CE224787 – PSoC 4 Lookup Table

Objective

This example shows how to use the Lookup Table (LUT) Component on a PSoC® 4 device.

Requirements

Tool: PSoC Creator 4.2

Programming Language: C (Arm® GCC 5.4.1)

Associated Parts: PSoC 4 devices with universal digital blocks (UDB).

Related Hardware: CY8CKIT-042 PSoC 4 Pioneer Kit

Overview

This example demonstrates how to set up the LUT Component to specify custom combinatorial functions and state machines. Control and status registers are used to show the how the LUT functions. In this example, the LUT is set up to behave like a 3-bit up/down counter. Output values are displayed on the terminal via a UART.

Hardware Setup

This code example is set up for CY8CKIT-042. If you are using a different kit, see

Reusing this Example. In this kit, the USB-UART bridge in the KitProg2 module is used:

1. Connect the \UART:rx\ pin P0[4] to P12[7] on header J8.
2. Connect the \UART:tx\ pin P0[5] to P12[6] on header J8.

Other kits use different pins for the UART. Make sure that you select the correct pins for your kit.

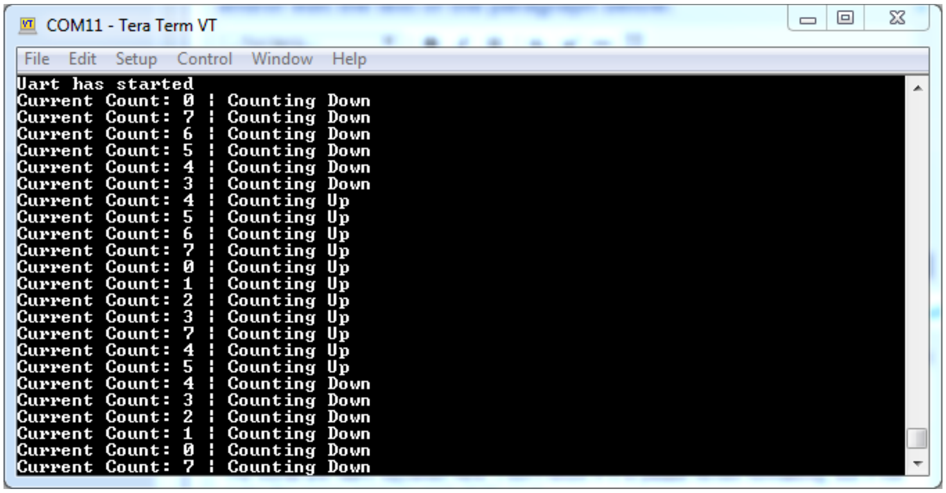
Software Setup

This design requires a terminal emulator such as PuTTY or Tera Term running on your computer.

Operation

1. Ensure that the right pins are connected for your kit, as noted in the Hardware Setup section.
2. Connect the USB cable between the PC and the PSoC 4 Pioneer Kit.
3. Build the project and program it into the PSoC 4 device. Choose **Debug** > **Program**. For more information on device programming, see PSoC Creator Help.
4. Open a terminal emulator on your computer and configure the program to the appropriate COM port. Configure the baud rate to 115200, data bits to 8, no parity bits, stop bit as 1, and no control flow.
5. Each time SW2 is pressed, observe that the mode changes between counting up or down, as Figure 1 shows:

Figure 1. Project Terminal Output



```
Uart has started
Current Count: 0 ¦ Counting Down
Current Count: 7 ¦ Counting Down
Current Count: 6 ¦ Counting Down
Current Count: 5 ¦ Counting Down
Current Count: 4 ¦ Counting Down
Current Count: 3 ¦ Counting Down
Current Count: 4 ¦ Counting Up
Current Count: 5 ¦ Counting Up
Current Count: 6 ¦ Counting Up
Current Count: 7 ¦ Counting Up
Current Count: 0 ¦ Counting Up
Current Count: 1 ¦ Counting Up
Current Count: 2 ¦ Counting Up
Current Count: 3 ¦ Counting Up
Current Count: 7 ¦ Counting Up
Current Count: 4 ¦ Counting Up
Current Count: 5 ¦ Counting Up
Current Count: 4 ¦ Counting Down
Current Count: 3 ¦ Counting Down
Current Count: 2 ¦ Counting Down
Current Count: 1 ¦ Counting Down
Current Count: 0 ¦ Counting Down
Current Count: 7 ¦ Counting Down
```

