Characterization of Long Wire Data Leakage in Deep Submicron FPGAs

George Provelengios
UMass Amherst
Amherst, MA
gprovelengio@umass.edu

Chethan Ramesh
UMass Amherst
Amherst, MA
cramesh@umass.edu

Shivukumar B. Patil
UMass Amherst
Amherst, MA
spatil@umass.edu

Ken Eguro
Microsoft Research
Redmond, WA
eguro@microsoft.com

Russell Tessier
UMass Amherst
Amherst, MA
tessier@umass.edu

Daniel Holcomb
UMass Amherst
Amherst, MA
dholcomb@umass.edu

ABSTRACT

The simultaneous use of FPGAs by multiple tenants has recently been shown to potentially expose sensitive information without the victim's knowledge. For example, neighboring long wires in SRAM-based FPGAs have been shown to allow for clandestine data exfiltration. In this work, we explore distinct characteristics of this signal crosstalk that could be used to enhance or prevent information leakage. First, we develop a mechanism to characterize the crosstalk coupling that exists between neighboring wires at the femtosecond scale. Second, we show that it is possible to reverse engineer channel layouts by determining which pairs of routing resources/links in the channel exhibit coupling to each other even if this information is not provided by the FPGA vendor. To fully characterize these effects, we examine long wire coupling on different types of wires across three devices implemented in different technology nodes from 65 to 20 nm. We experimentally demonstrate that information leakage is apparent for all three FPGA families.

CCS CONCEPTS

• **Computer systems organization → Security**;

KEYWORDS

FPGA, side channel, crosstalk

ACM Reference Format:
George Provelengios, Chethan Ramesh, Shivukumar B. Patil, Ken Eguro, Russell Tessier, and Daniel Holcomb. 2019. Characterization of Long Wire Data Leakage in Deep Submicron FPGAs. In *The 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays (FPGA '19), February 24–26, 2019, Seaside, CA, USA.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3289602.3293923

FPGA '19, February 24–26, 2019, Seaside, CA, USA

ACM ISBN 978-1-4503-6137-8/19/02…$15.00
https://doi.org/10.1145/3289602.3293923

1 INTRODUCTION

As the use of FPGAs for computing becomes ubiquitious, the platforms and compute models supported by FPGAs become more diverse. While most FPGA deployments continue to support the use of the entire FPGA with circuits created by a single entity, the emergence of *multi-tenant* FPGA scenarios with circuits created and used by multiple users has grown in interest. Recent work has recognized embedded computing with cores from multiple sources [1] and cloud computing with multiple users sharing FPGA hardware [4] as contemporary multi-tenant scenarios with many more virtualization opportunities on the horizon [6].

While the multi-tenant use of FPGAs provides a mechanism for maximizing the utilization of FPGA resources, it does also present unique security challenges. Several recent research studies have shown that multi-tenant scenarios can lead to side channel attacks where the attacker does not have physical access to the FPGA [1, 4, 5, 8]. One class of these attacks [1, 4] uses information obtained using a single "attacker" wire that is adjacent to a victim wire in an FPGA routing channel. Although initial work has shown these types of attacks to be robust, the full nature of the threat is unclear, as the level of accuracy in quantifying data transmission between neighboring wires has not been comprehensively explored.

In this work, we address three important issues related to crosstalk-based attacks in SRAM-based FPGAs. Our contributions include:

- We present a precise characterization of the effect of a neighboring wire on a channel wire's delay. This new model is shown to be robust across a range of hardware implementations of attack circuitry.
- In some cases, FPGA companies do not publicly disclose wiring adjacency for FPGA routing channels, limiting a user's ability to ensure that wires adjacent to critical routes are unused. In this work, we show that it is straightforward to determine routing adjacency for all channel wires using crosstalk effects as a guide in building an adjacency map.

The remainder of the paper is organized as follows. Section 2 provides perspective on the delay and adjacency characterizations performed in this paper. The methodology used in our work is detailed in Section 3. In Section 4 we explain our approach for determining channel adjacency. The sensitivity of each wire to

coupling is addressed in Section 5. Section 6 concludes the paper and offers directions for future work.

