CREATING DIGITAL FINGERPRINTS ON COMMERCIAL FIELD PROGRAMMABLE GATE ARRAYS

Author 1, Author 2, Author 3

Address line 1
Address line 2
Address line 3
email: email 1, email 2, email 3

Author 4

Address line 1
Address line 2
Address line 3
email: email 4

ABSTRACT

In this paper, we proposes the method of creating a circuit identifier, or digital fingerprint, for field programmable gate arrays (FPGAs). The proposed digital fingerprint is a function of the natural variations in the semiconductor manufacturing process that cannot be duplicated or forged. The proposed digital fingerprint allows the use of any arbitrary of nodes internal to the circuit or the circuit outputs as monitoring locations. Changes in the signal on a selected node or output can be quantified digitally over a period of time or at a specific instance of time. Two monitoring methods are proposed, one using cumulative observation of the nodes and the other samples the nodes based on a signal transition. Two monitoring methods were validated on a small sample of twenty Xilinx® Virtex-II Pro FPGAs, where both methods successfully created unique identifiers for each FPGA.

1. INTRODUCTION

Current industry trends in FPGA design creation mirror standard practices in software programming, namely the use and reuse of small modules of functionality. Thus, there has been an increase in the need to prevent unauthorized use of intellectual property (IP). A method is needed to identify a piece of hardware through the use of a unique signature that would tie functionality to the physical silicon on which it resides without modifying the internal architecture to allow usage of current commercial designs. This method can prevent the theft of the protected IP by not allowing it to be run on any other silicon. The first step in signature creation is identifying a methodology for differentiating multiple functionally and structurally identical circuits from the same vendor.

Recent research [1, 2] created physical uncloneable functions (PUFs) which are stand-alone structures specifically designed to create an ID via variations in internal delay of the PUF function due to manufacturing process variations. However, PUFs and other ID method [3] rely on finalized stable output values to characterize the circuit and often requires specialized physical hardware to generate IDs. In this paper, we proposed novel methods that monitor any selected nodes for their cumulative and transitional behaviors to determine the circuit's unique digital identification or fingerprint.

2. THE DIGITAL FINGERPRINT

The digital fingerprint creates an unique identifier based on the characterization of signals that are unique to a particular physical implementation of a functionality. Although the same circuit design and fabricated in the same way should be exactly identical, but in reality this is not the case. The semiconductor fabrication process is not perfect and variations exist in each chip. These variations can, for example, occur in the doping profile, mask alignment, metal deposition, oxide growth, or transistor gate width and length and have been proven to have a statistically significant effect on some circuit attributes [4, 5]. An examination of the Virtex-II Pro FPGA documentation [6, 7, 8] reveals delays for various components are given in ranges. If the fabrication process was perfect, these ranges would not exist.

The waveforms that occur on a signal line inside a circuit are dependent on both the circuit functionality that drives the signal line and the variations of the physical implementation of that functionality. A fingerprint that examines the signal characteristics at a specific node can create an identifier that will be unique to a chip due to these two dependencies.

Naturally, this approach assumes an ideal environment where temperature and supply voltage of the circuit are carefully monitored and controlled. In reality, there exists a range of temperatures and supply voltages that the circuit will operate at. The digital fingerprint must be robust enough to provide a consistent output over this operational range. In this paper, we also present the effects of various operat-

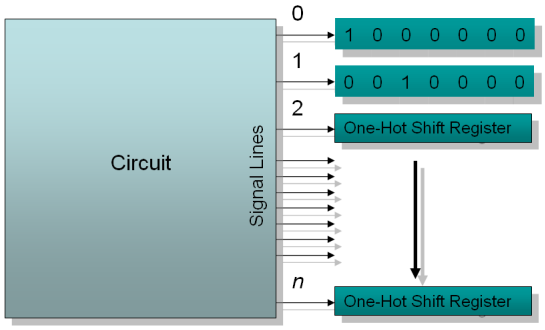


Fig. 1. Block structure of NCS fingerprint method.

