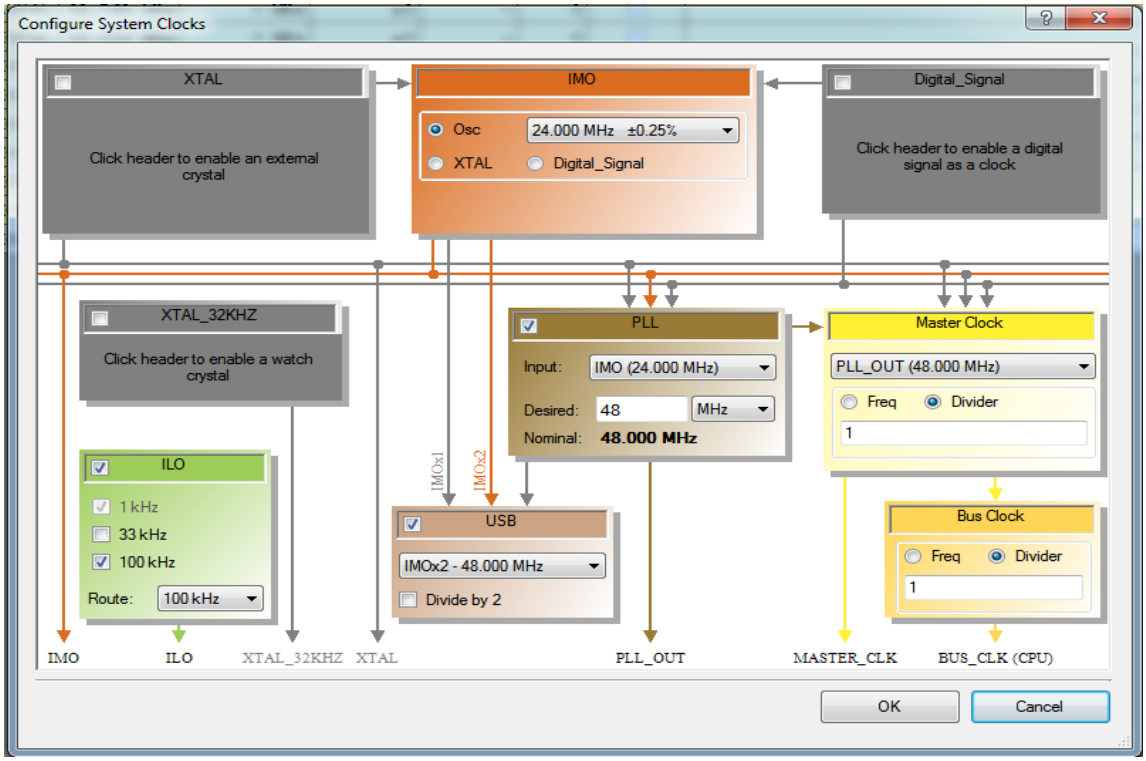
Component	Instance Name	Purpose	Non-default Settings
USBFS	USBFS_1	Implements the 3-Button mouse functionality	Product ID: 0xE011 Manufacturing String: Cypress Semiconductor Product String: Square Mouse Max Power (mA): 20 Class: HID HID Report: 3 Button Mouse Direction: IN Transfer Type: INT Enable SOF interrupt
Digital Input Pin	Button_1	Used to emulate the left click on the mouse	External Terminal Hardware Connection (Unchecked) Initial drive state: High (1)

Table 2 and Figure 2 show the pin selections and the required clock settings for USB operation.

Table 2. Pin Selections

Pin Name	Location
\USBFS_1: Dm\	P15[7]
\USBFS_1: Dp\	P15[6]
Button_1	P2[2]

Figure 2. Clock Settings for USB Operation



Reusing This Example

This code example is designed to run on CY8CKIT-059. To port the design to a different PSoC device and/or kit, change the target device in **Device Selector**, and update the pin assignments in the **Design Wide Resources Pins** settings as needed.

Related Documents

Application Notes	
AN57473 - USB HID Basics with PSoC® 3 and PSoC 5LP	Introduction to USB HID with PSoC
AN58726 - USB HID Intermediate with PSoC® 3 and PSoC 5LP	Intermediate level USB HID with PSoC
AN82072 - PSoC 3 and PSoC 5LP USB General Data Transfer with Standard HID Drivers	General data transfer using HID drivers with PSoC
AN57294 - USB 101: An Introduction to Universal Serial Bus 2.0	Introduction to USB

AN56377 - PSoC 3 and PSoC 5LP - Introduction to Implementing USB Data Transfers	Introduction to USB transfer types
Code Examples	
CE95390	USB Audio with PSoC 3/5LP
CE95393	USB Bulk Transfer with PSoC 3/5LP
CE95395	USB MIDI with PSoC 3/5LP
CE95396	USB UART with PSoC 3/5LP
PSoC Creator Component Datasheets	
USBFS	Details use of the USBFS Component
Device Documentation	
PSoC 3 Datasheets	PSoC 3 Technical Reference Manuals
PSoC 4 Datasheets	PSoC 4 Technical Reference Manuals
PSoC 5LP Datasheets	PSoC 5LP Technical Reference Manuals
Development Kit (DVK) Documentation	
PSoC 3 and PSoC 5LP Kits	
PSoC 4 Kits	

PSoC Resources

Cypress provides a wealth of data at www.cypress.com to help you select the right PSoC device for your design and quickly and effectively integrate it into your design. For a comprehensive list of resources, see KBA86521 – How to Design with PSoC 3, PSoC 4, and PSoC 5LP. The following is an abbreviated list:

