CE220823 – PSoC 6 MCU SMIF Memory Write and Read Operation

Objective

This example demonstrates interfacing with an external NOR flash memory in Quad Serial Peripheral Interface (QSPI) mode using the Serial Memory Interface (SMIF) block in PSoC® 6 MCU.

Requirements

Tool: PSoC Creator™ 4.2; Peripheral Driver Library (PDL) 3.1

Programming Language: C (Arm® GCC 5.4.1 and Arm MDK 5.22)

Associated Parts: All PSoC 6 MCU parts

Related Hardware: PSoC 6 BLE Pioneer Kit, PSoC 6 WiFi-BT Pioneer Kit

Overview

This example writes 64 bytes of data to an external memory in Quad SPI mode. This example also checks the integrity of the read data against written data.

Hardware Setup

This example uses the PSoC 6 WiFi-BT Pioneer kit's default configuration. See the kit guide to ensure the kit is configured correctly.

Software Setup

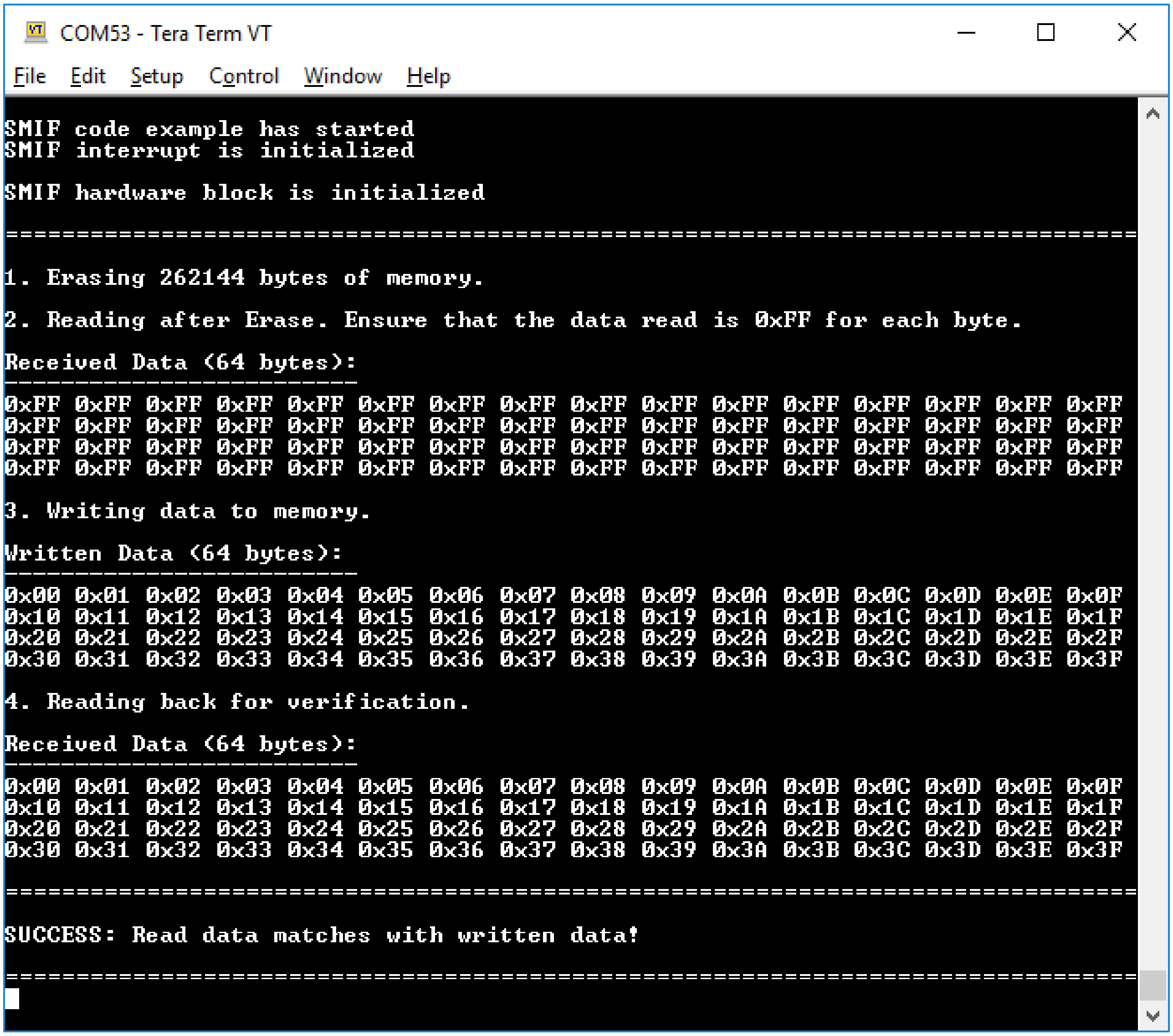
This example uses a serial terminal program. Install one on your PC if you don't have one. The instructions use Tera Term.

