Nexys4IO Product Guide

Revision 1.1
(Last Updated 30-Dec-2015)

Overview

NOTE: ALL REFERENCES TO NEXYS™4 REFER TO BOTH THE DIGILENT NEXYS4 AND NEXYS4 DDR FPGA DEVELOPMENT BOARDS.

The Digilent Nexys4 board is a complete, ready-to-use digital circuit development platform based on the latest Artix-7™ Field Programmable Gate Array (FPGA) from Xilinx. With its large, high-capacity FPGA (Xilinx part number XC7A100T-1CSG324C), generous external memories, and collection of USB, Ethernet, and other ports, the Nexys4 can host designs ranging from introductory combinational circuits to powerful embedded processors. Several built-in peripherals, including an accelerometer, temperature sensor, MEMs digital microphone, a speaker amplifier, and a lot of I/O devices allow the Nexys4 to be used for a wide range of designs without needing any other components.

The Nexys4 board has an excellent feature set for its price and includes basic I/O devices (switches, buttons, LEDs, seven segment display) that work well for ECE 544 applications. What is lacking, however, is hardware and software to make these devices available to embedded CPUs like the Microblaze. Digilent provides reference designs for this functionality but they are provided as hardware modules (some in VHDL) with no included interface to software applications.

Functional Description

Nexys4IO is a package of custom hardware and driver code that provides a register-based interface to the switches, pushbuttons, LEDs, and seven segment display on the Nexys4. *Nexys4IO* works in conjunction with *Pmod544IO* (another package of custom IP and drivers) to provide a consistent and (hopefully) easy to use interface to the I/O devices used in ECE 544. Providing this IP allows you to focus on your applications - we've done much of the hard work for you. Specifically, the *Nexys4IO* and *Pmod544IO* duo provides the following capabilities:

- Reading the debounced push-buttons and switches, including the push-button and slide switch on the PmodENC.
- Control of the discrete LEDs on the Nexys4 board
- Control of the two RGB LEDs on the Nexys4 board
- Control of the 8 seven segment display digits and decimal points on the Nexys4
- Control and status (including the count) of the PmodENC rotary encoder

- Support for the 2 line x 16 character display on the PmodCLP. The driver provides functions for writing text and numbers to the display, clearing the display, positioning the cursor, and shifting the characters on the display.

As a user of *Nexys4IO* you will be able to read the switches and pushbuttons on the Nexys4 and drive the LEDs, RGB LEDs and seven segment display from a Microblaze using C library calls.

Feature Summary

- AXI Lite slave interface to the basic I/O devices on the Nexys4 providing memory mapped register-based access to a Microblaze or Zynq-based embedded system.
- Packaged as IP for Vivado. Can be added to your design using either the Vivado project manager or the IP Integrator.

Required Hardware

Nexys4IO requires no hardware other than a Nexys4. The port assignments are made at the top level of a hierarchical Vivado RTL project and can be modified by changing the pin mapping in the top level module in the design and in the constraints file. Refer to the Port Description section of this document for the interface signal names and functionality.

The *ECE 544 Getting Started* project includes two files that can be adapted for use on the ECE 544 projects. `n4fpga.v` provides a top level Verilog module that includes an instantiation for an embedded system that includes *Nexys4IO* and *Pmod544IO*. `n4DDRfpga.xdc` is a constraints file that defines the pin mapping between the Nexys4 board and the Artix FPGA on the Nexys4. `n4DDRfpga.xdc` is based on the master `.xdc` file provided by Digilent and should be modified to include or exclude the pins that aren't used in your design. The file n4fpga.xdc provides the pin constraints for the original Nexys4 board

NOTE: DIGILENT PROVIDES DIFFERENT MASTER .XDC FILES FOR THE NEXYS4 DDR AND THE ORIGINAL NEXYS4. BE SURE TO USE THE APPROPRIATE FILE.

