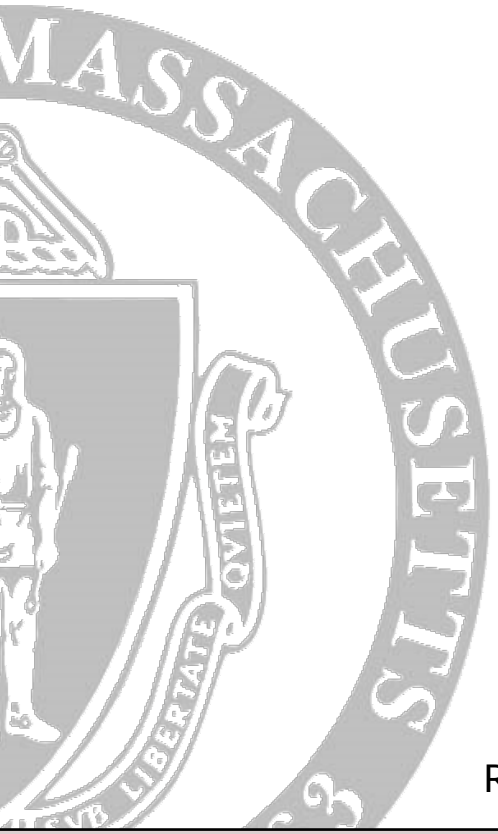
Security and Privacy Concerns for the FPGA-Accelerated Cloud and Datacenters



Russell Tessier, Daniel Holcomb, and George Provelengios
Electrical and Computer Engineering
University of Massachusetts, Amherst

May 6, 2020

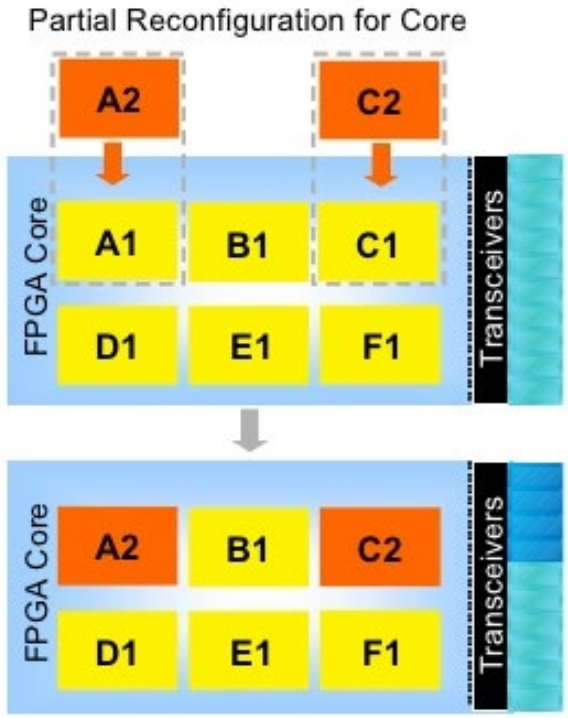
Research funded by the Intel Research Council and NSF grant CNS-1902532

Overview

- Background
  - FPGAs in the cloud
  - Multi-tenant FPGAs
  - FPGA voltage attack approaches
- Characterizing voltage attacks on Arria 10
  - Experimental approach
  - Characterization test results
  - Fault induction
- RSA attack using power fluctuation on Cyclone V and Arria 10
  - Induce delay faults in RSA
  - Use Chinese remainder theorem to extract key

Multi-Tenant FPGA

- Shared (multi-tenant) FPGAs
  - Devices are expensive. Desire to fully use resources
- Cloud computing: target for multi-tenant FPGAs?
  - Why not use partial reconfiguration?
  - User has no idea what "neighbor" is doing (side channels)
  - Don't want to risk leaking information
- Need to understand vulnerabilities
  - Previous: temperature, voltage
  - This work: no physical access needed

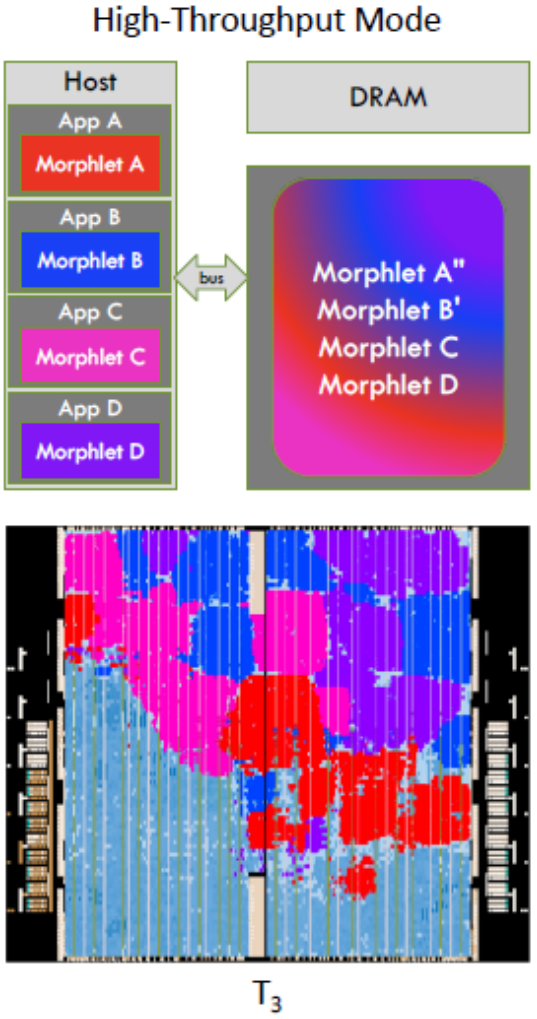


Source: Intel[1]

[1] Stratix V FPGAs: Built for Bandwidth, Intel Corporation, 2010

Example: AmorphOS for Amazon EC2 F1

- Multiple users deploy circuits (Morphlets) on FPGA
- Virtualizing software multiplexing software and memory interfaces
- Attempt to create "virtual machine" like environment on the FPGA
- Increase income level for hardware use.
- Security?: not really a focus

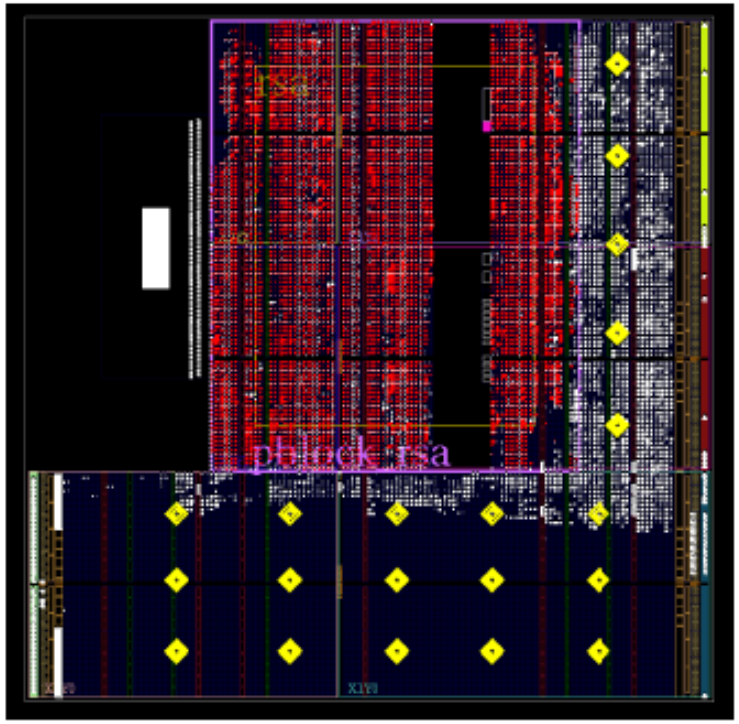


Source: Khawaja et al[1]

[1] A. Khawaja et al., Sharing, Protection, and Compatibility for Reconfigurable Fabric with AmorphoS, OSDI, Oct. 2018

What Type of FPGA Voltage Attacks are Possible?

