OPB General Purpose Input/Output (GPIO) (v1.00a)

DS466 August 19, 2004

Product Specification

Introduction

This document describes the specifications for the General Purpose Input/Output (GPIO) core for the On-Chip Peripheral Bus (OPB) bus. The GPIO is a 32-bit peripheral that attaches to the OPB.

Features

- OPB v2.0 bus interface with byte-enable support
- Supports 32-bit, 16-bit, and 8-bit bus interfaces
- Each GPIO bit dynamically programmable as input or output
- Number of GPIO bits configurable up to size of data bus interface
- Can be configured as inputs-only to reduce resource utilization

LogiCORE™ Facts		
Core Specifics		
Supported Device Family	QPro™-R Virtex-II™, QPro Virtex-II, Spartan-II™, Spartan-IIE™, Spartan-3™, Virtex™, Virtex-II, Virtex-II Pro™, Virtex-4™, Virtex-E	
Version of Core	opb_gpio	v1.00a
Resources Used		
	Min	Max
Slices	22	104
LUTs	8	49
FFs	31	193
Block RAMs	0	0
Provided with Core		
Documentation	Product Specification	
Design File Formats	VHDL	
Constraints File	N/A	
Verification	N/A	
Instantiation Template	N/A	
Reference Designs	None	
Design Tool Requirements		
Xilinx Implementation Tools	5.1i or later	
Verification	N/A	
Simulation	ModelSim SE/EE 5.6e or later	
Synthesis	XST	
Support		
Support provided by Xilinx, Inc.		

GPIO Organization

The GPIO is a simple peripheral consisting of two registers and a multiplexer for reading register contents and the GPIO I/O signals. The GPIO block diagram is shown in the following figure:

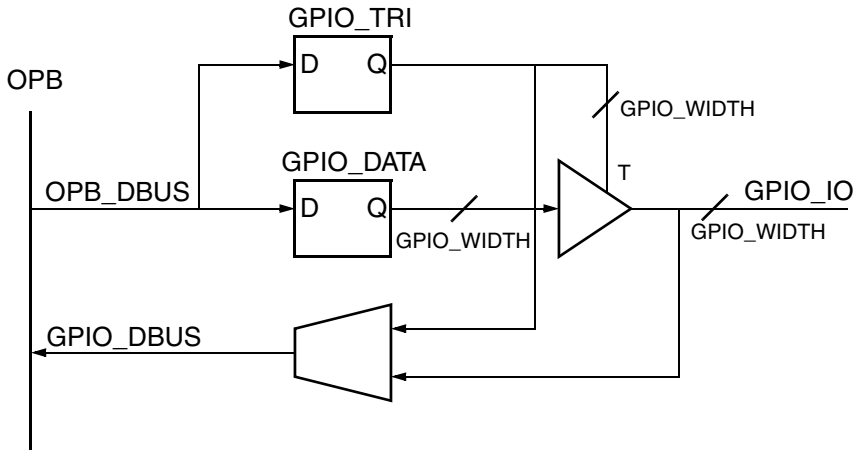

Figure 1: **GPIO Block Diagram**

Programming Model

Register Data Types and Organization

Registers in the GPIO are accessed as one of the following types:

- Byte (8 bits)
- Half word (2 bytes)
- Word (4 bytes)

Configuration

The following table shows GPIO configurations and access type.

Table 1: **GPIO Configuration and Access Type**

Configuration	Access Type
32-bit slave OPB peripheral	Word
16-bit peripheral	Half word
8-bit peripheral	Byte
32-bit, 16-bit, or 8-bit peripheral	All register accesses are on word boundaries to conform to the OPB-IPIF register location convention

The addresses of the GPIO registers when configured as a 32-bit OPB slave are shown in the following table:

Table 2: **GPIO Register Address Map (32-bit OPB)**

Register	Address (Hex)	Size	Type	Description
GPIO_DATA	0x00	Word	Read/Write	GPIO Data Register
GPIO_TRI	0x04	Word	Read/Write	GPIO Three-state Register

The GPIO registers are organized as big-endian data. The bit and byte labeling for the big-endian data types is shown in the following figure:

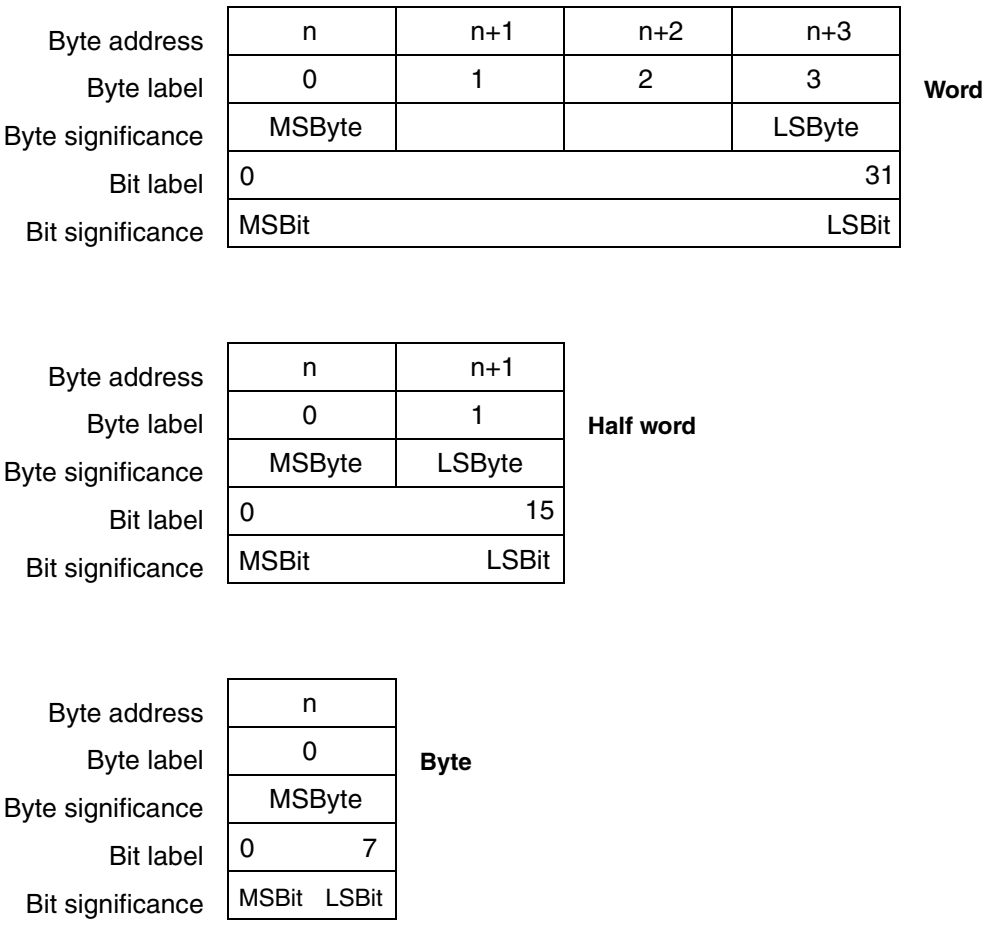

Figure 2: **Big-Endian Data Types**

Registers of the GPIO

Information on the registers used in assembly language programming are described in this section.

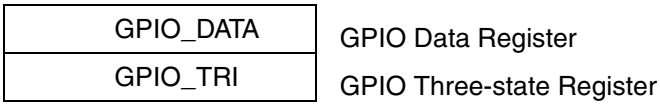

Figure 3: **GPIO Register Set**

Address Map

Table 3: **GPIO Register Address Map (32-bit OPB)**

Register	Address (Hex)	Size	Type	Description
GPIO_DATA	0x00	Word	R/W	GPIO Data Register
GPIO_TRI	0x04	Word	R/W	GPIO Three-state Register

GPIO Data Register (GPIO_DATA)

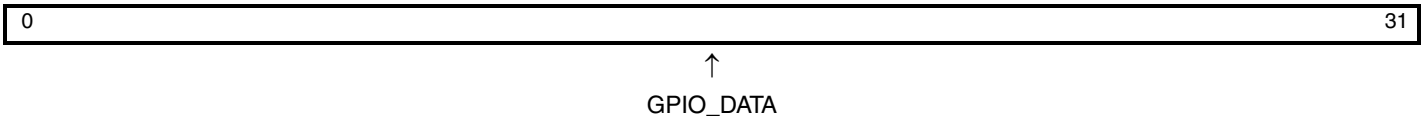

Figure 4: **GPIO_DATA**

Table 4: **GPIO_DATA Register**

Bits	Name	Description	Reset Value
0:31	GPIO_DATA	GPIO Data *For I/O programmed as inputs:* R: reads value on input pin W: no effect *For I/O programmed as outputs:* R: reads value in GPIO data register W: writes value to GPIO data register and output pin	0

GPIO Three-state Register (GPIO_TRI)

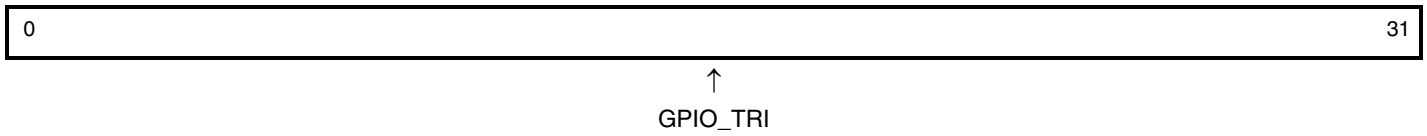

Figure 5: **GPIO_TRI**

Table 5: **GPIO_TRI Register**

Bits	Name	Description	Reset Value
0:31	GPIO_TRI	GPIO Three-state Control (Bit Direction). Each I/O pin of the GPIO is individually programmable as an input or output. For each bit: 0 I/O pin configured as output 1 I/O pin configured as input	all bits = 1

Operation

GPIO Operation

A write to the GPIO_DATA register causes the written data to appear on the GPIO I/O port for I/Os that are configured as outputs. The GPIO_DATA register reads either the content of the GPIO_DATA register (for I/Os configured as outputs), or the GPIO I/O port (for I/Os configured as inputs).