2 BACKGROUND

2.1 FPGA Long Wire Attacks

The presence of a communication channel between adjacent FPGA long wires ("long lines") has previously been confirmed for both Xilinx [1] and Intel [4] FPGAs. In both studies it was shown that the logic value carried on a wire changes the delay of its immediate neighbor in a significant and measurable way. A logic 1 value *transmitted* by the victim effectively reduces the delay on the adjacent wire and a logic 0 has the opposite effect. Effectively, the delay change allows a wire to *receive* information about its neighbor, potentially allowing the information to be used in a clandestine attack. Giechaskiel *et al.* [1] examined the effects of transmitter switching rate and wire length, among other parameters. Ramesh *et al.* [4] showed that crosstalk-based leakage could be used as a side channel to successfully obtain a 128-bit key from an FPGA implementation of the AES block cipher. The wire leakage is well suited for use in side channel attacks, which are inherently robust to noise and able to exploit small correlations between the side channel measurements and secret data.

Both studies noted above relied on the use of a ring oscillator (RO) to receive information from the victim (transmitter). One RO wire is adjacent to the victim wire, and the frequency of oscillation is obtained by a binary counter triggered by the RO. The difference in RO frequency for two trials is determined by using a relative count metric [4] determined over two measurement periods. The count difference ΔRC when first a logic 0 (first trial) and then a logic 1 (second trial) are transmitted can be represented as:

$$\Delta RC = \frac{C^1 - C^0}{C^1} \tag{1}$$

where C^1 and C^0 are the measured counts for transmitted logic 1 and 0, respectively. Although useful, the results of this approach depend on the delay of the entire RO rather than just the delay of the wire adjacent to the victim. In this work we more precisely quantify the delay effects caused by wire adjacency to make the characterization of transmitter values clearer.

The precise delay characterization of FPGA wires has been explored in several contexts unrelated to signal adjacency. Yu *et al.* [7] used ROs to measure the delay of a number of FPGA resources, including channel wires in isolation. Gojman *et al.* [3] employed a path-based approach to consider delays of all channel wires in an FPGA. Fine-grained FPGA timing measurement using time-to-digital converters (TDCs) was used by Gnad *et al.* [2] to assess process variations. None of these studies considered the differentiation of same-wire delays due to the behavior of surrounding wires.

2.2 Determining FPGA Channel Wire Adjacency

FPGA vendors differ in terms of providing customers with easy access to channel adjacency information for their FPGA devices.

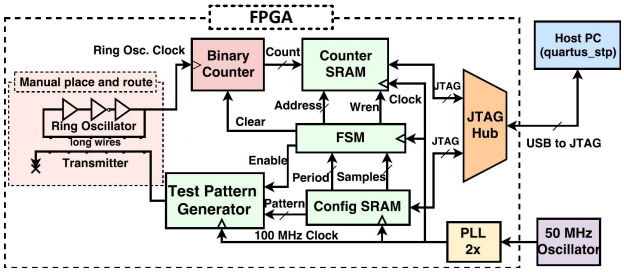


Figure 1: Experimental framework for evaluating long wire delay effects on SRAM FPGAs [4].

Adjacency information for Xilinx SRAM FPGA devices can be visually determined from Vivado floorplanning tools, version 2018.2. However, the corresponding view in Intel's visual editor does not allow a user to infer adjacency (Quartus Prime v18.1). Knowledge of adjacency is necessary if a user wishes to deploy fine-grained isolation by ensuring that sensitive wires have no neighbors that could snoop on their values using crosstalk.

3 METHODOLOGY

We perform experiments on three different classes of FPGAs that are fabricated in different technology nodes. Our experiments are performed on two Cyclone IV GX (EP4CGX150DF31) FPGA Development Kits, one Stratix V (5SGXEA7K2F40C2N) GX Development Kit, and one DE5a-Net Arria 10 GX (10AX115N2F45E1SG) FPGA Development Kit. The Cyclone IV, Stratix V, and Arria 10 devices are implemented in 60nm, 28nm, and 20nm CMOS technologies, respectively.