- On-chip voltage sensors to extract encryption key
  - Ring oscillators used to extract RSA key[1]
  - Time-to-digital converters used to extract AES key[2]
- Voltage fluctuation-based communication
  - Communication on single FPGA[3,4]
- On-chip voltage supply attacks
  - Induce stealthy faults[5, 6, 7]
- Drive FPGA into reset[7]



FPGA voltage sensors surrounding RSA core[1]

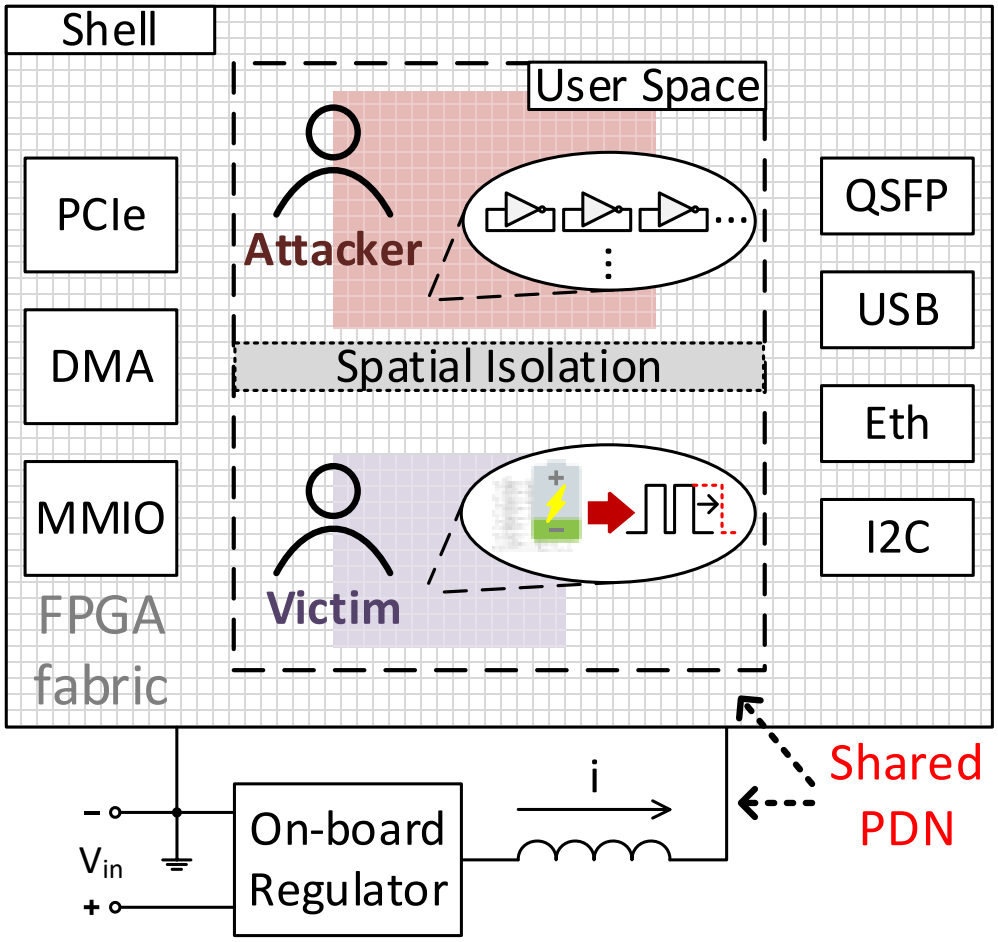
[1] Zhao and Kuh, FPGA-Based Remote Power Side-Channel Attacks, IEEE Symp. Security and Privacy, May 2018
[2] Schellenberg et al, An Inside Job: Remote Power Analysis Attacks on FPGAs, DATE, March 2018
[3] Gnad et al, "Voltage-based covert channels in multi-tenant FPGAs," Cryptology ePrint Archive, vol. Report 2019/1394, 2019
[4] Giechaskiel et al., "Reading between the dies: Cross-SLR covert channels on multi-tenant cloud FPGAs" ICCD, Oct. 2019
[5] Krautter et al, FPGAhammer: Remote Voltage Fault Attacks on Shared FPGAs, suitable for DFA on AES, CHES, vol 3 , 2018
[6] Mahmoud and Stojilovic, "Timing violation induced faults in multi-tenant FPGAs," in DATE 2019
[7] Provelengios, "Characterizing Power Distribution Attacks in Multi-User FPGA Environments", FPL 2019

Overview

- Two tenants are using simultaneously the device
- Tenant A (attacker) consumes power aggressively in an attempt to induce timing faults in tenant B (victim)

Threat model:

- ✓ Tenants are spatially isolated but share the FPGA power distribution network (PDN)
- ✓ Tenants do not have physical access to the board
- ✓ The tools used for interacting with the FPGA are secure

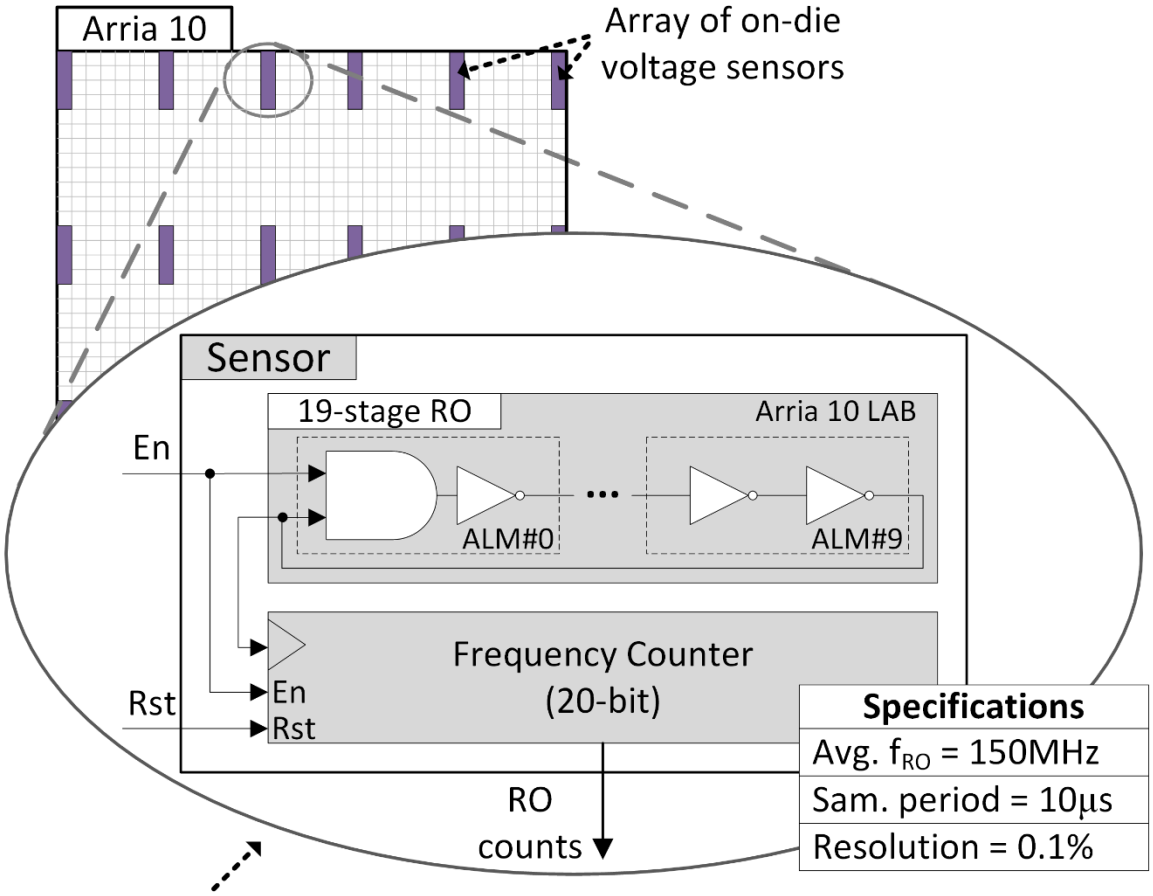


Contribution

- We investigate on-chip voltage attacks and specifically how their impact depends on:
  - Duration of voltage disruption
  - Consumed power by attacker
  - Distance between attacker & victim
- We evaluate the ability of power wasting circuits to induce timing faults to victim
- We examine the ability of power wasting circuits to reveal an RSA encryption key through fault injection

