

Please note that Cypress is an Infineon Technologies Company.

The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

Continuity of document content

The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

Continuity of ordering part numbers

Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

Objective

The operation of the 8-bit universal asynchronous receiver transmitter (UART) User Module in PSoC®1 is demonstrated in this example.

Overview

A command and some parameters (separated by spaces) are transmitted to PSoC 1 from a PC using a terminal software, through the serial port using the UART protocol. The data is received and decoded by PSoC 1 and the command and parameters are echoed back to the PC (line by line).

Requirements

Tool: PSoC Designer™ 5.4 CP1

Associated Parts: CY8C29/27/24/22/21xxx, CY8C23x33, CY7C64215, CYWUSB6953, CY8CNP102, CY8CLED02/04/08/16, CY8CLED03D/04D, CY8CTST110, CY8CTMG110, CY8CTST120 CY8CTMG120, CY8CTMA120, CY8C21x45, CY8C22x45, CY8CTMG300, CY8CTST300, CY8CTMA300 CY8CTMA301, CY8CTMA301D, CY8C28x45, CY8CPLC20, CY8CLED16P01, CY8C28x43, and CY8C28x52

Related Hardware: [CY3210 PSoCEval1 kit](#)

Design

This example demonstrates a UART-based serial communication interface between the PSoC 1 device on the CY3210 PSoCEval1 kit and a PC terminal. The terminal receives and displays UART messages and data from PSoC and is used to enter data from the user using the PC. The data entered from the terminal is received by PSoC and parsed in the format "<command> <parameter1> <parameter2>..." and transmitted back to the terminal. See the [Operation](#) section for an example.

The UART User Module enables serial communication in PSoC 1. The 24-MHz system clock is divided by 156 (VC3) to produce a 153.8-kHz clock, which is provided to the UART User Module. The transfer rate is one-eighth times the clock, that is, 19.2 kbps. Other parameters are set to their most commonly used values; for example, Stop = 1 bit and Parity = none.

The program execution flow chart is as follows.