Operation

1. Connect the Pioneer board to your PC using the provided USB cable through the USB connector.
2. Open a terminal program and select the KitProg COM port. Set the other serial port parameters as follows:
   - Baud Rate: 115200
   - Data: 8 bits
   - Parity: None
   - Stop: 1 bit
   - Flow Control: None
3. Build the project and program it into the PSoC 6 MCU device. Choose **Debug** > **Program**. For more information on device programming, see PSoC Creator Help. Flash for both CPUs is programmed in a single program operation.
4. Press and release the USER button on the kit when prompted on the terminal window. This is required only the first time the example is run on a kit to enable Quad mode in the memory. Enabling Quad mode is required for communicating in QSPI mode.
5. Observe the KIT_LED1 to determine the status of the SMIF operation.
   - LED is blinking: Successful operation
   - LED is on: Failed operation

   Make sure that debug messages displayed in the terminal window are as expected. Figure 1 is a snapshot of the serial terminal output.

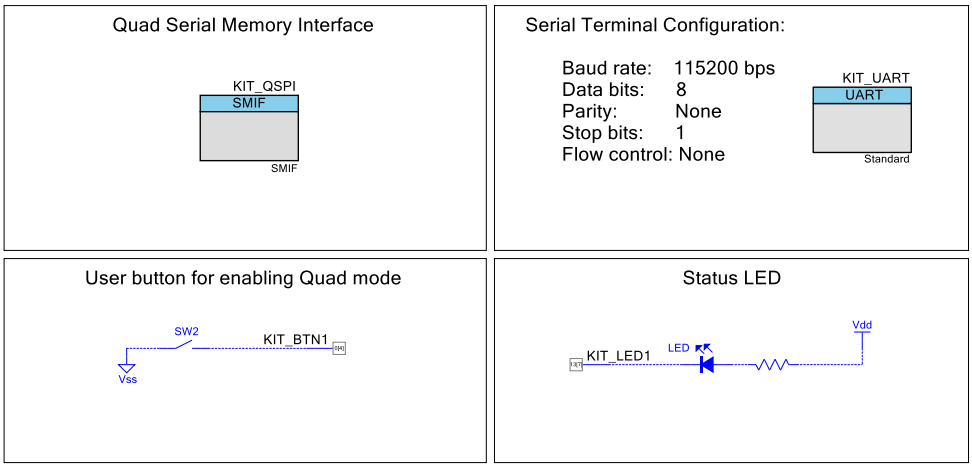
Figure 1. Serial Terminal Output



Design and Implementation

The QSPI resource implements a SPI-based communication for interfacing external memory devices with PSoC. QSPI resource is configured with four data lines and a single slave select line. The UART resource outputs debug information to a terminal window. A user LED indicates the status of read and write operation. Figure 2 shows the design for this code example.

Figure 2. Design Schematic



The firmware uses source code (*cy_smif_memconfig.c* and *cy_smif_memconfig.h* files) generated from the QSPI Configurator. This source code defines data structures that hold the memory configuration.

Components and Settings

Table 1 lists the PSoC Creator Components used in this example, how they are used in the design, and the non-default settings required so they function as intended.

Table 1. PSoC Creator Components

Component	Instance Name	Purpose	Non-default Settings
SMIF	KIT_QSPI	Communicate with QSPI NOR Flash	See Figure 3 and Figure 4
SCB	KIT_UART	UART to transmit debug data	Default
Digital Output Pin	KIT_LED1	LED to indicate SMIF transfer status	See Figure 5
Digital Input Pin	KIT_BTN1	User button for enabling Quad mode	See Figure 6

For information on the hardware resources used by a Component, see the Component datasheet.

Figure 3 through Figure 6 highlight the non-default settings for each component in this example.