To add an instance of *Nexys4IO* to an existing design make the **IP Catalog** tab visible. Use the search box to locate/select *Nexys4IO* and double click on it or right click and select *Add IP*. In the Block Design editor select either individually, or as a group, all of the inputs and outputs of *Nexys4IO* and make them *External*. You may use the Connection Wizard in the Block editor to connect the *Nexys4IO* to the AXI bus

NOTE: IF YOU DO NOT SEE *NEXYS4IO* IN THE IP CATALOG YOU NEED TO MAKE THE IP INTEGRATOR AWARE OF THE LOCATION OF YOUR REPOSITORY. THIS CAN BE DONE BY ADDING THE REPOSITORY LOCATION TO YOUR PROJECT OPTIONS.

Device Drivers

The Nexys4IO hardware is supported by a driver library that provides a (hopefully) easy-to-use application program interface (API). The API is provided as C source code that should be automatically included in the software board support package for a project that includes an instance of Nexys4IO. The drivers are implemented for the Xilinx standalone OS and are not thread-safe (i.e. could cause problems in multitasking applications such as those built with Xilkernel). The API provides the following functionality:

- Base-level functionality for reading the current values of the switches and pushbuttons on the Nexys4. The API also includes functions for detecting whether an individual pushbutton is pressed.
- Base-level functionality LEDs located above the slide switches on the Nexys4.
- Base-level functionality for driving the two RGB LEDs on the Nexys4. Each RGB LED can be driven by three PWM channels, one for the red LED, one for the green LED and one for the blue LED in the RGB LED.
- Base –level functionality for setting the digits and decimal points of the seven segment display on the Nexys4. The seven segment display is broken into two "banks", one for the leftmost 4 digits and the other for the rightmost 4 digits.
- Some intermediate-level functionality for the seven segment display. Specifically, there are API calls for writing binary (hex) and decimal numbers to the display.

The API is documented separately in the *Nexys4IO Device Driver User Guide* and is also available in HTML format in the /doc directory for the Nexys4IO drivers directory and can be accessed by your favorite browser by navigating to the html/index.html. This documentation was generated from the driver source code using doxygen (http://doxygen.org).

Product Specification

Port Descriptions

The Nexys4 I/O peripheral provides an AXI4_Lite slave interface to a Microblaze-based system. It also provides the interface signals between the Artix 7-series FPGA on the Digilent Nexys4 board and the associated devices on the board.

Signal Name	Interface	I/O	Initial State	Description
s_axi_aclk	Clock	I	NA	AXI Clock
s_axi_aresetn	Reset	I	NA	AXI Reset, active-Low
s_axi_*	S_AXI	NA	NA	AXI4-Lite Slave Interface signals. See Appendix A of the AXI Reference Guide (UG761) for AXI4-Lite signals
sw[15:0]	Nexys4IO	I	NA	Slide Switch 15 (SW15) to Slide Switch 0 (SW0) on the Nexys4 board. Nexys4IO debounces the slide switch inputs before making them available to the Microblaze.
led[15:0]	Nexys4IO	O	0000h	LED 15 (LD15) to LED 0 (LD0) on the Nexys4 board
RGB1_Red, RGB1_Green, RGB1_Blue	Nexys4IO	O	0h	RGB1 (LD16) red, green, and blue inputs to the RGB LED
RGB2_Red, RGB2_Green, RGB2_Blue	Nexys4IO	O	0h	RGB2 (LD17) red, green, and blue inputs to the RGB LED
seg[6:0]	Nexys4IO	O	NA	Segment inputs to the 7-segment display digits. The segment inputs to all of digits are time- multiplexed on these 7 signals. Each digit is selected in turn with its associated anode signal.
dp	Nexys4IO	O	NA	Decimal point input to the 7-segment display digits. The decimal points inputs to all of the digits are time-multiplexed on this signal. Each decimal point is selected in turn with its associated anode signal
an[7:0]	Nexys4IO	O	NA	Common anode inputs to each of the 7-segment display digits. These are essentially the digit select for each digit. Each digit is selected, one after the other at a rate fast enough to avoid flickering. The values for each of the segments and decimal point for the selected digit are driven to the display on the seg[6:0] and dp outputs
btnC, btnU, btnL,	Nexys4IO	I	NA	Inputs from the pushbuttons cluster on the

btnR, btnD				Nexys4. btnC is the center button, btnU and btnD are the top and bottom pushbuttons in the cluster and btnL and btnR are the left (West) and Right (East) pushbuttons in the cluster. Nexys4IO debounces the pushbutton inputs before making them available to the Microblaze.