Voltage sensor architecture

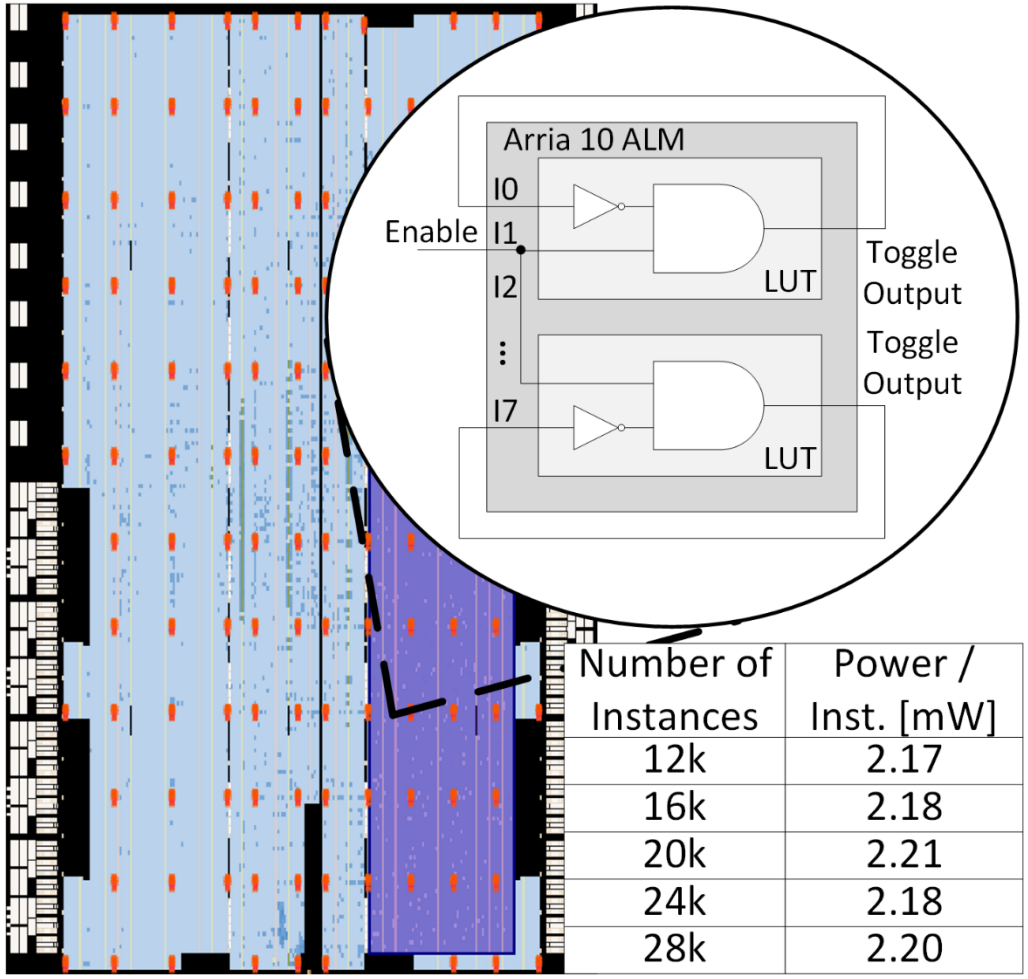
- A regular rectangular grid of 46 sensors
- 19 inverting stages:
  - ✓ Meet timing constraints
  - ✓ Minimize local effects[1]
  - ✓ Fit in a single CV LAB
- Arria 10 parameters
  - f_{RO}=150 MHz
  - Samp. period = 10µs



[1] M. Barbareschi, G. Di Natale, and L. Torres, "Implementation and analysis of ring oscillator circuits on Xilinx FPGAs," in *Hardware Security and Trust.* N. Sklavos, R. Chaves, G. Di Natale, and F. Regazzoni, Eds. Springer, 2017, ch. 12, pp. 237-251

Attacker circuitry

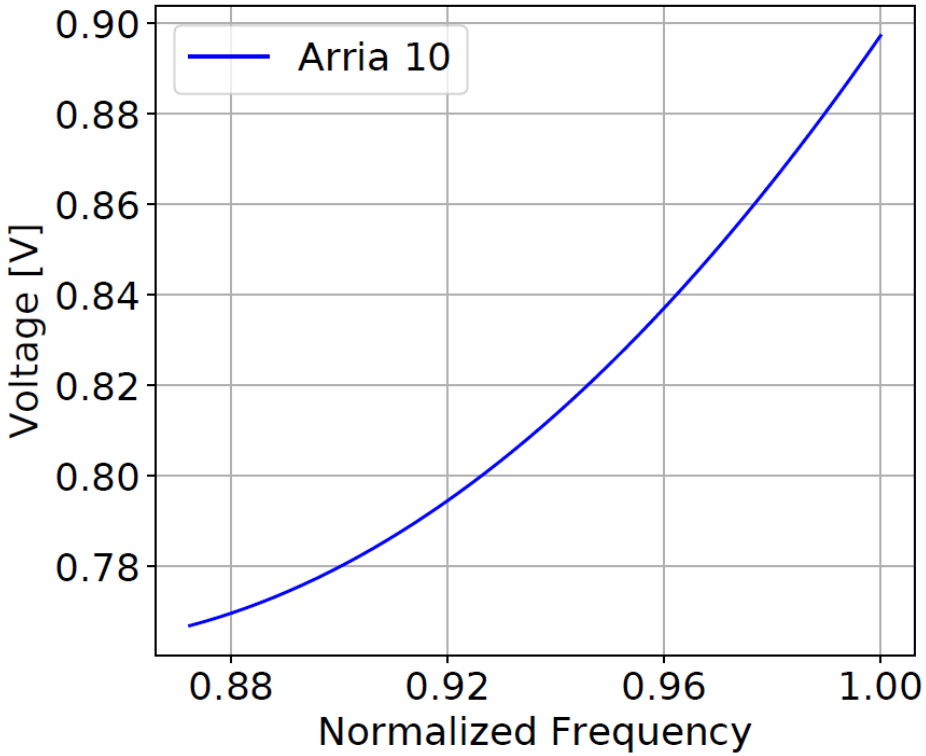
- $P_{dyn} = C \times V_{DD}^2 \times f_{SW}$
- 1-stage ROs as power wasters
- Arria 10: 11,424 LABs fit up to 28K PW
- Placed uniformly at random locations in the attack area



Number of Instances	Power / Inst. [mW]
12k	2.17
16k	2.18
20k	2.21
24k	2.18
28k	2.20