Fig. 1 shows the block diagram of the test setup used to assess the long wire covert channels in these system types. In each experiment, the *transmitter* and *receiver* are implemented in the FPGA in one or more vertical long wires. A test pattern generator assigns either a logic 1 or a logic 0 to the transmitter in each trial and the effect on the frequency of the receiver is measured by counting its oscillations during 1024 trials of 21ms each unless noted otherwise. Half of the 1024 trials use a transmitted value of 1, and the other half use a transmitted value of 0. The ring oscillator, transmitter and receiver are placed and routed using place and route constraints.

3.1 Metric

We introduce a new metric Δt that captures the amount by which the value of the transmitter affects the propagation delay of transitions on the receiver wire. This metric is designed to eliminate the RO-based variability introduced by the ΔRC metric in Eq. 1. The changes in propagation delay on the receiver wire are on the order of 100s of femtoseconds, and cannot be measured directly. However, they can be inferred from frequency measurements collected by on-chip circuitry counting ring oscillator cycles.

During each period of a ring oscillator, every circuit node in the ring makes exactly one rising and one falling transition. The period of a ring oscillator that contains a particular receiver wire of

interest can be described as the sum of four terms: $d_{rx\uparrow}$ represents the propagation delay of a rising transition on the receiver, $d_{n\uparrow}$ represents the summed propagation delays of one rising transition on all other ring nodes, and $d_{rx\downarrow}$ and $d_{n\downarrow}$ represent the receiver and summed ring node delays for the corresponding falling transitions. Using superscripts to denote the value of the transmitter during a measurement, the frequency of the ring when the transmitter holds a value of 1 can therefore be written as $f^{(1)}$ shown in Eq. 2. Term $f^{(0)}$ is defined analogously for the case of a 0-valued transmitter. Measuring the frequency of the same ring oscillator with a transmitted 0-value and 1-value allows for calculating Δt as shown in Eq. 3. The delay terms ($d_{n\uparrow}$ and $d_{n\downarrow}$) that are unrelated to the receiver wire cancel out from the two frequency measurements, leaving only the delay changes in the receiver wire. The value Δt represents the change in propagation delay on the receiver wire that is caused by the change on the value of the transmitter. More precisely, as shown by the second line of Eq. 3, Δt is the average propagation delay change over rising ($d_{rx\uparrow}$) and falling transitions ($d_{rx\downarrow}$) of the receiver. We make no claim as to whether the change in receiver delay is occurring predominantly on one transition or equally on both.

$$f^{(1)} = \frac{1}{d_{n\uparrow} + d^{(1)}_{rx\uparrow} + d_{n\downarrow} + d^{(1)}_{rx\downarrow}} \qquad (2)$$
$$f^{(0)} = \frac{1}{d_{n\uparrow} + d^{(0)}_{rx\uparrow} + d_{n\downarrow} + d^{(0)}_{rx\downarrow}}$$

$$\begin{aligned}\Delta t &= \left(\frac{1}{f^{(0)}} - \frac{1}{f^{(1)}}\right)/2 \\ &= \left(\left(d^{(0)}_{rx\uparrow} - d^{(1)}_{rx\uparrow}\right) + \left(d^{(0)}_{rx\downarrow} - d^{(1)}_{rx\downarrow}\right)\right)/2\end{aligned} \qquad (3)$$