Register Space

The registers for *Nexys4IO* are mapped into the Microblaze address space (memory mapped I/O as is typically the case for RISC CPU's). The base address for the *Nexys4IO* register space is included in `xparameters.h` which is generated when the board support package is built.

Address Offset	Register Name	Description
0h	BTNSW_IN	Debounced pushbuttons and switches for the Nexys 4
04h	LEDS_DATA	LED outputs
08h	RGB1_DATA	RGB LED 1 (LD16 on Nexys4) PWM channel data
0Ch	RGB1_CNTRL	RGB LED 1 PWM channel control
10h	RGB2_DATA	RGB LED 2 (LD17 on Nexys4) PWM channel control
14h	RGB2_CNTRL	RGB LED 2 PWM channel control
18h	SSEGLO_DATA	Seven Segment display digits 3-0 (rightmost 4 digits) data
1Ch	SSEGHI_DATA	Seven Segment display digits 7-4 (leftmost 4 digits) data
20h – 2Fh	RSVD	Reserved

Register Description

Debounced Pushbuttons and Switches Register (BTNSW_IN)

This register can be read by the Microblaze to retrieve the debounced pushbuttons and switches from the Nexys4 board. Both the switches and the pushbuttons return 1 when asserted (switch is on/up, button is pressed). Note: This register is read-only.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved								0	0	0	Buttons					Switches															

Bit	Name	Access Type	Reset Value	Description
31-24	Reserved	N/A	-	Reserved
23-21	Zeros	Read	-	Returns 0s. Buttons are right justified in an 8-bit field
20	BtnCenter	Read	-	Center button in pushbutton cluster
19	BtnUp	Read	-	Up (North) button in pushbutton cluster
18	BtnDown	Read	-	Down (South) button in pushbutton cluster

17	BtnLeft	Read	-	Left (West) button in pushbutton
16	BtnRight	Read	-	Right (East) button in pushbutton cluster
15-0	Switches	Read	-	Slide Switches15 (left) to 0 (right)

LEDs register (LEDS_DATA)

This register can be written by the Microblaze to control the LEDs on the Nexys4 board. Write a 1 to the corresponding bit to turn an LED on. Write a 0 to turn an LED off.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																LED Data															

Bit	Name	Access Type	Reset Value	Description
31-16	Reserved	N/A	-	Reserved
15-0	LED data	Read/Write	0000h	LED 15 (left) to LED 0 (right) control. Returns the current value of the LED data register when read.

RGB LED 1 (LD16 on the Nexys4 board) Channel Data (RGB1_DATA)

The Nexys4 board has 2 RGB LEDS. An RGB LED has separate control for individual Red, Green, and Blue LEDs. Each of the three LEDs is driven with an 8-bit PWM value that is used to control the brightness of the LED giving an almost unlimited set of colors. Returns the current values of the duty cycle registers when read.

NOTE: DIGILENT RECOMMENDS THAT EACH OF THE LEDS BE DRIVEN WITH NO MORE THAN A 50% DUTY CYCLE. THIS IS TAKEN CARE OF IN THE NEXYS4IO RGB PWM LOGIC.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved								Red LED Duty Cycle								Green LED Duty Cycle								Blue LED Duty Cycle							

Bit	Name	Access Type	Reset Value	Description
31-24	Reserved	N/A	-	Reserved
23-16	Red_DC	Read/Write	00h	Red LED duty cycle. A value of 00h turns the Red LED off; a value of FFh turns the Red LED fully on on (see note above).
15-8	Green_DC	Read/Write	00h	Green LED duty cycle. A value of 00h turns the Green LED off; a value of FFh turns the Green LED fully on (see note above).
7-0	Blue_DC	Read/Write	00h	Blue LED duty cycle. A value of 00h turns the Blue LED off; a value of FFh turns the Blue LED fully o on (see note above).