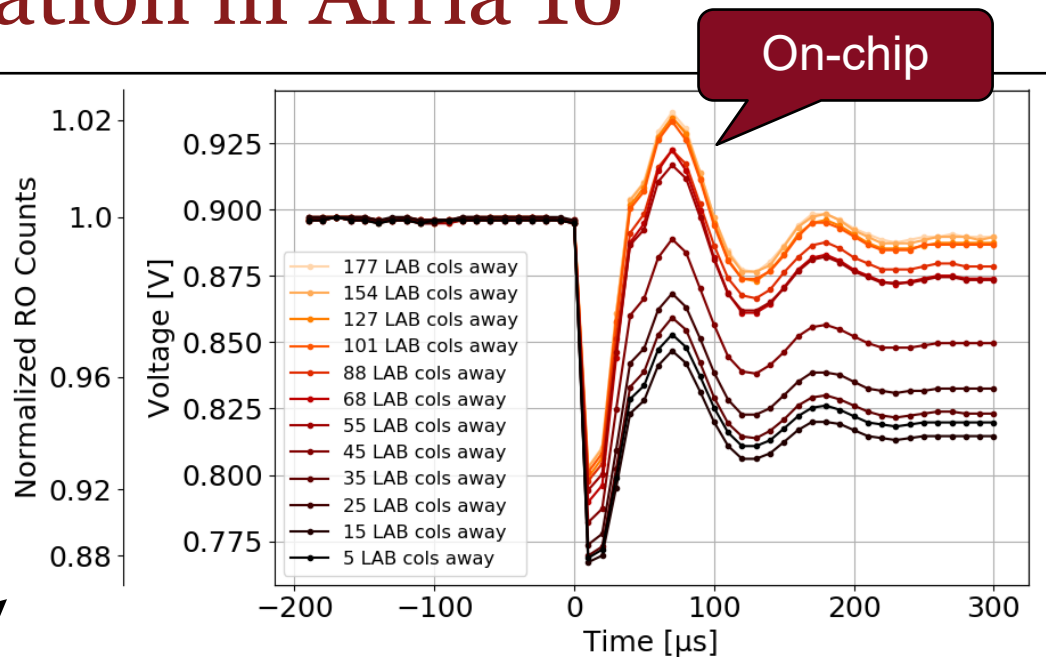
Arria 10 sensor calibration

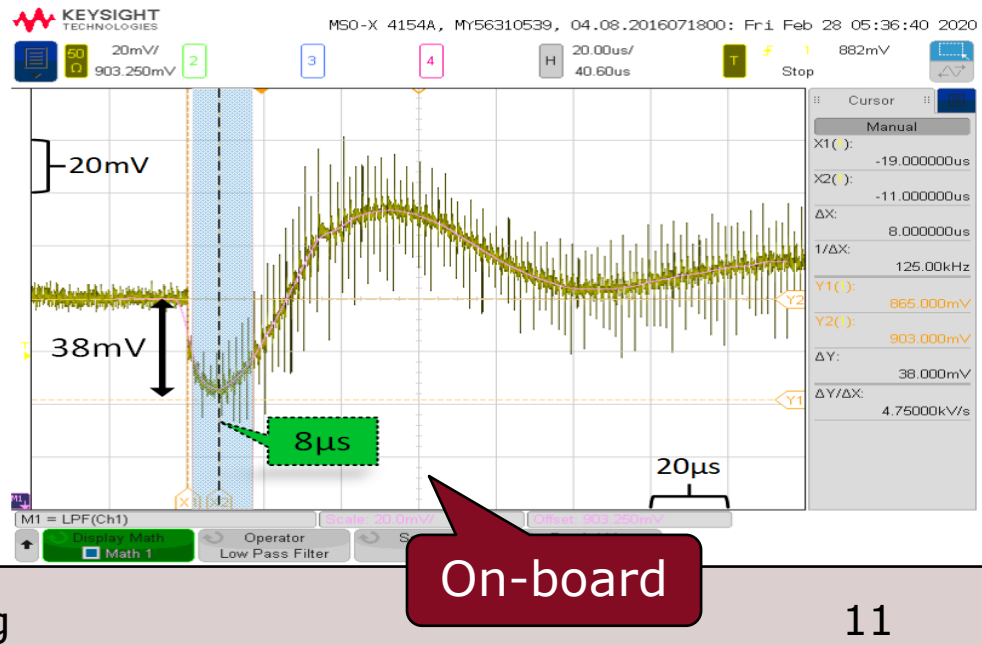
- To use ROs as on-chip voltage sensors:
  - Vary power waster count between 8,000 and 28,000 and record:
    - ✓ Voltage on on-chip sensor
    - ✓ RO counts from on-chip sensors
- Minimize the power drawn by the FPGA during measurements



Voltage drop characterization in Arria 10

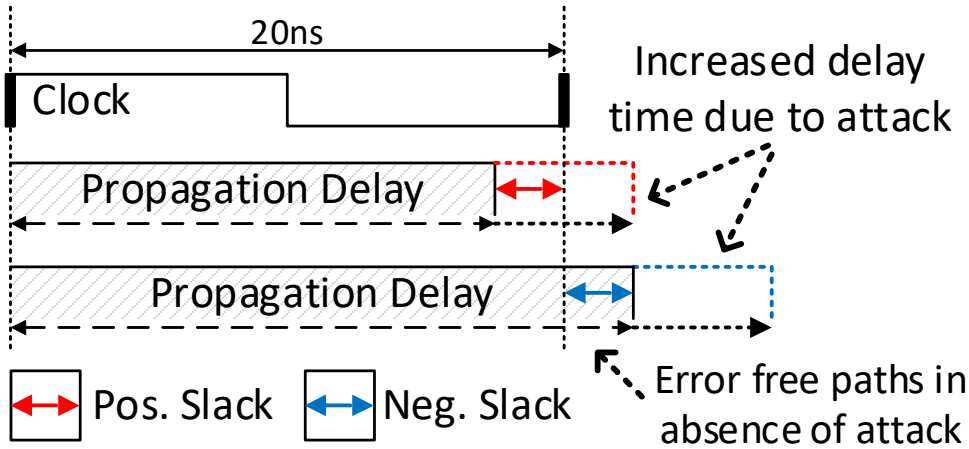
- Evaluate the Arria 10 PDN response
- 28k RO-based PW instances
- 12 on-chip sensors at different distances to the center of the waster
- Peak voltage drop ~8us after activating PWs