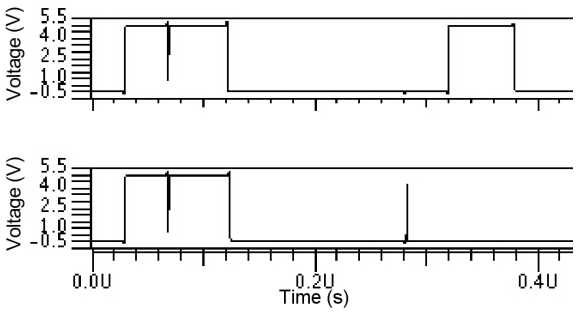


Fig. 2. Arbitrary signals with transitions.

ing temperatures and supply voltages on the proposed digital fingerprinting methods.

3. TRANSITION BASED FINGERPRINTS

Instead of using a stand-alone structure to create a digital fingerprint, we use only the existing structure to monitor the transitions between '0' and '1'. We propose two methods, nodal cumulative sampling method and transitional sampling method to create a unique ID of FPGA.

3.1. Nodal Cumulative Sampling

The nodal cumulative sampling (NCS) method summarizes transitions, either $0 \rightarrow 1$ or $1 \rightarrow 0$, over multiple signal lines to create a digital fingerprint. These lines are from various places in a circuit and are connected to the clock input of one-hot-encoded shift register, as shown in Figure 1. The register has a '1' value on the LSB and as transitions are detected on a signal line, '0's are shifted in causing the '1' to shift towards the MSB. The bit that the '1' ends on after all transitions are counted is the fingerprint value for that line. Performing this process over multiple lines results in a base-*n* digital fingerprint, where *n* is the maximum number of transitions seen for the signal set. Variations in the signals and the setup/hold times of the one-hot shift register will result different fingerprint values across multiple circuit implementations.

As an example, Figure 2 shows the basic concept by performing the summation of the $0 \rightarrow 1$ transitions for two signal lines. The fast $0 \rightarrow 1$ transition in the bottom signal may or may not be captured by the shift register depending on its setup/hold times resulting in two unique identifiers.

3.2. Transitional Sampling

Transitional sampling involves capturing the current value on multiple signal lines through the use of a shift register, as shown in Figure 3. The trigger to sample the signals is a

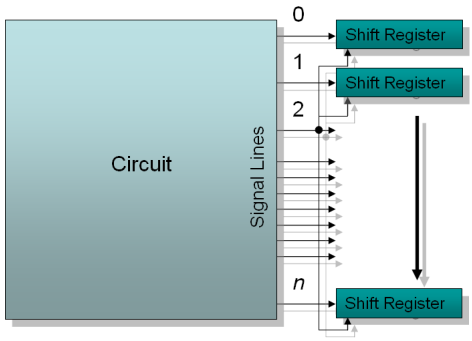


Fig. 3. Block structure of the transitional sampling fingerprint method with sig. 2 acting as a clock.

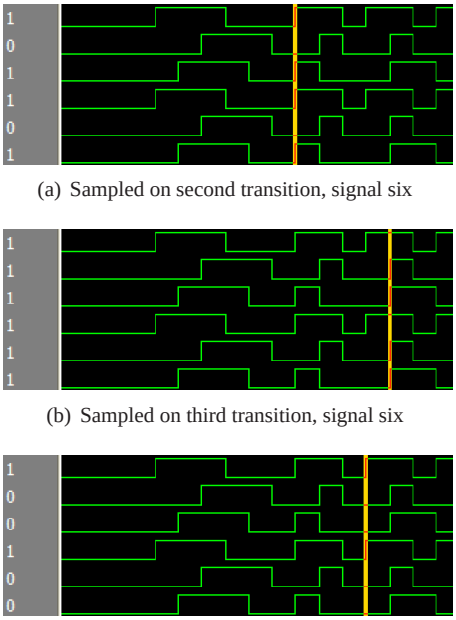
transition, either $0 \rightarrow 1$ or $1 \rightarrow 0$. Figure 4 shows three different fingerprints created using the sample six signals with the sampling happening at different transitions on different signals. As with the NCS method, variations in the signals will result in different digital fingerprints.

4. RESULTS

4.1. Nodal Cumulative Sampling Digital Fingerprint

Both the NCS and transitional sampling metric were tested using a 32-bit combinational multiplier and examining its outputs. While more complex circuit could have been used and an internal node examined, the multiplier provided easy access to the FPGA LUTs.