Figure 1. Program Execution Flow Chart

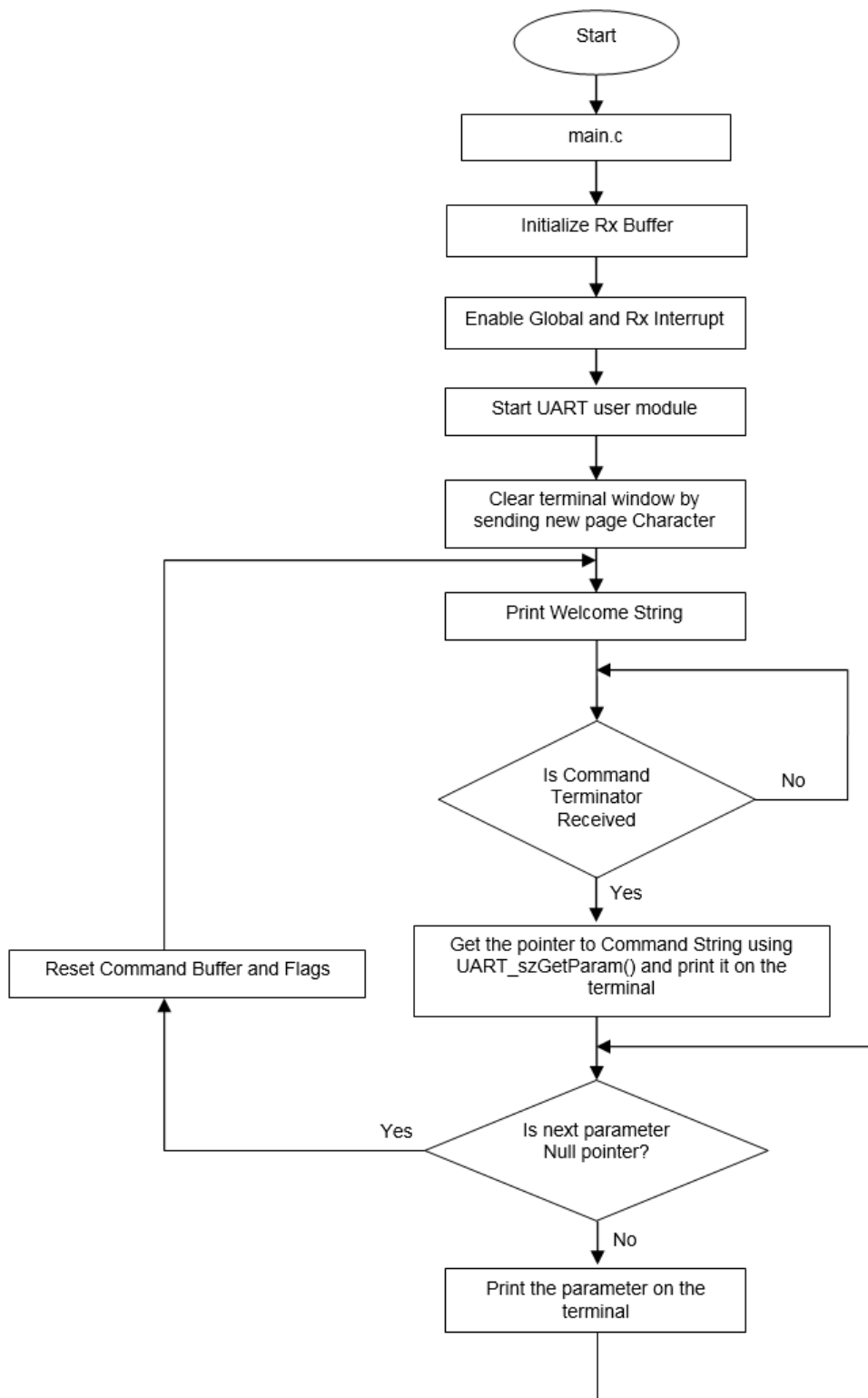
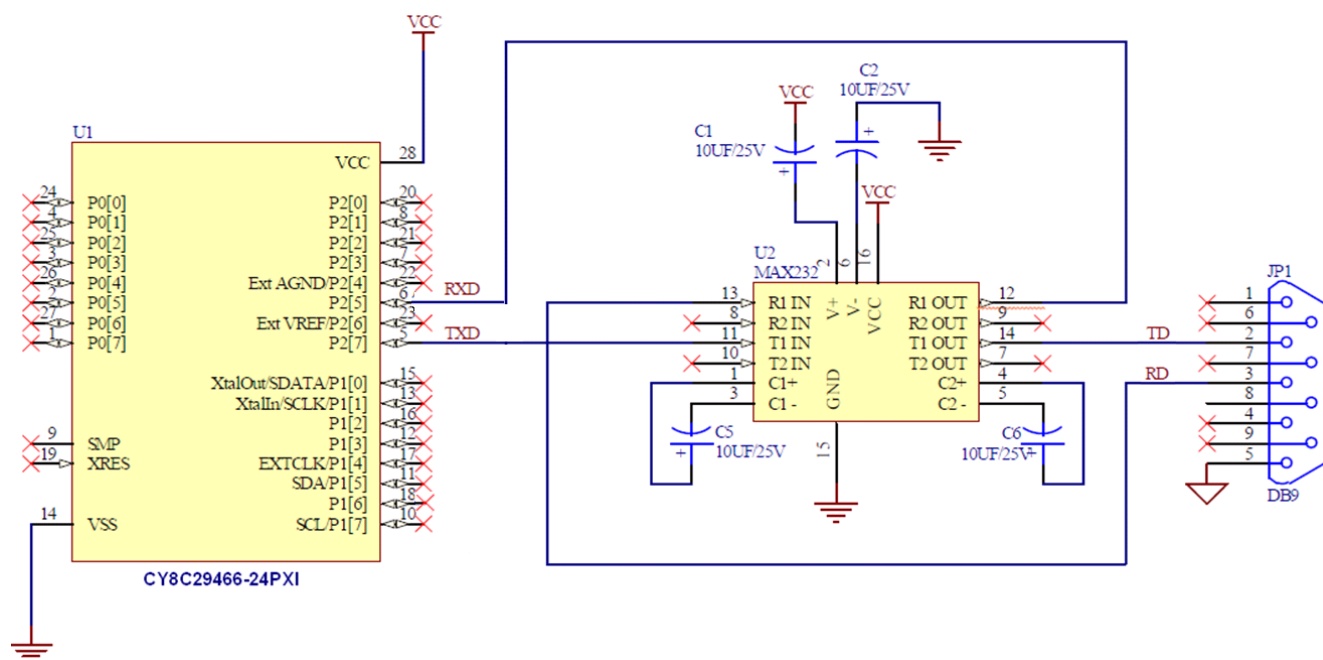


Figure 2. Schematic Diagram



JP1 is a 9-pin female serial port connector used to connect the project with a PC.

- Connect P25 of J7 to RX of J13
- Connect P27 of J7 to TX of J13

Use **TeraTerm** or any other terminal program to test the project. Configure TeraTerm as follows in Windows.