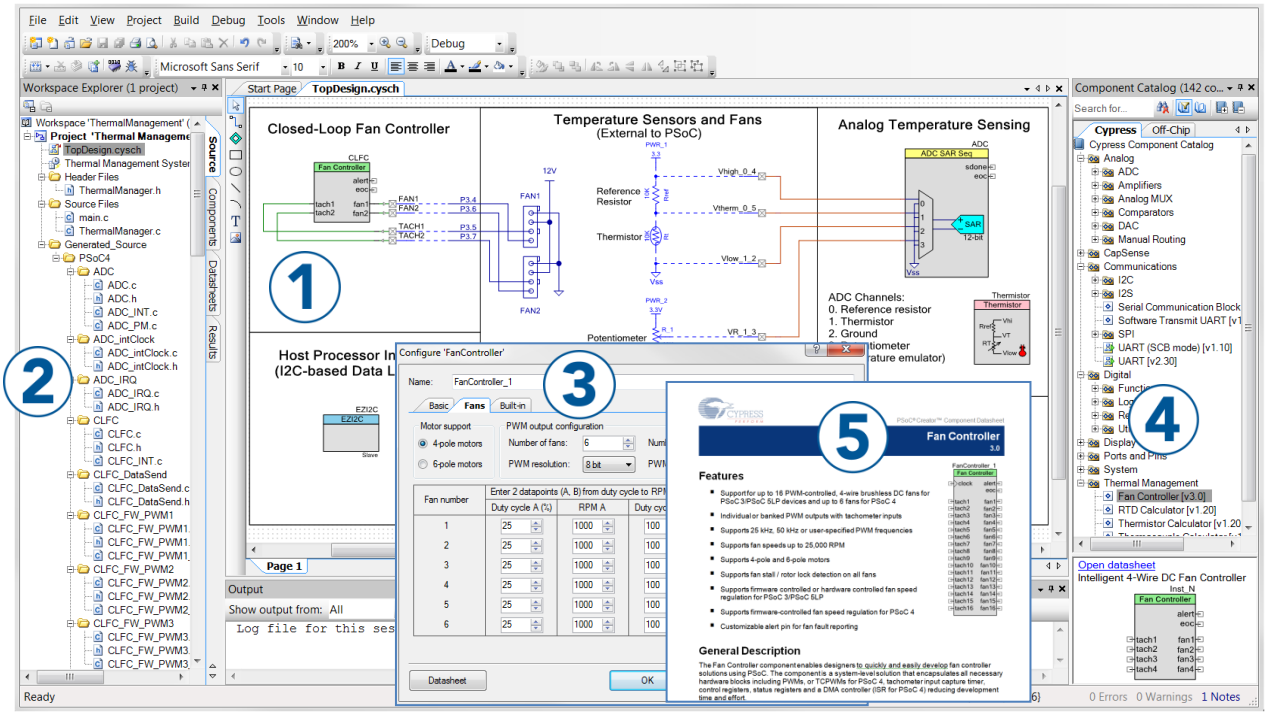
- **Overview:** PSoC Portfolio, PSoC Roadmap
- **Product Selectors:** PSoC 1, PSoC 3, PSoC 4, PSoC 5LP, or PSoC 6. In addition, PSoC Creator includes a device selection tool.
- **Datasheets:** Describe and provide electrical specifications for the PSoC device families.
- **CapSense Design Guides:** Learn how to design capacitive touch-sensing applications.
- **Application Notes:** Cover a broad range of topics, from basic to advanced level.
- **Code Examples**: for PSoC 3, PSoC 4, and PSoC 5LP; or for PSoC 6.
- **PSoC Technical Reference Manuals (TRM):** Provide detailed descriptions of the architecture and registers for PSoC device family.
- **Training Videos:** These videos provide guidance on getting started with various Cypress product families and tools.
- **Development Kits:** Some examples include:
  - PSoC 6 BLE Pioneer Kit is a low-cost hardware platform that enables design and debug of the PSoC 63 series. It comes with an E-Ink display shield board.
  - CY8CKIT-042 and CY8CKIT-040, Pioneer kits, are easy-to-use and inexpensive development platforms. These kits include connectors for Arduino™ compatible shields and Digilent® Pmod™ daughter cards.
  - CY8CKIT-049 is a series of very low-cost prototyping platform for sampling PSoC 4 devices.
  - CY8CKIT-030 and CY8CKIT-050 are designed for analog performance. They enable you to evaluate, develop, and prototype high-precision analog, low-power, and low-voltage applications powered by PSoC 3 and PSoC 5LP, respectively.
  - CY8CKIT-001 is a common development platform for all PSoC family devices.
  - CY8CKIT-059 is a rapid prototyping kit for PSoC 5LP.
- The MiniProg3 device provides an interface for flash programming and debug.

PSoC Creator

PSoC Creator is a free, Windows based Integrated Design Environment (IDE). It enables you to design system hardware and firmware concurrently based on PSoC 3, PSoC 4, PSoC 5LP, and PSoC 6 MCU (see Figure 3). With PSoC Creator, you can do the following:

1. Drag and drop Components to build your hardware system design in the main design workspace.
2. Codesign your application firmware with the PSoC hardware.
3. Configure Components using configuration tools.
4. Explore the library of 100+ Components.
5. Review Component datasheets.

Figure 3. PSoC Creator Features



Document History

Document Title: CE195394 - USB HID Mouse with PSoC 3/PSoC 5LP

Document Number: 001-95394

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	4675951	KLMZ	03/04/2015	New code example
*A	5739970	AESATP12	05/17/2017	Updated logo and copyright.
*B	6012102	SAGA	01/16/2018	Updated template

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Community Forums | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

© Cypress Semiconductor Corporation, 2015-2018. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.