As the number of transitions on an output change between FPGAs, these values can be used to create a digital fingerprint. Transition counts range from 0-6 suggesting a base-7 fingerprint. For proof of concept and easy of understanding, a base-8 fingerprint will be used. For a 32-bit multiplier output (64 bits), this provides the theoretical upper bound of unique fingerprints as : $8^{64} \approx 6.28 * 10^{54}$. Examination of the data shows that the actual number is less due to two constraints. First, a number of outputs, mainly focused in outputs 0-5 and 59-63, report the same value across all FPGAs. This is due to a lack of transitions on these sig-



(a) Sampled on second transition, signal six

(b) Sampled on third transition, signal six

(c) Sampled on third transition, signal 4

Fig. 4. Transitional sampling performed on six signals.

Table 1. Transition counts for selected outputs.

FPGA	Pin_{15}	Pin_{14}	Pin_{13}	Pin_{11}	Pin_{10}	Pin_{9}	Pin_{7}	Pin_{6}
1	2	2	2	3	2	1	1,2,3	1
2	2	2	2	3	2	1	1,2,3	1
3	2	2	2	2	2	1	2	1
4	2	2	2	2	2	1	3	2
5	2	2	2	2	3	1	3	1
6	2	2	2	2	2	1	2	1
7	2	2	2	2	2	1	2	2
8	2	3	2	2	2	1	2	2
9	2,3	3	3	2	2	2	3	1,2
10	2,3	2	2	3	3	1	3	1,2

nal lines, as the number of LUTs for feeding these outputs is lower than for other outputs due to the staggered nature of the ripple adders.

The second constraint is the number of outputs that show some oscillation of their transition count values. Depending on the severity of the oscillation, these outputs could still be used but a error correction factor for the fingerprints would have to be created to account for this activity. As an example of the effect of these constraints, Table 1 shows selected outputs for all ten FPGAs. Only five of the outputs meet the constraints. Some outputs that show no change, may also meet the constraints with testing of additional FPGAs.

Applying these two constraints to the entire multiplier output results in 18 outputs that can be used to create a digital fingerprint. Table 2 shows the resulting fingerprints for

Table 2. Digital Fingerprints for the nodal cumulative sampling circuit FPGAs.

FPGA	Multiplier Outputs																	
	Pin_{51}	Pin_{48}	Pin_{47}	Pin_{46}	Pin_{42}	Pin_{35}	Pin_{34}	Pin_{30}	Pin_{27}	Pin_{25}	Pin_{21}	Pin_{19}	Pin_{17}	Pin_{14}	Pin_{13}	Pin_{11}	Pin_{10}	Pin_{9}
1	3	5	4	2	2	5	5	4	2	4	4	4	2	2	2	3	2	1
2	3	5	4	3	1	5	4	5	2	4	4	3	3	2	2	3	2	1
3	3	5	4	2	1	5	4	5	2	4	4	4	3	2	2	2	2	1
4	3	5	4	2	1	5	5	4	2	3	1	4	2	2	2	2	2	1
5	3	4	4	3	1	5	5	4	3	4	4	3	3	2	2	2	3	1
6	3	5	4	2	1	6	4	6	2	4	4	4	3	2	2	2	2	1
7	2	5	4	2	2	5	3	5	2	3	4	5	3	2	2	2	2	1
8	3	5	3	2	2	3	5	4	2	3	1	4	3	3	2	2	2	1
9	2	4	4	2	2	3	5	4	3	3	1	3	3	3	3	2	2	2
10	2	5	4	2	1	3	4	4	3	3	4	4	3	2	2	3	3	1

the ten FPGAs. Each is unique and with using only 18 values, potentially $8^{18} = 18,014,398,509,481,984$ FPGAs can be identified.

Because implementation of this digital fingerprint will be done on a binary system, conversion from base-8 to binary results in a three-fold increase in the length of the ID without altering the number of possible IDs. This allows 18 bits of the multiplier output to become a 54-bit ID or through restructuring of the ripple adders, the entire 64-bit output becomes a 192-bit value.

Regardless of which implementation is used, all ten FPGAs can be distinctly identified.