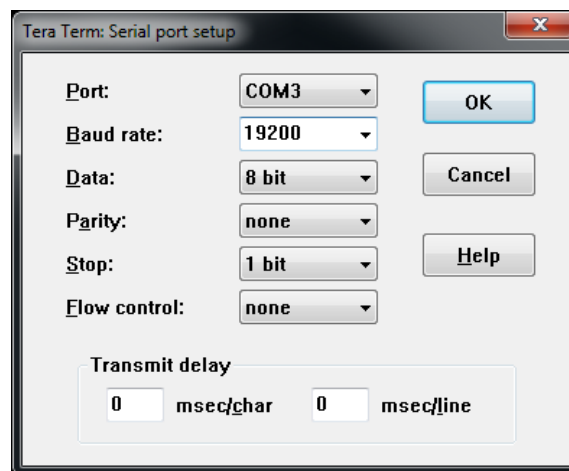
1. Connect the CY3210 board to the PC serial port using a serial port cable.
2. Start **TeraTerm** from the Windows Start menu.

3. Go to **Setup > Serial port** and configure the following parameters:

- Port = select the desired serial port (for example, COM3)
- Baud rate = 19200
- Data = 8 bits
- Parity = None
- Stop = 1 bit
- Flow Control = none

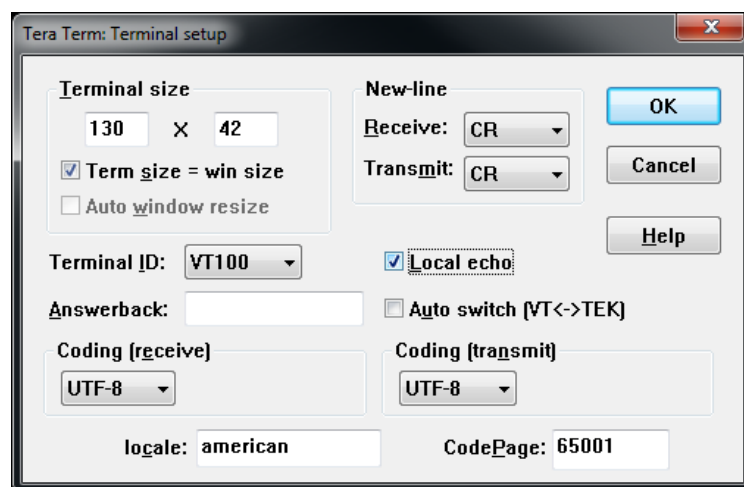
Click **OK** to apply the settings and connect to the COM port.