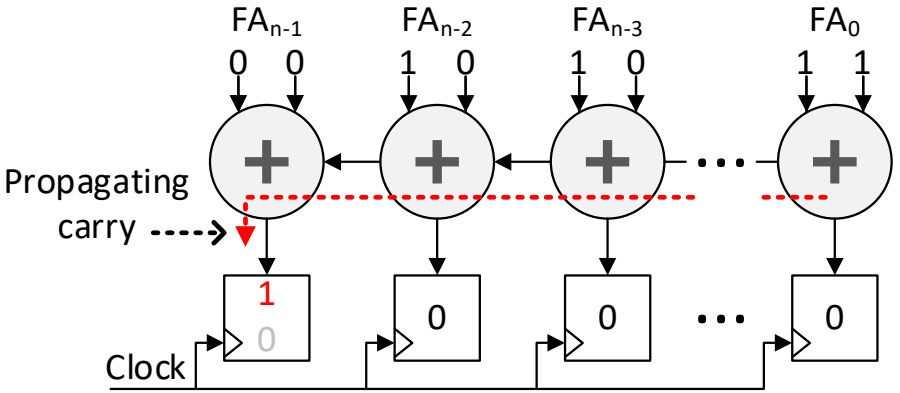




Characterizing timing faults

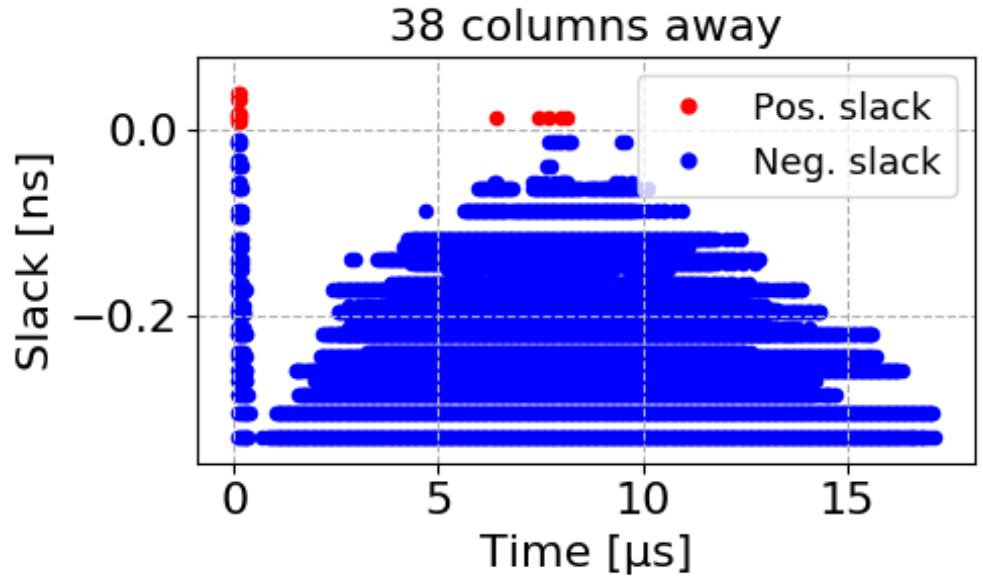
- Voltage drop causes delay of combinational logic to increase
- Wrong values captured if paths do not complete before capturing clock edge arrives
- Must overcome conservative timing models
- Use ripple carry adder as a representative test circuit which allows us to sensitize various path lengths

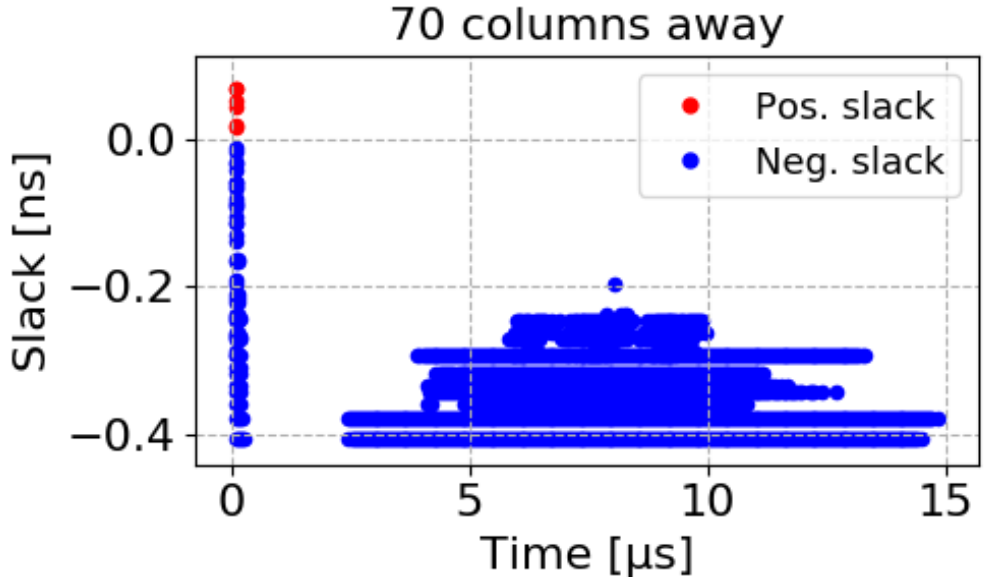




Arria 10 timing faults

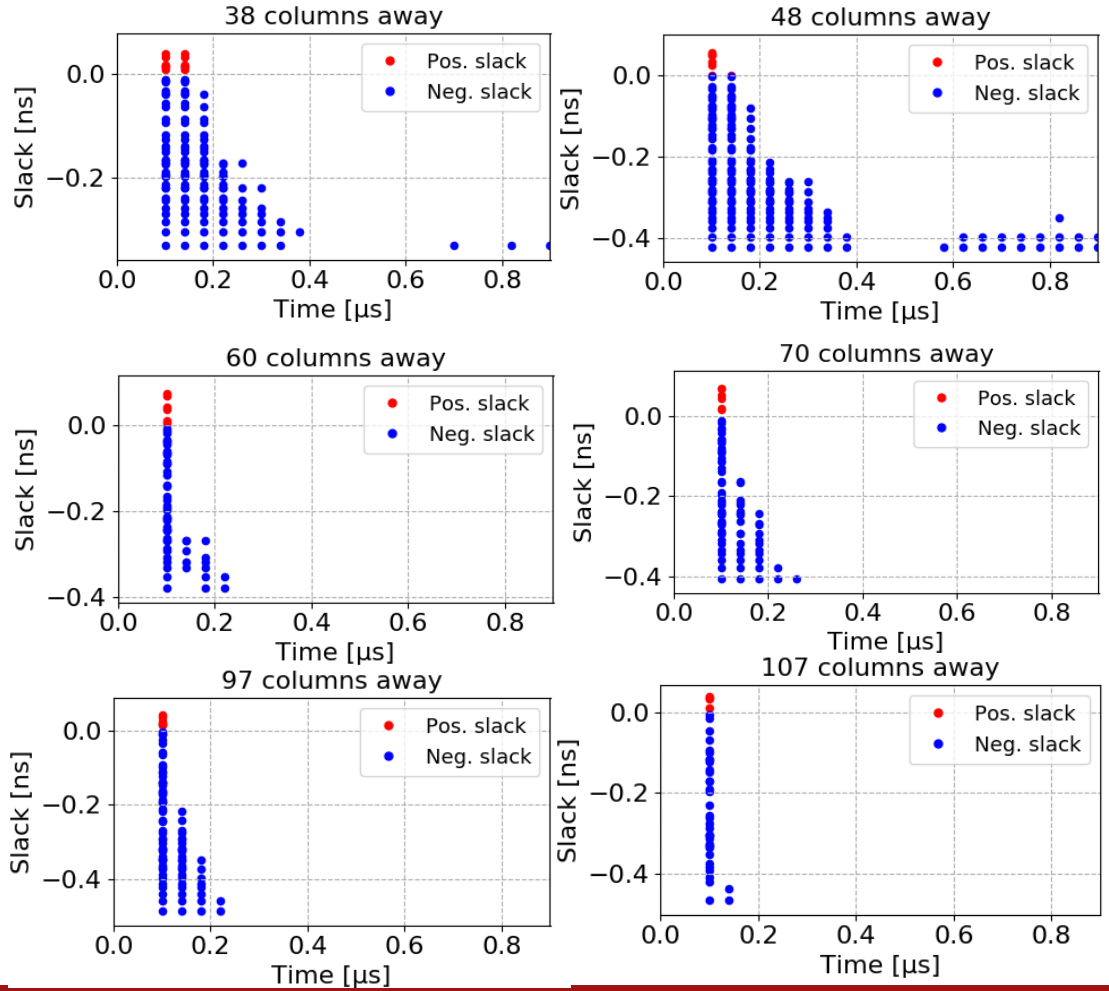
- 28k PWs randomly placed in an area of 11,424 LABs (168x68)
- Steep voltage drop at 20 ns induces faults
- Faults peak at 8 µs
- Substantially fewer faults than Cyclone V