The Δt metric is different from the metric of fractional change in oscillator counts that is used in prior work [1, 4] and also shown in Eq. 1, and we use Fig. 2 to demonstrate the motivation for using the new metric. The two cases shown in Fig. 2 use the same neighboring transmitter and receiver wires on the same Cyclone IV chip, with the transmitter and receiver wires running parallel to each other for a length of 10 C4 wire segments running upward from position X113Y2. The only difference between the two scenarios is that the ring oscillator circuit used for measurement in the figure at left has extra wiring delay added to the ring intentionally, such that its period is roughly 50% higher than the circuit used for the figure at right. The added delay is on a part of the ring away from the transmitter and receiver wires, and therefore does not impact their coupling. A good metric should indicate the same amount of coupling in both cases. In each case, we measure the oscillator frequencies as shown and compute the value of Δt using Eq. 3, obtaining values of 3.28 ps in the first case and 3.32 ps in the second. The good agreement between the two experiments demonstrates that Δt captures the change of the receiver delay while being insensitive to the overall ring delay. The prior metric of ΔRC is sensitive to the overall ring delay and yields a fractional change in oscillator frequency of 2.66E-4 and 4.05E-4 in the two experiments (a 52% discrepancy), despite no changes to the part of the circuit in which the coupling occurs. Removing the dependence of the characterization metric on oscillator frequency is important for accurately characterizing the leakage, because the ring oscillator frequency will inevitably change across experiments that vary parameters such as technology node, or the length or type of wires used for transmitter and receiver.

4 RECOVERING CHANNEL LAYOUT THROUGH MEASUREMENT

In this section we show that characterizing the coupling between wires makes it possible to infer channel layout, which could enable design isolation techniques to reduce the risk of leakage between adjacent wires. Layout/adjacency information of a channel is inferred by testing all possible transmitter-receiver pairs in the channel, and measuring the value of Δt to check for evidence of coupling for each pair. Wires that impact each other are reasonably assumed to be neighbors in the channel.

The C4 channel in a Cyclone IV device has 96 wires, of which 48 travel in the upward direction. We explore these 48 wires to determine which are neighbors. Each LAB can connect to 12 of the 48 wires, and it takes a vertical span of 4 logic array blocks (LABs) to fill the channel. Each of these 48 wires can be the receiver or transmitter, so 2,304 pairs of wires are considered to exhaustively characterize the channel. Fig. 3 demonstrates the coupling that exists between all pairs of the 48 wires. The measurements are collected using transmitter and receiver wires that are 10 C4 wires long, and then normalizing the value of Δt to the length of a single C4. In particular, the 48 wires in the channel being characterized are driven from LABs `X12Y2`, `X12Y3`, `X12Y4`, and `X12Y5`. The correspondence between the indices $0-47$ and the physical resources of Cyclone IV are given in Tab. 1. Looking carefully at Fig. 3 we can see for example that transmitters at the 16th and 40th indices induce significant values of Δt on a receiver in the 4th index. This implies that wires `X12Y4S0I4` (16th index) and `X12Y6S0I4` (40th index) are likely the neighbors of `X12Y3S0I4` (4th index). For most of the 48 wires, when used as receivers, we are able to identify two other wires that as transmitters cause significant values of Δt. These are hypothesized to be the left and right neighbors in the channel. Some wires in the channel do not have two clear neighbors, and this will be investigated in future experiments. The coupling is observed to be bidirectional; if there is a significant effect when the transmitter is index i and the receiver is index j, then a similar value of Δt will occur when the receiver is index i and the transmitter is index j.

5 CHARACTERIZATION

5.1 Susceptibility of Each Wire to Leakage

We are able to identify neighbors for all C4 wires in the channel of the Cyclone IV device using the technique from the previous section (see Fig. 3). Based on finding the same adjacency information for six different channels on the device, we assume that all channels are similar, and collect results from experiments performed across multiple channels. Fig 4 shows for each wire in the channel, the range of Δt values indicating how much the wire delay can be changed by the value of its neighbor. There is a range of values