Figure 3. QSPI Component Configurator

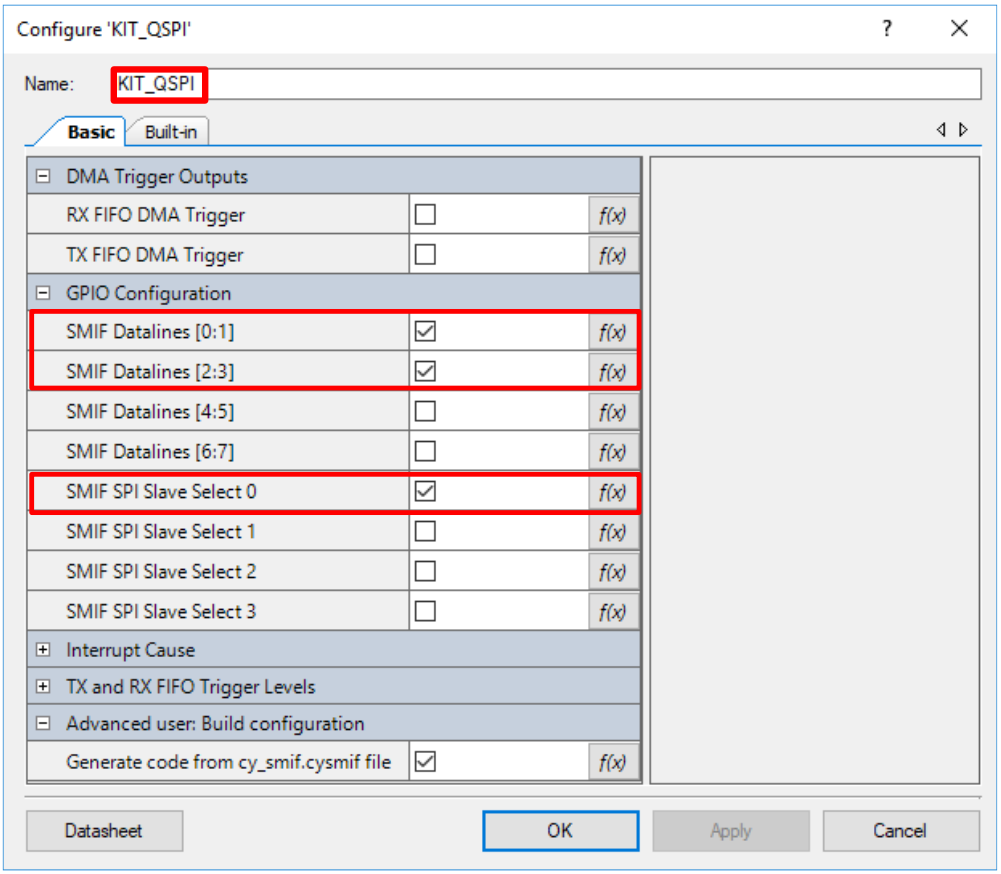


Figure 4. QSPI Configurator Tool

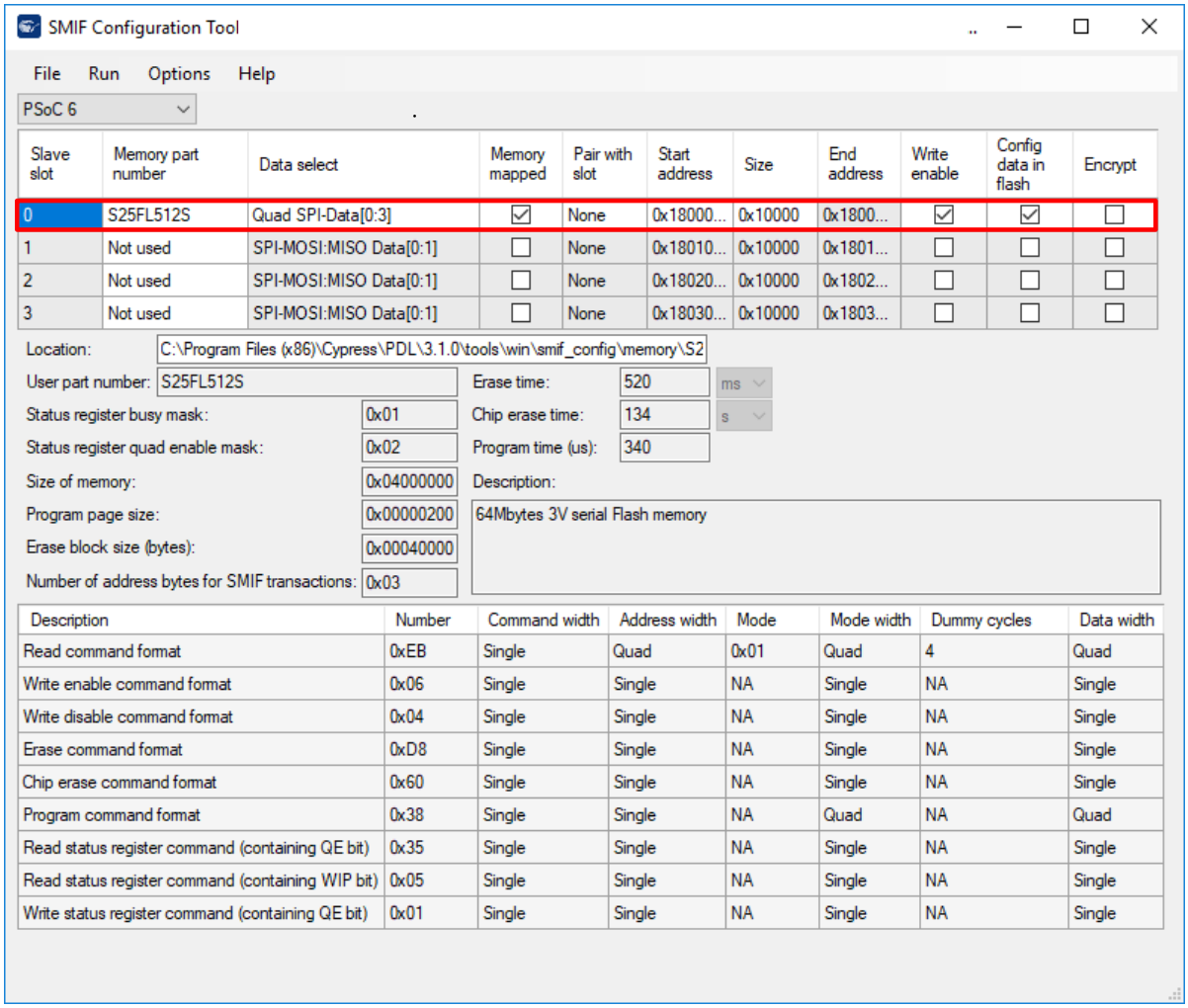


Figure 5. KIT_LED1 Pin Configuration