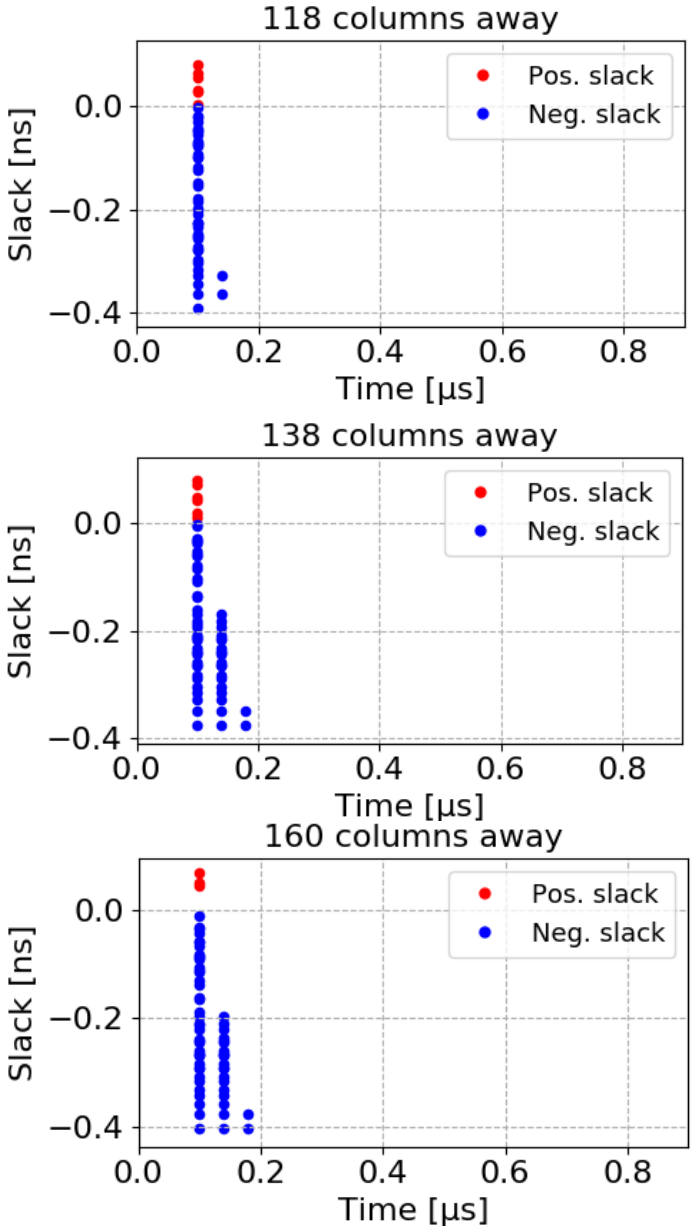




Can a victim evade the attack?

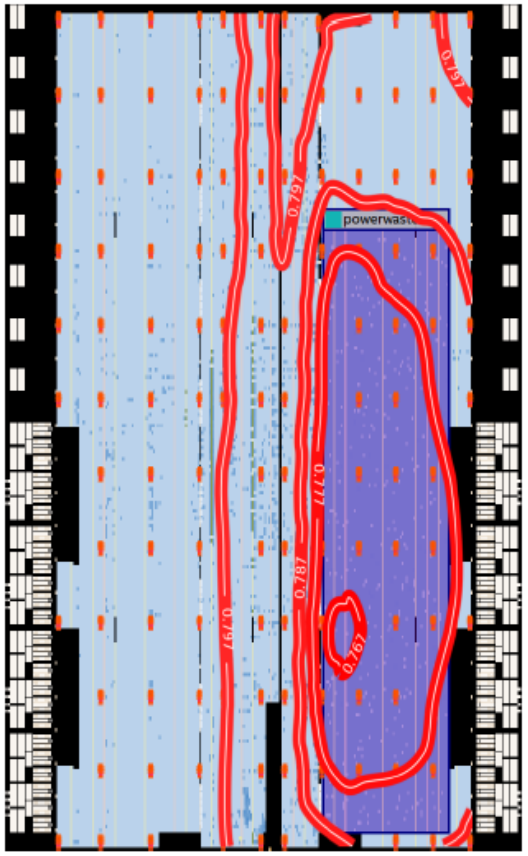
- In Arria 10, the initial fast voltage drop is not location dependent
- Faults from legal paths reported even at the edge of the device



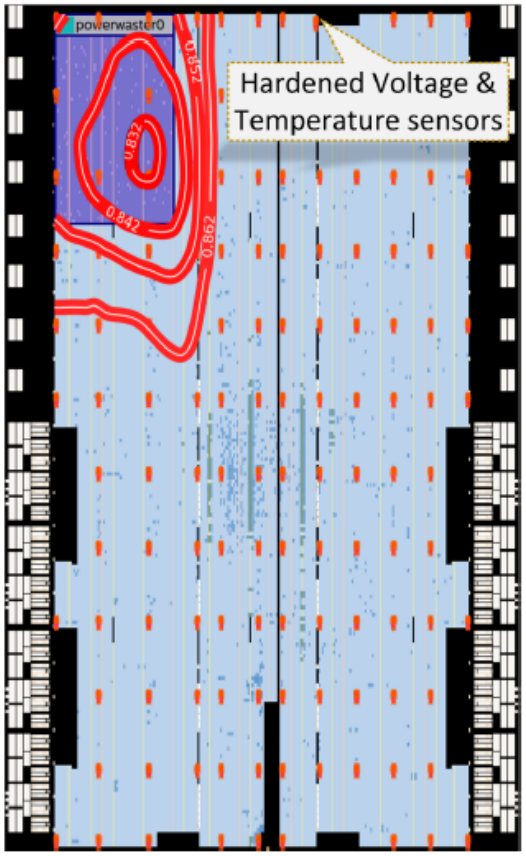


Mapping the Arria 10 voltage drop

- Using 132 on-chip sensors for deriving the voltage contours
- Varying the magnitude of disturbance and location of attacker
- Center of attack:
  - 28K PWs: 767mV
  - 8K PWs: 862mV
- Upper right corner of the chip:
  - 28K PWs: 797mV



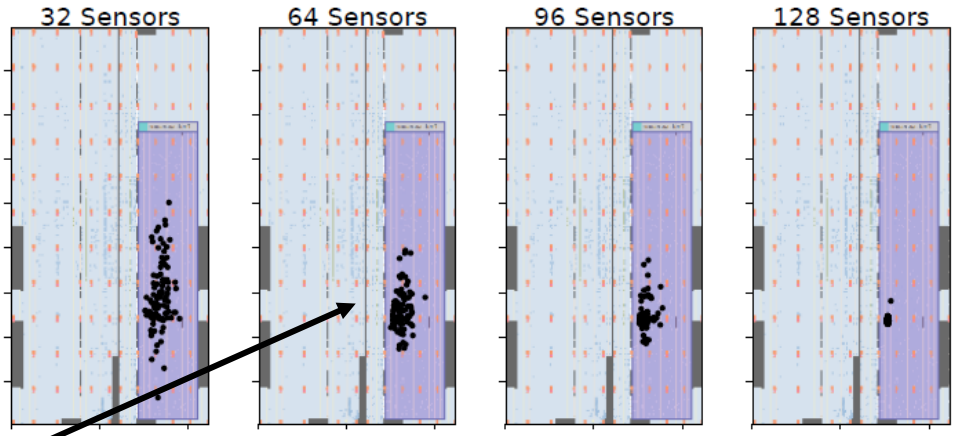
(A) 28K power waster attack



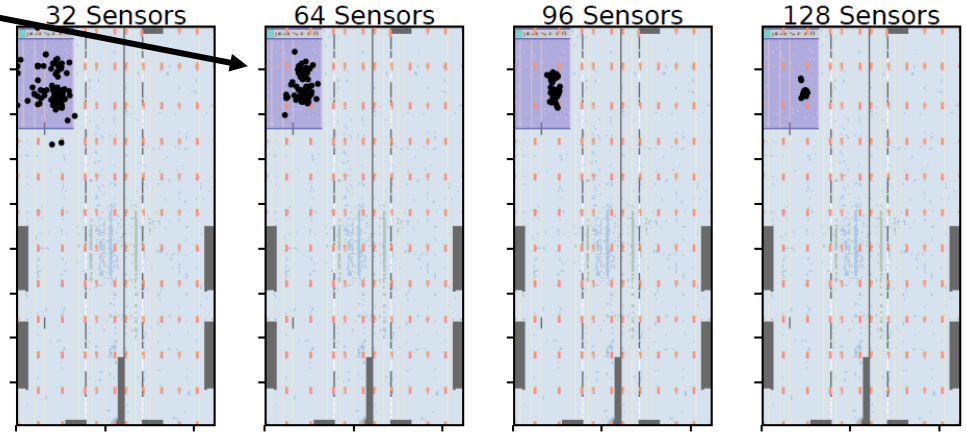
(B) 8K power waster attack

Locating the Arria 10 attack area

- The disturbance of the shared PDN reveals the location of the attacker
- Evaluate how many sensors required to find its location
- 64 sensors are sufficient to identify the attacker