RGB LED 1 (LD16 on the Nexys4 board) Channel Control (RGB1_CNTRL)

The Nexys4 board has 2 RGB LEDS. An RGB LED has separate control for individual Red, Green, and Blue LEDs. Each of the three LEDs is driven with an 8-bit PWM value that is set by writing the RGB0_DATA register. The channel control register can be used to enable/disable each of the 3 LEDs independently. When the channel is disabled channel output is 0 (i.e. that LED is off)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																												0	Channel Enables		

Bit	Name	Access Type	Reset Value	Description
31-4	Reserved	N/A	-	Reserved
3	0	Read	0	Returns 0 on Read. Ignored on write
2	Red_EN	Read/Write	0	Red Channel Enable. Write a 1 to enable the red LED PWM channel; 0 to disable the channel
1	Green_EN	Read/Write	0	Green Channel Enable. Write a 1 to enable the green LED PWM channel; 0 to disable the channel
0	Blue_EN	Read/Write	0	Blue Channel Enable. Write a 1 to enable the blue LED PWM channel; 0 to disable the channel

RGB LED 2 (LD17 on the Nexys4 board) Channel Data (RGB2_DATA)

The Nexys4 board has 2 RGB LEDS. An RGB LED has separate control for individual Red, Green, and Blue LEDs. Each of the three LEDs is driven with an 8-bit PWM value that is used to control the brightness of the LED giving an almost unlimited set of colors. Returns the current values of the duty cycle registers when read.

NOTE: DIGILENT RECOMMENDS THAT EACH OF THE LEDS BE DRIVEN WITH NO MORE THAN A 50% DUTY CYCLE. THIS IS TAKEN CARE OF IN THE NEXYS4IO RGB PWM LOGIC.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved								Red LED Duty Cycle								Green LED Duty Cycle								Blue LED Duty Cycle							

Bit	Name	Access Type	Reset Value	Description
31-24	Reserved	N/A	-	Reserved
23-16	Red_DC	Read/Write	00h	Red LED duty cycle. A value of 00h turns the Red LED off; a value of FFh turns the Red LED fully on on (see note above).
15-8	Green_DC	Read/Write	00h	Green LED duty cycle. A value of 00h turns the Green LED off; a value of FFh turns the Green LED fully on on (see note above).
7-0	Blue_DC	Read/Write	00h	Blue LED duty cycle. A value of 00h turns the Blue LED off; a value of FFh turns the Blue LED fully on on (see note above).

RGB LED 2 (LD17 on the Nexys4 board) Channel Control (RGB2_CNTRL)

The Nexys4 board has 2 RGB LEDS. An RGB LED has separate control for individual Red, Green, and Blue LEDs. Each of the three LEDs is driven with an 8-bit PWM value that is set by writing the RGB0_DATA register. The channel control register can be used to enable/disable each of the 3 LEDs independently. When the channel is disabled the PWM channel output is 0 (i.e. that LED is off)

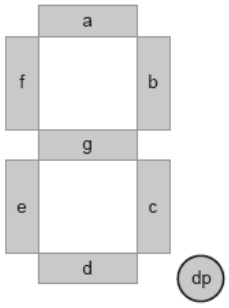
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved																												0	Channel Enables		

Bit	Name	Access Type	Reset Value	Description
31-4	Reserved	N/A	-	Reserved
3	0	Read	0	Returns 0 on Read. Ignored on write
2	Red_EN	Read/Write	0	Red Channel Enable. Write a 1 to enable the red LED PWM channel; 0 to disable the channel
1	Green_EN	Read/Write	0	Green Channel Enable. Write a 1 to enable the green LED PWM channel; 0 to disable the channel
0	Blue_EN	Read/Write	0	Blue Channel Enable. Write a 1 to enable the blue LED PWM channel; 0 to disable the channel

Seven Segment Display Digits 3-0 Data (SSEGLO_DATA)

This Nexys4 board has a total of eight 7-segment display digits organized in two banks of 4. Display digits 3-0 are the in the rightmost bank of digits on the board. Each of the digits can be written with a 5-bit code that selects the character displayed on the screen. The characters are encoded as follows:

Code	Code (hex)	Display Character
0-9	0h-9h	Characters 0 to 9
10 – 15	0Ah-0Fh	Upper case characters A to F
16 – 22	10h – 16h	Single Segments a to g
23	17h	Space (Blank)
24	18h	Upper case character H
25	19h	Upper case character L
26	1Ah	Upper case character R
27	1Bh	Lower case character L (l)
28	1Ch	Lower case character R (r)
29	1D	Lower case character Y (y)
30 – 31	1Eh -1Fh	Space (blank)



It is also possible to light (or not) the decimal points to the right of each of the digits. This is done by writing the Decimal Point (DP) portion of the register. Writing a 1 to a decimal point bit position lights the decimal point; writing a 0 turns the decimal point off.

Since the register is written 32-bits at a time, all four digits and the decimal points will be updated on the write. To ensure that only the digits and/or decimal points that need to be updated are updated it is recommended that you read the register first, change only bits in the returned value that need to be changed, and then write the entire 32-bit register back to the display.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved				Decimal Pts				0	Digit 3 Code					0	Digit 2 Code					0	Digit 1 Code					0	Digit 0 Code					

Bit	Name	Access Type	Reset Value	Description
31-28	Reserved	N/A	-	Reserved
27-24	DecPtsLo	Read/Write	0h	Decimal points for Digit 3 (bit 27) to Digit 0 (bit 24). Digit 3 is the leftmost of the 4 digits in the display bank
23-18	Digit3	Read/Write	1Fh	Display code for Digit 3 (left digit of cluster). Digit is initialized to 1Fh (all segments off). Bit 23 returns 0 on read and is ignored on write
17-12	Digit2	Read/Write	1Fh	Display code for Digit 2. Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write
11-6	Digit1	Read/Write	1Fh	Display code for Digit 1. Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write
5-0	Digit0	Read/Write	1Fh	Display code for Digit 0 (right digit of cluster). Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write

Seven Segment Display Digits 7-4 Data (SSEGHI_DATA)

This Nexys4 board has a total of eight 7-segment display digits organized in two banks of 4. Display digits 7-4 are the in the leftmost bank of digits on the board. Each of the digits can be written with a 5-bit code that selects the character displayed on the screen. See the description for SSEGLO_DATA for the character codes.

It is also possible to light (or not) the decimal points to the right of each of the digits. This is done by writing the Decimal Point (DP) portion of the register. Writing a 1 to a decimal point bit position lights the decimal point; writing a 0 turns the decimal point off.

Since the register is written 32-bits at a time, all four digits and the decimal points will be updated on the write. To ensure that only the digits and/or decimal points that need to be updated are updated it is

recommended that you read the register first, change only bits in the returned value that need to be changed, and then write the entire 32-bit register back to the display.

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Reserved				Decimal Pts				0	Digit 7 Code						0	Digit 6 Code						0	Digit 5 Code						0	Digit 4 Code			

Bit	Name	Access Type	Reset Value	Description
31-28	Reserved	N/A	-	Reserved
27-24	DecPtsHi	Read/Write	0h	Decimal points for Digit 7 (bit 27) to Digit 4 (bit 24). Digit 7 is the leftmost of the 4 digits in the display bank
23-18	Digit7	Read/Write	1Fh	Display code for Digit 7 (left digit of cluster). Digit is initialized to 1Fh (all segments off). Bit 23 returns 0 on read and is ignored on write
17-12	Digit6	Read/Write	1Fh	Display code for Digit 6. Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write
11-6	Digit5	Read/Write	1Fh	Display code for Digit 5. Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write
5-0	Digit4	Read/Write	1Fh	Display code for Digit 4 (right digit of cluster). Digit is initialized to 1Fh (all segments off). Bit 16 returns 0 on read and is ignored on write

Revision History

1.0	05-Jan-2015	Roy Kravitz	First release
1.1	30-Dec-2015	Roy Kravitz	Minor edits, no functionality changes

Related Documents

- *Nexys4IO Device Driver User Guide*; Roy Kravitz; December 2015
- *Digilent Nexys4™ Board Reference Manual,* Copyright Digilent, Inc.