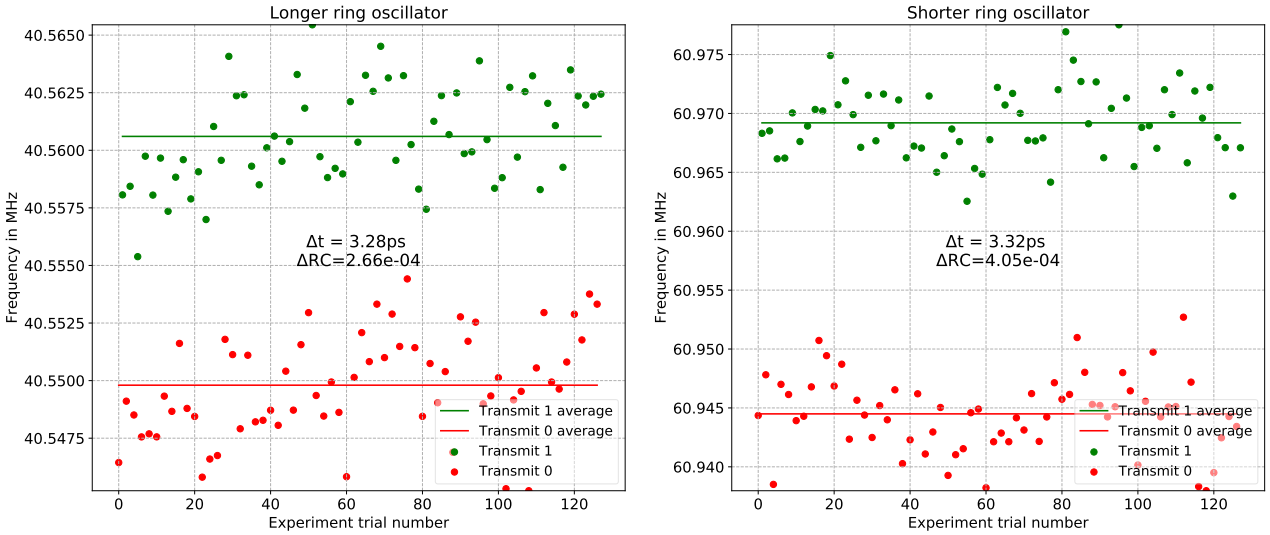


Figure 2: Figure shows measured receiver frequency for the same transmitter and receiver wires when measured with two different length ring oscillators. The two cases yield a similar value of Δt but different values of the prior metric of fractional count difference (ΔRC). This result demonstrates that Δt is invariant to ring frequency but ΔRC is not. Receiver frequency in each trial is measured by counting the number of oscillations in one second.

Index	Logic Element	Wire in Channel
0	LCCOMB_X12_Y2_N0	X12Y3S0I0
1	LCCOMB_X12_Y2_N2	X12Y3S0I1
2	LCCOMB_X12_Y2_N4	X12Y3S0I2
3	LCCOMB_X12_Y2_N6	X12Y3S0I3
4	LCCOMB_X12_Y2_N10	X12Y3S0I4
5	LCCOMB_X12_Y2_N14	X12Y3S0I5
6	LCCOMB_X12_Y2_N16	X12Y3S0I6
7	LCCOMB_X12_Y2_N18	X12Y3S0I7
8	LCCOMB_X12_Y2_N20	X12Y3S0I8
9	LCCOMB_X12_Y2_N22	X12Y3S0I9
10	LCCOMB_X12_Y2_N24	X12Y3S0I10
11	LCCOMB_X12_Y2_N28	X12Y3S0I11

Table 1: Correspondence between indices of Fig. 3 and physical resources on the target Cyclone IV device. This list includes only the first LAB. The next 12 indices use the same resources at position X12Y3, and so forth for the remaining 24 indices at positions X12Y4 and X12Y5.

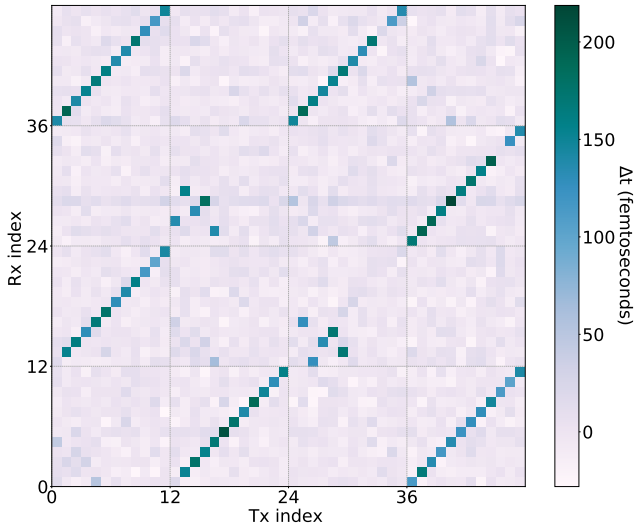


Figure 3: Figure shows the measured value of Δt per C4 wire segment for all pairs of wires in a C4 channel on a Cyclone IV device. See Tab. 1 for explanation of how the indices correspond to physical resources.

for each index because the same measurements are taken using 6 different channels at different columns in the chip and 5 trials for each. This result shows that, regardless of which wire is used for routing a sensitive signal, there exists another wire in the channel with the potential to exfiltrate that sensitive data if used as a covert receiver.