4.2. Transitional Sampling Digital Fingerprint

Testing of transitional sampling was done on twenty FPGAs, utilizing nine circuit outputs serving as trigger lines to capture four samples per output. Analysis shows that the sample values are fairly consistent over multiple data capture runs for all FPGAs and output trigger signals. For a number of the output samples, variations in the data stored in the shift registers exist. These are caused by two different types of errors. The first is due either to outputs being on the edge of readability by the shift register or clock signal skew causing the capture to be off. This causes only a few bits to change over multiple runs. The second is due to the clock signal being read inconsistently by the shift registers resulting in a large number of bits changing over multiple runs. Both of these errors are due to the same underlying cause, namely the transition being on the edge of readability by the shift registers. Table 3 shows the samples of a single board with both output and clock based errors.

In order validate this digital fingerprint method, all twenty FPGAs must be differentiable across a single sample. This is done by performing a bitwise XOR between the sample values across all the FPGAs for a particular clock signal. Any differences will result in a '1' value for that bit. Since

Table 3. Output and clock errors for bits 47-32 of FPGA 5, Clk 25. commonly appearing values, clock error, and output error.

Output	Samples n_0	n_1	n_2	n_3
1	0x9258	0x9055	0x1007	0x3a55
2	0x9258	0x921e	0x9055	0x1007
3	0x9208	0x9055	0x1007	0x3a55

Table 4. FPGA 16, Clk 25 w/ error factors

Output	Samples n_0	n_1	n_2	n_3
1	0x9258	0x9055	0x1007	0x3a55
2	0x9258	0x921e	0x9055	0x1007
3	0x9208	0x9055	0x1007	0x3a55
Errors	2	5	13	15

errors are being seen in the samples, any differentiation between FPGAs must also take into account errors when the sample value is read. To do this, we create an error factor for each sample of each FPGA. This factor is the maximum number of bits that a sample is seen to change from its normal output, i.e. the one that shows up the most often. An output error or a clock error value is used as the error factor as they are mutually exclusive. In the case of no clear, consistent output, one was chosen at random from the multiple runs of data. Table 4 adds the error factors for the samples of the FPGA given in Table 3.

Table 5 summarizes the results showing FPGA differentiation for each clock signal and sample. Also shown is the breakdown for the number of bits that varied and stayed the same between FPGAs for that clock along with the number of bits that had an error for at least one FPGA for that sample and clock.

Examination of these results shows that on average the best differentiation is achieved with sample n_2 despite the number of errors, with clock 24 correctly identifying all 20 FPGAs. The erratic differentiation values across all samples and clocks come from which bits have error. If the majority of error is on bits that should be the same between FPGAs, there is little effect on the differentiation. Based on the data, these cases happen when there are to few bits to differentiate FPGAs, thus the result is the same. When the error effects bits that are different between FPGAs, naturally the differentiation is greatly reduced.

The creation of a digital fingerprint is dependent upon being able to have a repeatable value. The transitional sampling circuit is hard to control and requires a good clock signal that provides transitions that meet the shift register's flip-flops' setup and hold times. This circuit may be more applicable in conjunction with nodal cumulative sampling, by providing an asynchronous sampling of the transition count

Table 5. FPGA differentiation results by sample.

(a) n_0

Clk	#Differentiated	#of bits same	#of bits different	#errors
18	8	28	36	26
19	11	53	11	4
21	9	53	11	7
24	17	36	28	8
25	4	56	8	9
28	4	56	8	10
30	9	35	29	17
34	2	47	17	18
40	10	43	21	6

(b) n_1

Clk	#Differentiated	#of bits same	#of bits different	#errors
18	14	29	35	33
19	5	33	31	27
21	6	38	26	16
24	8	43	21	5
25	3	45	19	20
28	1	45	19	37
30	5	28	36	35
34	5	25	39	28
40	11	31	33	19

(c) n_2

Clk	#Differentiated	#of bits same	#of bits different	#errors
18	13	30	34	28
19	4	28	36	27
21	8	34	30	12
24	20	30	34	3
25	7	22	42	30
28	1	21	43	43
30	11	19	45	43
34	13	14	50	24
40	11	19	45	29

(d) n_3

Clk	#Differentiated	#of bits same	#of bits different	#errors
18	13	44	20	20
19	4	38	26	32
21	8	35	29	18
24	13	32	32	19
25	5	14	50	38
28	2	17	47	51
30	15	13	51	47
34	3	14	50	52
40	2	14	50	30

value.