Figure 3. Serial Port Setup Window

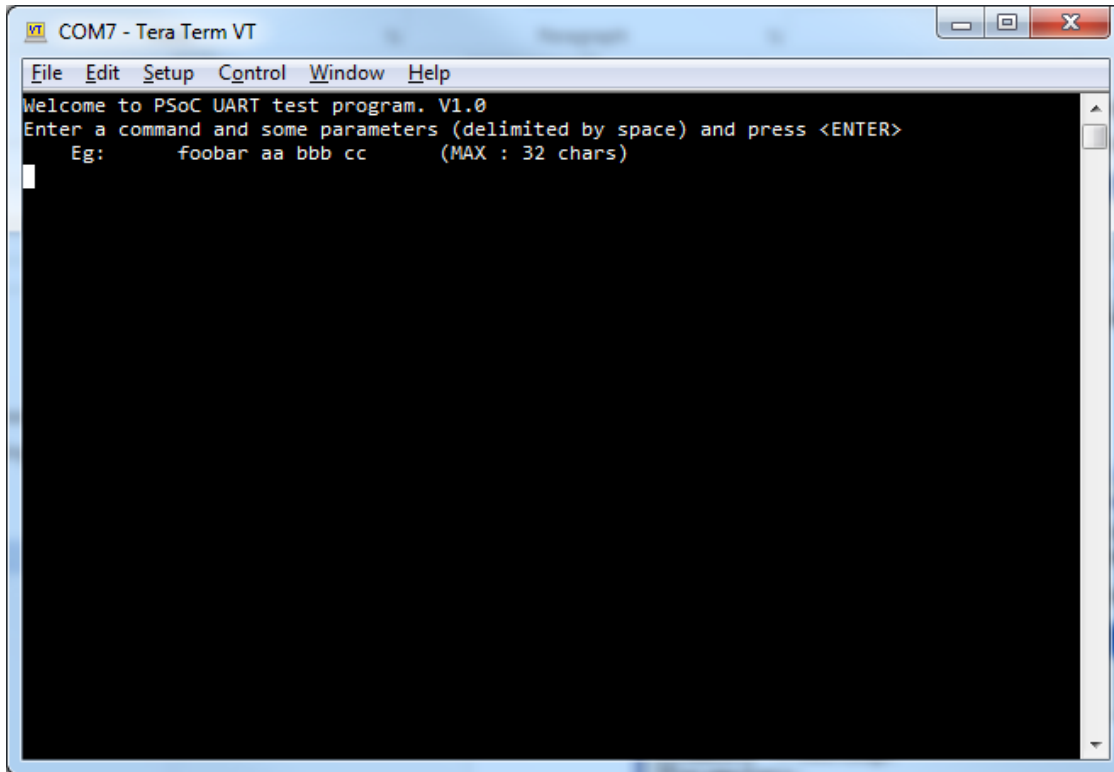


4. Select **Setup > Terminal**. Enable the **Local echo** checkbox and click **OK**.

Figure 4. Setting Up ASCII

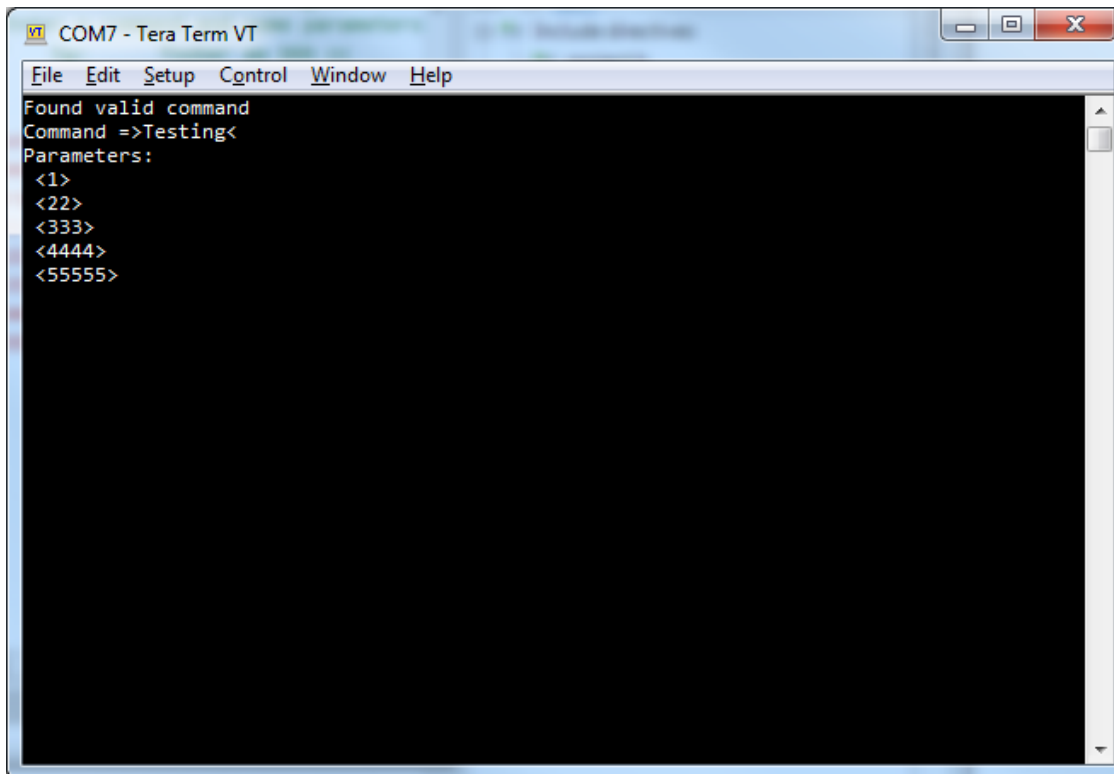


5. Power up the CY3210 board. The terminal displays the welcome screen. Type the command followed by parameters in the terminal. PSoC decodes the command and echoes back to the terminal.



```
COM7 - Tera Term VT
File Edit Setup Control Window Help
Welcome to PSoC UART test program. V1.0
Enter a command and some parameters (delimited by space) and press <ENTER>
Eg:      foobar aa bbb cc      (MAX : 32 chars)

```



```
COM7 - Tera Term VT
File Edit Setup Control Window Help
Found valid command
Command =>Testing<
Parameters:
<1>
<22>
<333>
<4444>
<55555>

```

The sections that follow discuss the user modules and parameter settings used to make the example.

User Modules

Table 1 lists the user modules used in this example, as well as the placement used by each.