5.2 Comparing Different Long Wire Types

Fig. 5a shows the value of Δt for chains of C4 wires that are combined to create different length transmitter and receiver wires in the three devices. These specific neighboring wires were chosen arbitrarily, but are representative of the typical coupling between neighbors (see Fig. 4). Each line in the plot represents an experiment performed in a single column, and the points on the line correspond to measurements made within that column using different lengths of adjacent transmitter and receiver wires. The experiment is repeated at different columns in the chip to produce the multiple lines.

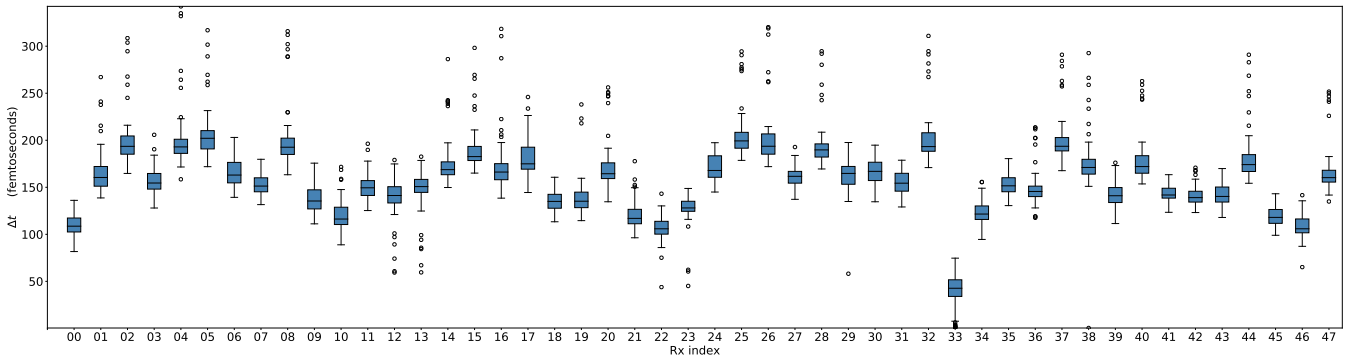
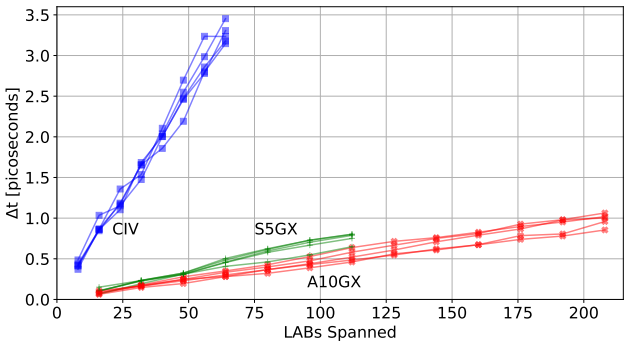


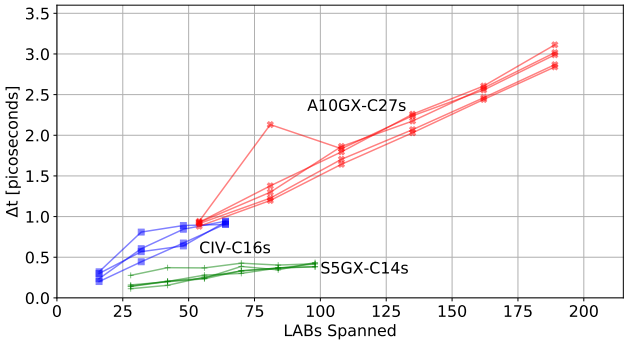
Figure 4: The distribution of observed Δt for each wire in the channel when the wire is used as a receiver and its neighbor is used as a transmitter. In this context, neighbor is defined as the single wire that has the largest impact on the receiver. There is a range of values for each index because the same measurements are taken on 6 different channels at different columns in the chip.