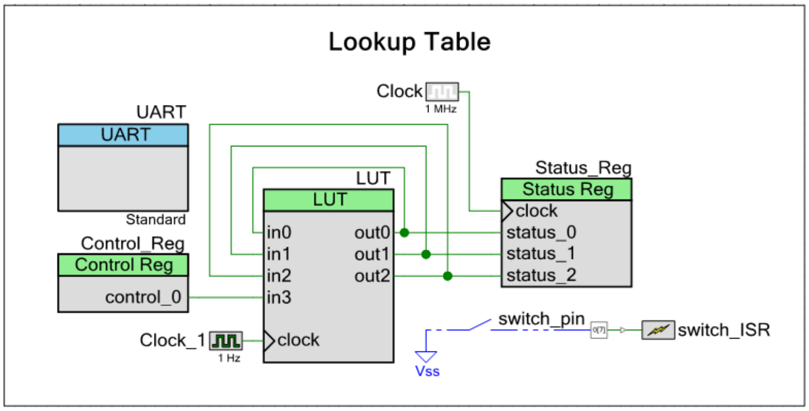
Design and Implementation

In this example, the LUT is used to implement a 3-bit up/down counter, as Figure 2 shows. The LUT is configured with four inputs: in0, in1, and in2 are feedback inputs, in3 controls the counting mode (up or down). The LUT configuration table (Figure 3) implements the bidirectional counter.

The firmware does the following:

1. Sets up UART operation and interrupt handler
2. Updates the control register and LUT output
3. Displays the LUT output and count mode, if there is a change in the LUT output