(A) 28K power waster attack

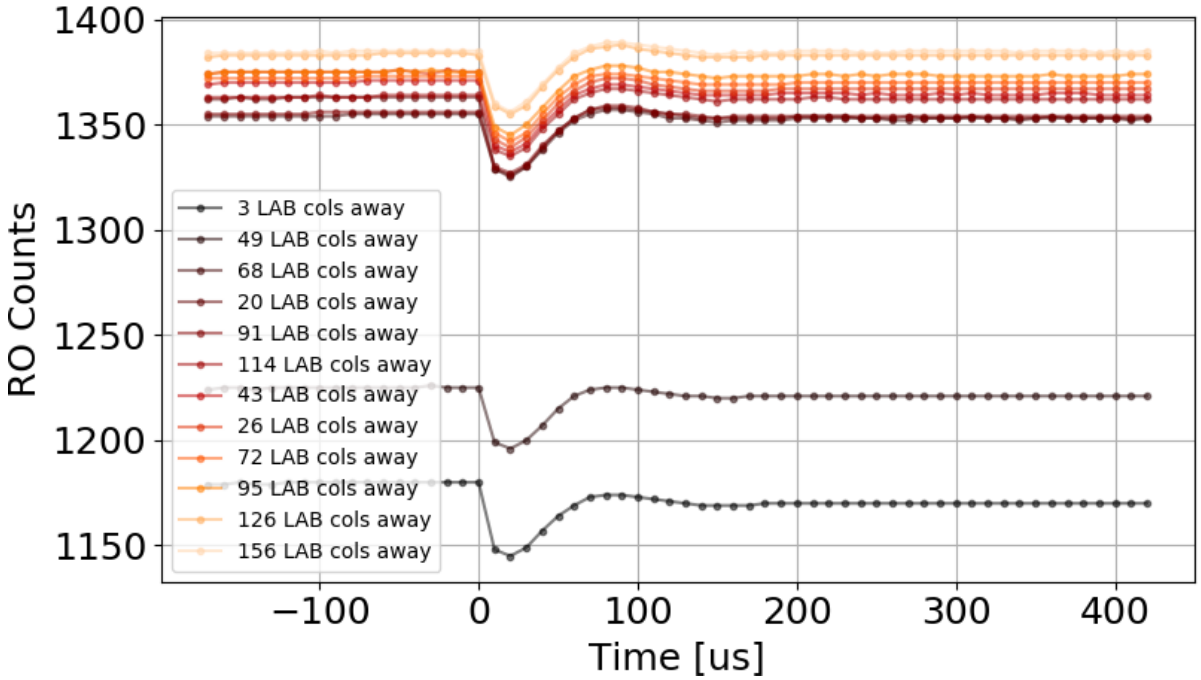


(B) 8K power waster attack

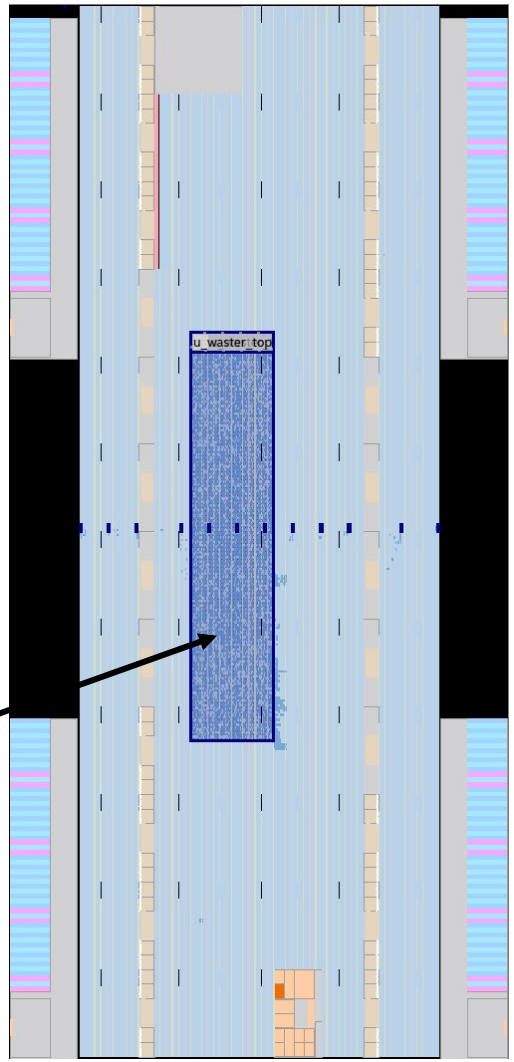
Resource utilize.: Arria 10AX115N2F45E1SG

Num. RO Sensors	ALMs (Avail.:427,200)	Flip-flops (Avail.: 1,708,800)
64	1,280 (<1%)	1,280 (<1%)
132	2,640 (<1%)	2,640 (<1%)
Controller	1,008 (<1%)	134 (<1%)

Preliminary Results with Stratix 10 on DE10-Pro



- 12 ring oscillators located in middle of device
- 10,000 power wasters
- Sampling period of 10 µs
- Ongoing: translating RO count to voltage



Stratix 10 1SX280HU2F50E1VG

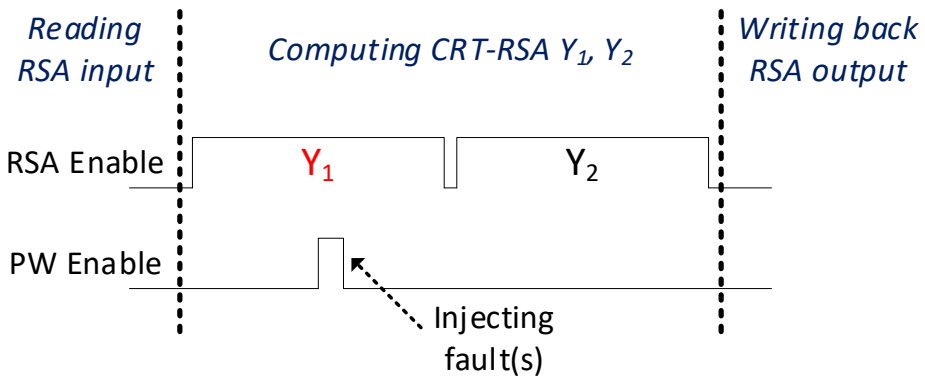
Attacking RSA through fault injection

- Exploiting the use of the Chinese Remainder Theorem (CRT) [1]:

Direct RSA	*RSA with CRT (4x faster)*
$Y = X^e \bmod N$	$Y = aY_1 + bY_2$ $Y_1 = (X \bmod p)^{e \bmod (p-1)} \bmod p$ $Y_2 = (X \bmod q)^{e \bmod (q-1)} \bmod q$

X, Y	Input, Output
e	Priv. key exponent
N	n-bit Modulus
p, q	n/2-bit Primes
a, b	Constants

- Goal: Inject fault(s) while computing Y_1 or Y_2



- Fault during CRT reveals key
  - Output Y is assembled with a faulty Y_1
  - Prime number q is revealed
  - Private key e can be reconstructed
  - e can also be extracted with a faulty Y_2