Cyclone IV is measured at locations X12, X36, X60, X84, X100, and X113. Stratix V is measured at locations X12, X50, X108, X171 and X204. Arria 10 is measured at locations X14, X60, X108, X160 and X208. Because the three devices have different numbers of rows, the longest wire that can be created within a column is different for each device. The lengths of wires are given in terms of the number of LABs spanned vertically by the receiver and transmitter. The change in propagation delay on the receiver wire is observed to be linear in the length of the adjacency, so we consider for comparison a single value of Δt/LAB which reflects the slope of the lines in Fig. 5a. We observe values of 47.8fs/LAB, 14.0fs/LAB, and 8.2fs/LAB in Cyclone IV, Stratix V, and Arria 10.

Fig 5b shows an analogous plot to Fig. 5a but using the longer C14, C16, and C27 wires on the devices. Cyclone IV is measured at locations X12, X27, X59, X107 to create the different lines. Stratix V is measured at locations X12, X50, X108, X171 and X204. Arria 10 is measured at locations X14, X60, X108, X160 and X208. The values we observe for Δt/LAB in these cases are 14.6fs/LAB in Cyclone IV, 3.9fs/LAB in Stratix V, and 16.5fs/LAB in Arria 10. The unknown layout strategies that may be employed for each wire type prevents a carefully controlled comparison, yet our results do show that the coupling exists across wires and designs, and that its effect is linear in the length of the adjacent wires.



(a) C4 wires



(b) C14/C16/C27 wires

Figure 5: Measured values of Δt versus length of wire for three different devices. Different lines represent the same wire when measured in different columns across the chip. Each measurement is repeated three times and results are averaged to minimize noise.

5.3 Technology Comparison

Fig. 6 compares the coupling of different long wire types on Cyclone IV, Stratix V, and Arria 10 devices. It should be noted that these devices differ not only in their process technology, but also may have different layout strategies tailored to their technology node and intended market segment. The coupling is given in terms of Δt/LAB as in the previous section, meaning that the given number represents the additional increment by which the receiver is slowed down by the transmitter value for every LAB spanned vertically by the two adjacent wires. If any application requires routing sensitive signals on long wires in which the other wires in the channel are untrusted, this analysis can guide a designer in deciding whether to

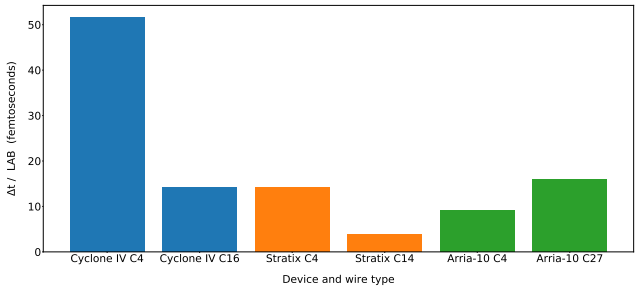


Figure 6: Values of Δt/LAB observed using two different wire types on three different devices. Significant leakage is observed in all devices, and there is not a clear trend across technology nodes.

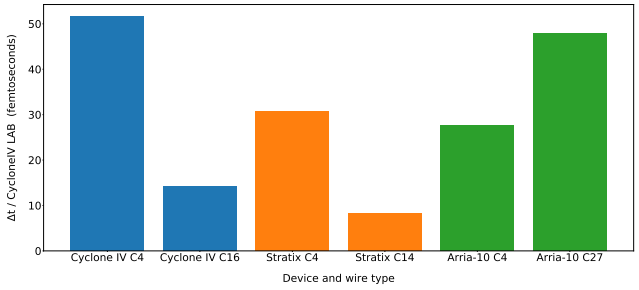


Figure 7: Values of Δt/LAB when normalized to a wire length that matches the height of a LAB on a 60nm technology Cyclone IV device.

use a long sequence of C4 wires, or a reduced number of the longer wire types.