Table 1. PSoC Creator Components

| User Module | Placement |
|-------------|--|
| UART | DCB02 (TX Configuration) DCB03 (RX Configuration) |

Parameter Settings

The following table lists the UART User Module parameter settings.

| UART | | |
|--------------------|-------------------|--|
| Parameter | Value | Comments |
| Clock | VC3 | Input is a 153.8-kHz clock (eight times baud rate - 19200). VC3 = SysClk / 156 |
| RX Input | Row_0_Input_1 | Routed from pin P2[5] through GlobalInEven_5 |
| TX Output | Row_0_Output_3 | Routed to pin P2[7] through GlobalOutEven_7 |
| TX InterruptMode | TXRegEmpty | Not used |
| Clock Sync | Sync to Sys.Clock | Clock is synchronized with the SysClk, as VC3 is a derivative of SysClk |
| RxCmdBuffer | Enable | Enables the command buffer and stores the received command in a RAM buffer |
| RxBufferSize | 32 | Byte length of buffer is 32 characters (including parameters) |
| Command Terminator | 13 | Carriage return (13) is the command terminator |
| Param_Delimiter | 32 | Space (32) is the parameter delimiter |
| IgnoreCharsBelow | 32 | Ignore control characters that have an ASCII value below 32 |
| Enable Backspace | Disable | Not used |
| Rx Output | None | Not used |
| Rx Clock Out | None | Not used |
| Tx Clock Out | None | Not used |
| InvertRX Input | Normal | Do not invert Rx input |

Note The clock to the TX8 User Module should be eight times the desired baud rate. The Command Terminator, ParamDelimiter, and IgnoreCharsBelow parameters are used by the high-level UART API to process the commands received. When the UART receives the Command Terminator character set, the API reports through a flag that a valid command has been received. When the ParamDelimiter character set is encountered, the API understands that the characters that follow ParamDelimiter are the next parameters in the command. The API ignores any characters below the value set in IgnoreCharsBelow and does not store them in the command buffer.

Global Resources

| Important Global Resources | | |
|----------------------------|----------|--|
| Parameter | Value | Comments |
| CPU_Clock | SysClk/2 | Sets the CPU frequency to 12 MHz |
| VC3 Source | SysClk/1 | Set System Clock as the source for VC3 |
| VC3 Divider | 156 | Divide the 24-MHz system clock by 156 (effective baud rate is 19.2 kbps) |

Note Leave all other global resources at their default.

Pin Configuration

| Pin | Select | Drive | Interrupt |
|--------|-----------------|--------|------------|
| P2 [7] | GlobalOutEven_7 | Strong | DisableInt |
| P2 [5] | GlobalInEven_5 | High Z | DisableInt |

Related Documents

Table 2. Related Documents

| Application Notes | |
|---|---|
| AN75320 Getting Started with PSoC 1 | Describes the PSoC 1 architecture and development tools. |
| PSoC Designer User Module Datasheets | |
| UART | 8-bit Universal Asynchronous Receiver Transmitter that supports duplex RS-232-compliant, data format serial communications over two wires |
| Device Documentation | |
| PSoC 1 Datasheets | PSoC 1 Technical Reference Manuals |
| Development Kit (DVK) Documentation | |
| PSoC 1 Kits | |

Document History

Document Title: CE54939 - 8-bit Universal Asynchronous Receiver Transmitter

Document Number: 001-54939

| Revision | ECN | Orig. of Change | Submission Date | Description of Change |
|----------|---------|-----------------|-----------------|--|
| ** | 2746859 | DIMA | 07/30/2009 | New example project. |
| *A | 3002139 | DIMA | 08/05/2010 | 1. Updated Software version to PSoC Designer 5.1. 2. Included Related hardware. 3. Included Author name. 4. Included Pin configuration table in page 2. |
| *B | 3250854 | DIMA | 05/06/2011 | Added CE54939 to the header, made minor modifications to the operations section, Updated to PSoC Designer 5.1 FCS. |
| *C | 3340464 | DIMA | 08/09/2011 | Updated to PSoC Designer 5.1 SP2 |
| *D | 3719882 | RICA | 08/22/2012 | Removed Example Projects. |
| *E | 4483393 | ARVI | 08/25/2014 | Updated to PSoC Designer 5.4 CP1 |
| *F | 5879925 | ARVI | 09/11/2017 | Updated template. Updated document with instructions to set up and use the Teraterm software. |

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturers' 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 |

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

PSoC® Solutions

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6](#)

Cypress Developer Community

[Forums](#) | [WICED IOT Forums](#) | [Projects](#) | [Videos](#) | [Blogs](#) | [Training](#) | [Components](#)

Technical Support

cypress.com/support



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

© Cypress Semiconductor Corporation, 2009-2017. 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. 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.