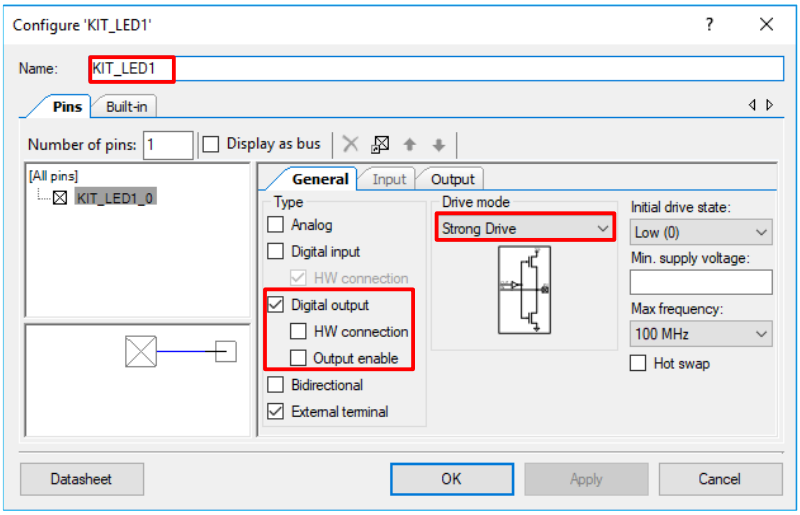


Figure 6. KIT_BTN1 Pin Configuration

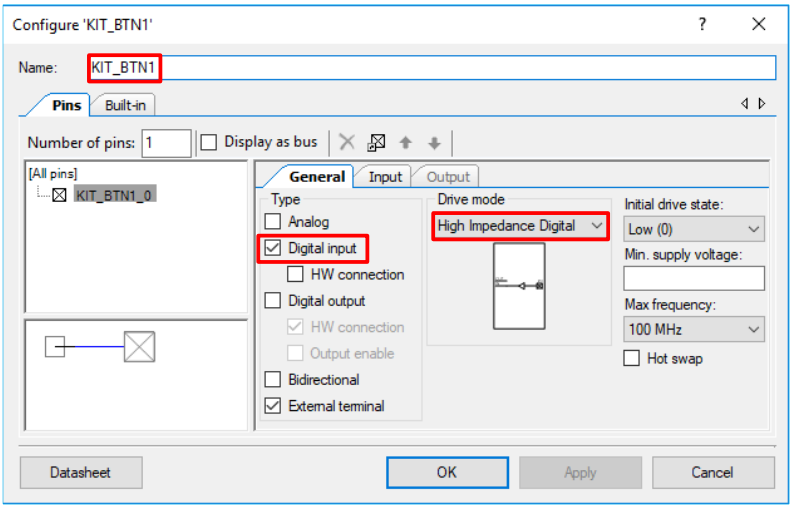


Figure 7 shows pin assignment for the code example.