Figure 2. PSoC Creator Project Schematic



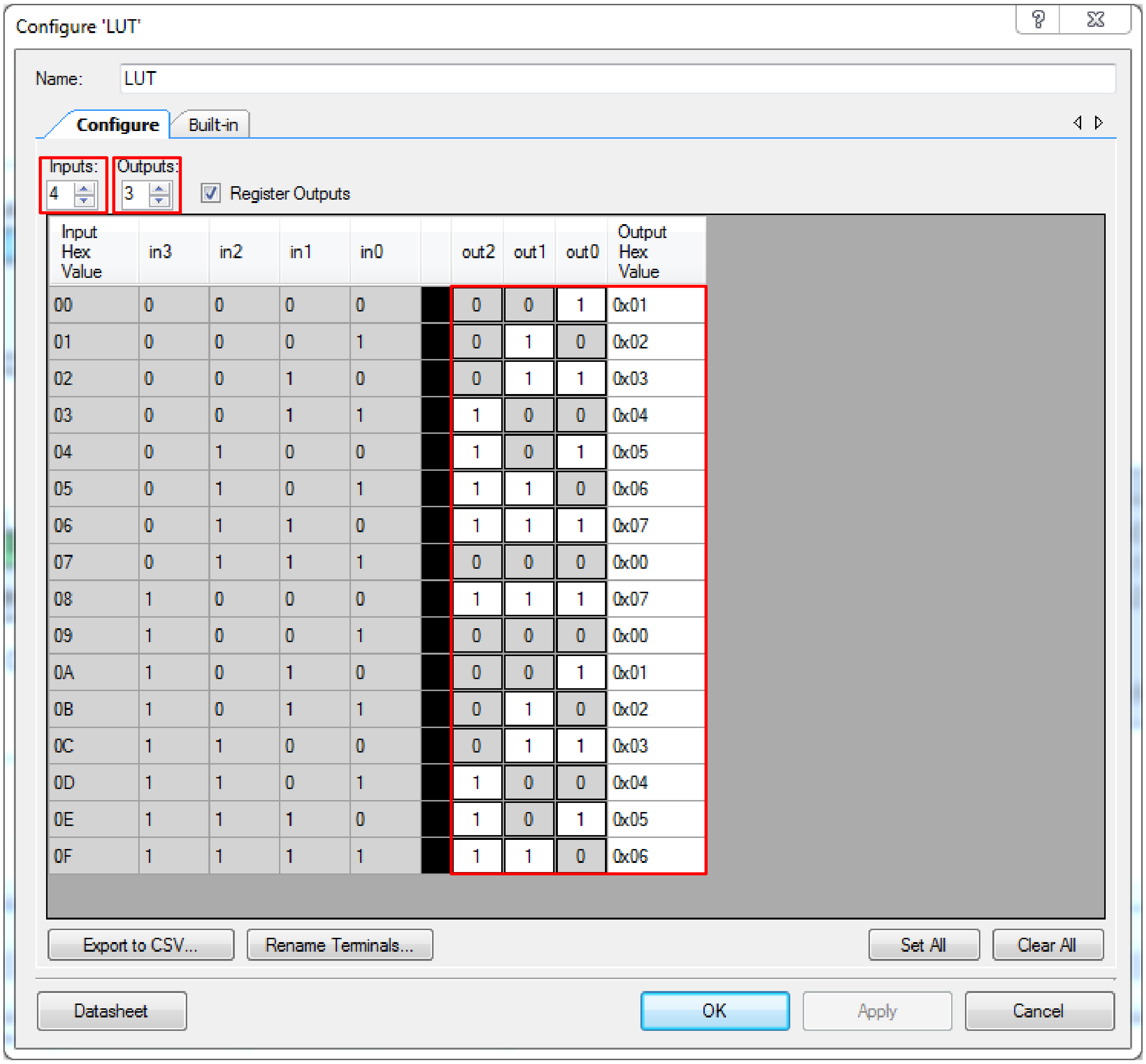
Components and Settings

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
Lookup Table [v1.60]	LUT	Handles Lookup Table operations	See Figure 3. Check the **Register Outputs** box.
Control Register [v1.80]	Control_Reg	CPU writes digital signals	Set **Outputs** to 3
Status Register [v1.90]	Status_Reg	CPU reads digital signals	Set **Inputs** to 3
UART (SCB mode) [v4.0]	UART	Handles UART operation	None

Figure 3. LUT Parameter Settings



Configure 'LUT'

Name: LUT

Inputs: 4

Outputs: 3

☑ Register Outputs

Input Hex Value	in3	in2	in1	in0		out2	out1	out0	Output Hex Value
00	0	0	0	0		0	0	1	0x01
01	0	0	0	1		0	1	0	0x02
02	0	0	1	0		0	1	1	0x03
03	0	0	1	1		1	0	0	0x04
04	0	1	0	0		1	0	1	0x05
05	0	1	0	1		1	1	0	0x06
06	0	1	1	0		1	1	1	0x07
07	0	1	1	1		0	0	0	0x00
08	1	0	0	0		1	1	1	0x07
09	1	0	0	1		0	0	0	0x00
0A	1	0	1	0		0	0	1	0x01
0B	1	0	1	1		0	1	0	0x02
0C	1	1	0	0		0	1	1	0x03
0D	1	1	0	1		1	0	0	0x04
0E	1	1	1	0		1	0	1	0x05
0F	1	1	1	1		1	1	0	0x06

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

Reusing this Example

This example is designed for the CY8CKIT-042 Pioneer kit. To port the design to a different PSoC 4 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. Table 2 shows the pin assignments required for UART operation on other PSoC 4 devices.

Note: This project cannot be built for PSoC 4 devices with no UDBs.

Table 2. Pin Assignments for Different Kits

Pin Name	**Development Kit**			
	CY8CKIT-042	**CY8CKIT-042-BLE**	**CY8CKIT-044**	**CY8CKIT-046**
\UART:rx\	P0[4]	P1[4]	P7[0]	P3[0]
\UART:tx\	P0[5]	P1[5]	P7[1]	P3[1]

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 device supports.

Related Documents

Application Notes	
AN79953 – Getting Started with PSoC® 4	Describes PSoC 4 and shows how to build the attached code example
PSoC Creator Component Datasheets	
Lookup Table	This Component can be set up to perform any logic function with up to five inputs and eight outputs. Outputs can be registered.
SCB	A multifunction hardware block that implements the following communication components: I2C, SPI, UART, and EZI2C.
Control Register	The Control Register allows the firmware to output digital signals.
Status Register	The Status Register allows the firmware to read digital signals.
Device Documentation	
PSoC 4 Datasheets	PSoC 4 Technical Reference Manuals
Development Kit (DVK) Documentation	
CY8CKIT-042 PSoC® 4 Pioneer Kit	
PSoC 4 Kits	
Tool Documentation	
PSoC Creator	Go to the Downloads tab for Quick Start and User Guides

Document History

Document Title: CE224787 – PSoC 4 Lookup Table

Document Number: 002-24787

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	6302259	SYAO	09/14/2018	New code example

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