4.3. Effects of Temperature on Digital Fingerprints

Silicon circuit performance is dependent on the temperature of the environment in which it is operating. A transistor turns 'on' when it conducts current from the source to the drain. The transistor is said to be in saturation mode when the voltage difference between the source and drain is greater the threshold voltage, V_t. The current conducted through the transistor from source to drain is denoted by I_{ds}. The saturation current is dependent on the temperature because the carrier mobility of the electrons that pass between the source and drain is reduced at higher temperatures. Thus, the current at the drain during saturation, I_{dsat} reduces as temperature increases. Also, the magnitude of the threshold voltage V_t decreases with rising temperatures, which increases the transistor's noise sensitivity. This is shown by the equation

$$V_t = V(T_r) - k_{vt}(T - T_r) \quad (1)$$

where T is the absolute temperature in Kelvin, T_r is the room temperature in Kelvin, and k_{vt} is a constant that ranges from 0.5 to 3.0 $mV/$K. Overall, a circuit will function at a lower frequency at higher temperatures. So, it is expected that the multiplier in the digital fingerprint will function slower, producing longer glitches. Glitches that were too short to meet the setup and hold times of the output shift register flip-flops at room temperature would now be able to be captured. Thus the glitch count could increase. In addition, the unstable bits could become more stable. Suppose bit X was an unstable bit producing between 3 and 4 glitches. It is assumed that this was because the fourth glitch was sometimes too short to be captured by the shift register. However, with an increase in temperature , the fourth glitch would be longer and there would be a greater chance of it getting captured.

However, in addition to the multiplier slowing down, the output shift register transistors will also slow down at higher temperatures. Thus, the setup and hold times of the shift registers would increase as well. So the actual effect of temperature on the digital fingerprint circuit would be a race between which component slows the most at higher temperatures - the multiplier or the output shift registers.

4.4. Results of Temperature on Digital Fingerprints

The Xilinx Virtex 2 Pro FPGA was heated to the specified maximum operating temperature of 85 degrees C [9]. Measurements were then done using a Agilent 16902A logic analyzer to read the glitch count. The center multiplier output bits were recorded in table 6 as these were the bits that showed the maximum variation among the different FPGAs in previous testing. Results showed that the majority of the glitch counts remained the same for the bits that were tested. Bits 24, 28, 30 and 31 showed different results under heat than at room temperature. These results do not show a definite pattern in the bit shifts that occur when the FPGA is operating under high heat. In the case of bits 24 and 31, the glitch counts were unstable at room temperature but became stable under heat. The reverse was true for bit 28 where the glitch count was stable at room temperature and unstable when heated. Also, in the case of bit 30, glitch counts were stable but different for room temperature and under heat. However, the fact that the majority of the bits stayed the same under heat shows promise. It may be possible to allow for certain bit flips by inserting an error correction component to the Digital Fingerprint circuit. This could be done by using a strategy such as error detection and error correction codes using Hamming distance.

4.5. Effects of Supply Voltage on Digital Fingerprints

A transistor conducts current when the voltage difference between its source and drain, V_{ds} becomes equal or greater than the threshold voltage, V_t. When $V_{ds} > V_t$, the transistor is in saturation mode. As V_{ds} increases, the mobility of the electrons going from drain to source begins to level off. At some V_{ds} the electron mobility becomes fully saturated and raising V_{ds} will not effect the carrier mobility. For 180nm technology used in Xilinx Virtex 2 Pro FPGAs, this voltage is 0.36V [10]. For testing purposes, we assume that the supply voltage V_{DD} is the same as V_{ds}. The Xilinx power supply operating voltage range is $5V \pm 0.5V$ [11]. This is much higher than 0.36V and so the glitch count is not expected to change much due to supply voltage fluctuations.

Table 6. Effects of Heat on Digital Fingerprint Glitch Counts

Glitch Count			
Bit No.	Room Temp.	Max. Temp (85∘C)	Bit Diff.
24	3,4	3	0, +1
25	4	4	0
26	4	4	0
27	2	2	0
28	5	4,5	-1, 0
29	5	5	0
30	5	4	-1
31	6,7,8	6	0, -1, -2
32	4	4	0
33	4,5	4,5	0,0
34	3,4	3,4	0,0
35	5	5	0

4.6. Results of Supply Voltage on Digital Fingerprints

The Agilent E3631A output power supply was used to vary the supply voltage to the FPGA board. Voltage was confirmed using a multimeter. Glitch counts for various input bits were recorded for 4.5V, 5V and 5.5V and are shown in Table 7. On the whole, most of the glitch counts remained the same. For bits 26 and 28, applying 4.5V to the supply increased their instability. However as the majority of the bits remained the same, this too can be accounted for by using an error correction circuit.