Figure 7. Pin Assignment

Name	Port
\KIT_QSPI:spi_clk\	P11[7]
\KIT_QSPI:spi_data_0\	P11[6]
\KIT_QSPI:spi_data_1\	P11[5]
\KIT_QSPI:spi_data_2\	P11[4]
\KIT_QSPI:spi_data_3\	P11[3]
\KIT_QSPI:spi_select0\	P11[2]
\KIT_UART:rx\	P5[0]
\KIT_UART:tx\	P5[1]
KIT_BTN1	P0[4]
KIT_LED1	P13[7]

Reusing This Example

This example is designed for the supported kit(s). To port the design to a different PSoC 6 MCU device and/or kit, change the target device using the Device Selector and update the pin assignments in the Design Wide Resources Pins settings as needed. For single-core PSoC 6 MCU devices, port the code from *main_cm4.c* to *main.c*. If you change the device or design, you may need to adjust the source files that come with the example. These are not automatically generated and may not work with a modified design.

In some cases a resource used by a code example (for example, an IP block) is not supported on another device. In that case the example will not work. If you build the code targeted at such a device, you will get errors. See the device datasheet for information on what a particular device supports.

Related Documents

For a comprehensive list of PSoC 6 MCU resources, see KBA223067 in the Cypress community.

For a comprehensive list of PSoC 3, PSoC 4, and PSoC 5LP resources, see KBA86521 in the Cypress community.

Application Notes	
AN210781 – Getting Started with PSoC 6 MCU with Bluetooth Low Energy (BLE) Connectivity	Describes PSoC 6 MCU with BLE Connectivity devices and how to build your first PSoC Creator project
AN215656 – PSoC 6 MCU: Dual-CPU System Design	Describes the dual-CPU architecture in PSoC 6 MCU, and shows how to build a simple dual-CPU design
AN219434 – Importing PSoC Creator Code into an IDE for a PSoC 6 MCU Project	Describes how to import the code generated by PSoC Creator into your preferred IDE
PSoC Creator Component Datasheets	
Pins	Supports connection of hardware resources to physical pins
UART	Supports UART communication
Serial Memory Interface	Supports Single/Dual/Quad/Octal SPI Memories
Device Documentation	
PSoC 6 MCU Datasheets	PSoC 6 Technical Reference Manuals
Development Kit Documentation	
CY8CKIT-062-BLE PSoC 6 BLE Pioneer Kit	
CY8CKIT-062-WiFi-BT PSoC 6 WiFi-BT Pioneer Kit	
CY8CPROTO-062-4343W PSoC 6 Wi-Fi BT Prototyping Kit	
CY8CPROTO-063 BLE PSoC 6 BLE Prototyping Kit	
Tool Documentation	
PSoC Creator	Look in the downloads tab for Quick Start and User Guides
Peripheral Driver Library (PDL)	Get the latest version for use with PSoC Creator. Look in the *<PDL install folder>/doc* for the User Guide and the API Reference

Cypress Resources

Cypress provides a wealth of data at www.cypress.com to help you to select the right device, and quickly and effectively integrate the device into your design.

The following is an abbreviated list of resources related to this code example:

- **Overview:** MCU Portfolio, PSoC & MCU 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 MCU and PSoC device families.
- **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 a PSoC device family.
- **Training Videos**: These videos provide guidance on getting started with various Cypress product families and tools.
- **PSoC 6 MCU Training Videos**: Provide guidance on getting started.
- **CapSense Design Guides:** Learn how to design capacitive touch-sensing applications.
- **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.
  - PSoC 6 WiFi-BT Pioneer Kit supports the PSoC 62 series MCU along with Wi-Fi and BT connectivity
  - 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.
- The MiniProg3 device provides an interface for flash programming and debug.

Document History

Document Title: CE220823 – PSoC 6 MCU SMIF Memory Write and Read Operation

Document Number: 002-20823

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	5856613	VJYA	08/17/2017	Initial public release
*A	5918188	VJYA	11/03/2017	Updated project name
*B	6003180	VJYA	12/22/2017	Updated Figure 6 and Figure 7
*C	6554903	AJYA	04/23/2019	Code example has been revised completely for clarity and correctness

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 | 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