The GPIO_TRI register configures the I/O as either input or output. Each bit of the I/O port has a corresponding bit in the GPIO_TRI register. Each I/O bit can be individually configured as input or output. If only inputs are required, the C_ALL_INPUTS parameter can be set to true. As a result, the GPIO_TRI register and the read multiplexer are removed from the logic to reduce resource utilization.

Implementation

I/O Summary

Table 6: **Summary of GPIO I/O (32b OPB interface)**

Signal	Interface	I/O	Description	Page
OPB_Clk	OPB	I	OPB Clock	
OPB_Rst	OPB	I	OPB Reset	
OPB_ABus[0:31]	OPB	I	OPB Address Bus	
OPB_BE[0:3]	OPB	I	OPB Byte Enables	
OPB_DBus[0:31]	OPB	I	OPB Data Bus	
OPB_RNW	OPB	I	OPB Read, Not Write	

Table 6: **Summary of GPIO I/O (32b OPB interface) *(Continued)***

Signal	Interface	I/O	Description	Page
OPB_select	OPB	I	OPB Select	
OPB_seqAddr	OPB	I	OPB Sequential Address	
GPIO_DBus[0:31]	OPB	O	GPIO Data Bus	
GPIO_errAck	OPB	O	GPIO Error Acknowledge	
GPIO_retry	OPB	O	GPIO Retry	
GPIO_toutSup	OPB	O	GPIO Timeout Suppress	
GPIO_xferAck	OPB	O	GPIO Transfer Acknowledge	
GPIO_IO[0:31]	Ext.	I/O	General Purpose Input/Outputs. Number of I/O bits is configurable at FPGA configuration, direction of each I/O bit is programmable at run-time.	

MPD File Parameters

The opb_gpio.mpd (Microprocessor Peripheral Definition) file contains a list of the peripheral's parameters that are fixed at FPGA configuration time. The parameters are described in the following table:

Table 7: **MPD Parameters**

Parameter	Description	Type
C_OPB_AWIDTH	Width of the address bus attached to the peripheral	integer
C_OPB_DWIDTH	Width of the data bus attached to the peripheral	integer
C_BASEADDR	Indicates the base address of this peripheral expressed as a std_logic_vector	std_logic_vector (0 to C_AWIDTH-1)
C_HIGHADDR	Indicates the highest address occupied by this peripheral expressed as a standard logic vector	std_logic_vector (0 to C_AWIDTH-1)
C_GPIO_WIDTH	Width of the GPIO bus (number of GPIO bits used)	integer
C_ALL_INPUTS	Indicates that all I/O are configured as inputs; results in lower resource utilization if only inputs are needed 0: I/O are programmable as input or output. 1: All I/O are inputs	integer

Parameterization

The following characteristics of the GPIO can be parameterized:

- Base address for the GPIO registers
- Width of OPB data bus attached to the peripheral
- Width of OPB address bus attached to the peripheral
- Number of GPIO bits
- I/Os are input-only or programmable as input or output

Device Utilization

The following table shows approximate resource utilization for the OPB GPIO. The estimates shown are not guaranteed and can vary with FPGA family and speed grade, parameters selected for implementation, user timing constraints, and implementation tool version. Only parameters that affect resource utilization are shown in the following table:

Table 8: **OPB GPIO Resource Utilization (Virtex-II Pro)**

Parameter Values		Device Resources		
C_ALL_INPUTS	C_GPIO_WIDTH	Slices	Slice Flip-Flops	4-input LUTs
0	1	27	35	11
0	8	42	70	18
0	16	62	111	28
0	32	104	193	49
1	1	22	31	8
1	8	28	45	8
1	16	36	62	8
1	32	56	96	9

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
08/24/01	1.0	Initial Xilinx release.
03/20/02	2.0	Updated for MDK 2.2
05/22/02	2.1	Update for EDK 1.0
07/23/02	2.2	Add XCO parameters for System Generator
11/11/02	2.3	Added device utilization table
01/23/03	2.4	Update for EDK SP3
07/10/03	2.5	Update for EDK Granite
07/29/03	2.5.1	Change DS213 to DS466 because of duplications
8/19/04	2.6	Updated for Gmm; updated trademarks and supported device family listing.