The physical size of each LAB changes with technology node. Therefore, a comparison of coupling per-LAB-span on devices implemented in different technologies is not a fair comparison of coupling per-unit-length. To consider coupling per wirelength in absolute terms, one must adjust for technology scaling. We do this by trying to estimate the amount of coupling on a wire span that is equivalent in length to the LAB height in the Cyclone IV's 60nm technology. Assuming that LAB height scales proportional to minimum feature size of the technology node, then the height of one LAB in the Cyclone IV's 60nm technology is equivalent in height to 2.14 LABs in Stratix V (28nm) and 3 LABs in Arria 10 (20nm). Adjusting by these factors yields the data shown in Fig. 7.

6 CONCLUSION

Previous work shows the existence of coupling between neighboring long wires on both Xilinx and Intel SRAM FPGAs. In this paper we have presented an accurate method for quantifying the amount of coupling that exists between neighboring long wires. Our approach can detect and quantify delay changes on the order of femtoseconds that are caused by the logic value of neighboring wires. We use the method to characterize coupling on FPGAs in three different technology nodes including 20nm technology. We show that coupling between long wires can be used to recover adjacency information from channels if the information is not freely available from the device vendor. Our findings show that the leakage exists and is significant across all the FPGAs tested.

The experimentally measured delay in the examined wires verifies that length is not the only factor contributing to Δt values. Physical design and layout strategies may determine the propagation delay of the wires as well. Nevertheless, the experimental methods in this paper can help designers to quantify the data leakage susceptibility of sensitive signals in their design in order to decide whether mitigation is needed. Leakage can be avoided by disallowing multiple tenants to share use of a single channel, and future work in this direction can analyze how to share channels and maximize utilization while still ensuring that all wires carrying sensitive data are protected from snooping by neighbors.

Acknowledgement

This research was funded by NSF/SRC grant CNS-1619558 and a grant from Intel's Corporate Research Council.

REFERENCES

[1] I. Giechaskiel, K. B. Rasmussen, and K. Eguro. 2018. Leaky Wires: Information Leakage and Covert Communication Between FPGA Long Wires. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security (ASIACCS '18)*. ACM, New York, NY, USA, 15–27.

[2] D. R. E. Gnad, F. Oboril, S. Kiamehr, and M. B. Tahoori. 2018. An Experimental Evaluation and Analysis of Transient Voltage Fluctuations in FPGAs. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 26, 10 (Oct 2018), 1817–1830.

[3] B. Gojman, S. Nalmela, N. Mehta, N. Howarth, and A. DeHon. 2015. GROK-LAB: Generating real on-chip knowledge for intra-cluster delays using timing extraction. *ACM Transactions on Reconfigurable Technology and Systems (TRETS)* 7, 4 (2015), 32.

[4] C. Ramesh, S. B. Patil, S. N. Dhanuskodi, G. Provelengios, S. Pillement, D. Holcomb, and R. Tessier. 2018. FPGA side channel attacks without physical access. In *IEEE International Symposium on Field-Programmable Custom Computing Machines*. 45–52.

[5] F. Schellenberg, D. R. E. Gnad, A. Moradi, and M. B. Tahoori. 2018. An inside job: Remote power analysis attacks on FPGAs. In *Design, Automation & Test in Europe Conference & Exhibition, DATE 2018*.

[6] S. Yazdanshenas and V. Betz. 2018. Interconnect Solutions for Virtualized Field-Programmable Gate Arrays. *IEEE Access* 6 (Feb. 2018), 10497–10507.

[7] H. Yu, Q. Xu, and P. H. W. Leong. 2010. Fine-grained characterization of process variation in FPGAs. In *2010 International Conference on Field-Programmable Technology*. 138–145.

[8] M. Zhao and G. E. Suh. 2018. FPGA-based remote power side-channel attacks. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 229–244.