4.7. Statistical Sampling on FPGAs

With both digital fingerprinting circuits, all FPGAs can be identified, 10/10 for the glitch count and 20/20 for the asynchronous capture. These numbers represent a fairly small number compared to the total population of Virtex-II Pro FPGAs currently available. In order to relate this success to the larger population, use of statistical sampling theory is required. If ε is the sampling error and *n* is the sample size, then we have

$$1 \backslash \varepsilon^2 \approx n \tag{2}$$

Thus, for a sample size of 10, we have the statistical

Table 7. Effects of Voltage Fluctuations on Digital Fingerprint Glitch Counts

Glitch Count			
Bit No.	5V	5.5V	4.5V
24	3,4	3,4	3,4
25	4	4	4
26	4	4	4,5,6
27	2	2	2
28	5	5	4,5
29	5	5	5
30	5	5	5
31	6,7,8	6,7,8	6,7,8
32	4	4	4
33	4,5	4,5	4,5
34	3,4	3,4	3,4
35	5	5	5

sampling error is $\pm 31.62\%$ and for a sample size of 20, it is $\pm 22.36\%$.

5. CONCLUSION

This paper proposed that variations in semiconductor fabrication have a measurable effect and can be used in conjunction with circuit functionality to create a natural serial number, or digital fingerprint, that is unique to each FPGA. Three methods of creating this digital fingerprint were proposed, asynchronous LFSR, nodal cumulative sampling, and transitional sampling, of which the latter two were able to be implemented on an FPGA. Test circuit functionality was implemented on the FPGA to gain access to the LUTs directly in order to properly test the digital fingerprint methods. The resulting data shows variations in the number of transitions of the output bits across multiple FPGAs. The variations are for the most part consistent and provide a good foundation to allow the creation of a digital fingerprint. The two fingerprint methods implemented, using the number of transitions directly as a fingerprint and using the transitions to sample signals asynchronously, both resulted in successful identification of all FPGAs.

While $0 \rightarrow 1$ transitions were utilized $1 \rightarrow 0$, 1 and 0-level pulses can also be used. Additionally NCS, transitional sampling, and a LFSR could be combined to make a much more complex fingerprint.

6. REFERENCES

[1] B. Gassend, D. Lim, D. Clarke, M. van Dijk, and S. Devadas, "Identification and authentication of integrated circuits," *Concurr. Comput. : Pract. Exper.*, vol. 16, no. 11, pp. 1077–1098, 2004.

[2] G. E. Suh and S. Devadas, "Physical uncloneable functions for device authentication and secret key generation," in *DAC '07: Proceedings of the 44th Annual Conference on Design Automation*. ACM, 2007, pp. 9–14.

[3] Y. Su, J. Holleman, and B. Otis, "A 1.6pj/bit 96circuit using process variations," *Solid-State Circuits Conference, 2007. ISSCC 2007. Digest of Technical Papers. IEEE International*, pp. 406–611, Feb. 2007.

[4] K. Bernstein, *High Speed CMOS Design Styles*. Kluwer Academic Publishers, 1998.

[5] D. Boning and S. Nassif, *Design of High Performance Microprocessor Circuits*. Wiley-IEEE Press, 2000, ch. Models of Process Variations in Device and Interconnect.

[6] Xilinx, *Virtex-II Pro and Virtex-II Pro X Platform FPGAs: Complete Data Sheet*, v4.6 ed., March 5 2007.

[7] ——, *Local Clocking Resources in Virtex-II Devices*, v1.2.1 ed., April 23 2007.

[8] ——, *Local Clocking for MGT RXRECCLK in Virtex-II Pro Devices*, v1.1 ed., November 18 2004.

[9] *Virtex-II Pro and Virtex-II Pro X Platform FPGAs: Complete Data Sheet*, 4th ed., Xilinx, November 2007.

[10] N. H. Weste and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective*. Addison-Wesley, 2004.

[11] *Xilinx University Program Virtex-II Pro Development System: Hardware Reference Manual*, 1st ed., Xilinx, March 2005.