➢ The attack works for any key length

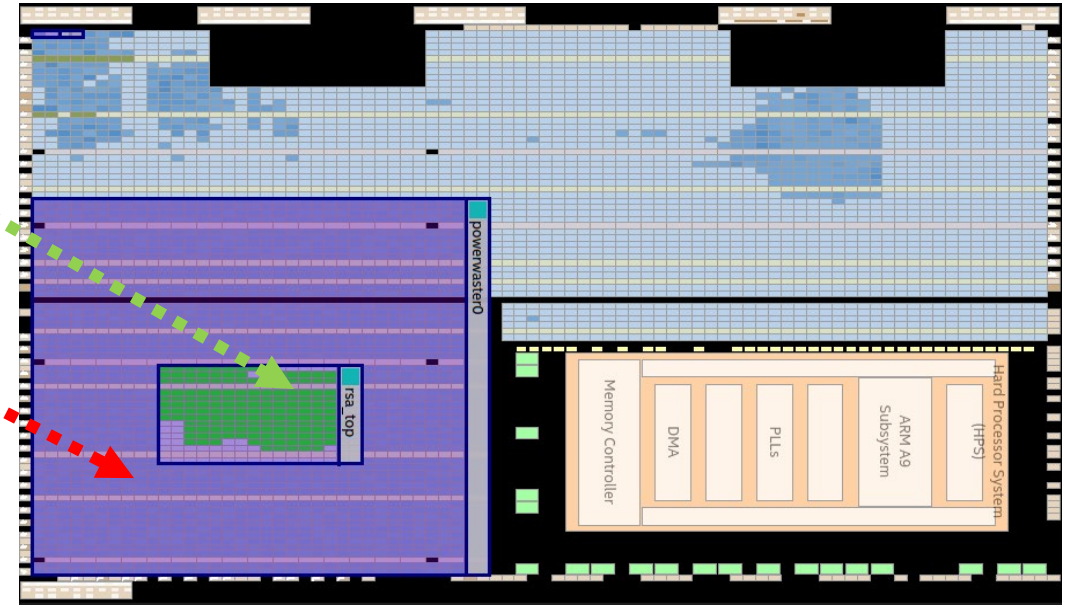
➢ A single interaction is sufficient [2]

[1] D. Boneh et al., On the Importance of Eliminating Errors in Cryptographic Computations, Journal of Cryptology, 2001

[2] A.K Lenstra, Memo on RSA signature generation in the presence of faults, 1996

RSA experimental setup

- 128-bit RSA implementation is placed in an area of 256 LABs
- Wasters are placed at random locations around the RSA core covering an area of 1,940 LABs
- A script running on host PC is responsible for controlling the experiment



(Quartus Prime 17.1 - ChipPlanner)

Resource utilization: Cyclone V 5CSEMA5F31C6

RSA core	ALMs (Avail.: 32,070)	Flip-flops (Avail.: 128,280)	Memory [Kb] (Avail.: 3,970 Kb)	F_{max} [MHz]
128-bit	1,236 (3.9%)	1,925 (1.5%)	16	94.74

Extracting the RSA private key

```
lab@lab:~/project/$ make rsa_attack B=builds/build_2020-02-15_04-48-53

Resetting RSA core and setting test parameters ...

Starting experiment ...

Turn on 10900 wasters at time 0.00000183

Waiting for 1 seconds ... 1, done

[mif_parser] parsing extracted MIF and dumping to file(s) ... done

Retrieving Y1 and Y2 RSA outputs ... done

******** Test Completed ********
RSA test        : FAIL
Key length      : 128 bit
RSA core output: 0x35f3ad9c676ffcedbd2b5d621ed718b9
Expected output: 0x34bc3c6b59eaf60fcb488411e3fdbae2

[extract_rsakey] RSA output: 0x35f3ad9c676ffcedbd2b5d621ed718b9

[extract_rsakey] Extract the first prime number using the faulty
output (p1 = gcd(x - y^pub_exp, N)):
p1 = 0x8cabce4e903ea111

[extract_rsakey] Factor modulus N (p2 = N/p1) to get the second prime number:
p2 = 0xdb53798e1cb036bd

[extract_rsakey] Calculate phi ((p1-1)*(p2-1)):
phi = 0x7884d7fc6471889a7e87ffab9ef7a7c0

[extract_rsakey] Calculate the priv. exponent (pr_exp = (1/pub_exp) mod phi):
Extracted priv. exponent = 0x710c43115c59495105ea53342309a0a9

[extra
[extract_rsakey] ... done. Elapsed time is 0:00:00.000702

lab@lab:~/project/$
```

timing fault(s) occurred!

Original key: 0x710c43115c59495105ea53342309a0a9

- Hardware computes Y_1 and Y_2 while wasters are turned on
- Reading out Y_1, Y_2 and assembling final output Y
- Y does not match the expected output
- Extract the first prime
- Factor N to get the second prime
- Reconstruct the private exponent
- ✓ Extracted and original keys match

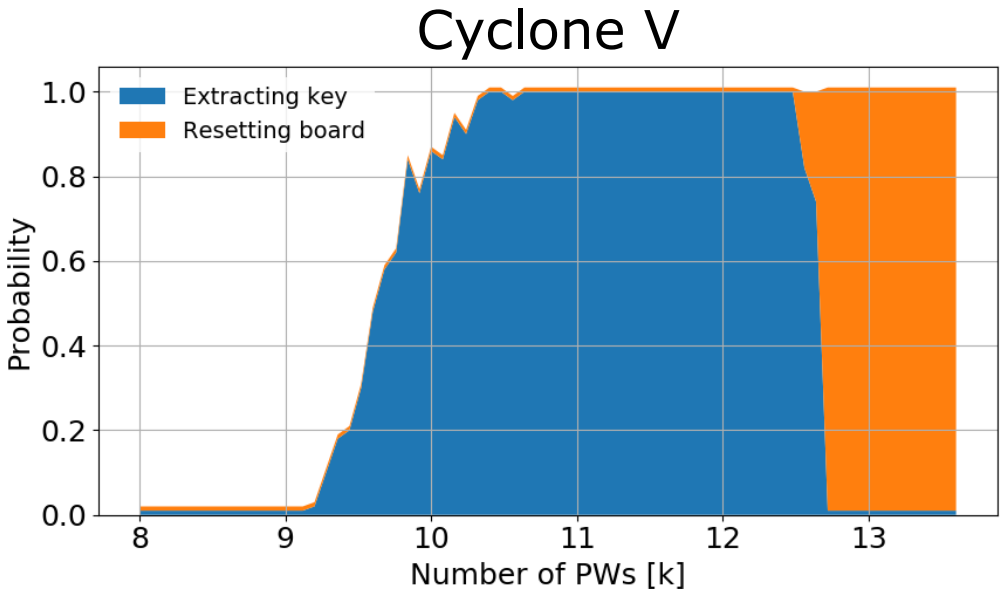
How many wasters are required?

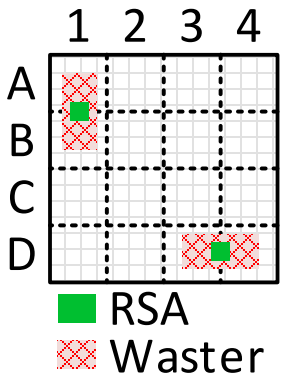
- Vary the number of wasters and find the probability of extracting the key
- Cyclone V:
  - 11K-12K PWs: high chance of extracting the key undetected
  - F_{max}: 94.74MHz, F_{break}: 166MHz (w/o wasters)

~4.5ns margin!

- Identifying weak spots in Arria 10 based on:
  - Number of PWs that can safely be activated
  - Yield in less timing margin

work in progress





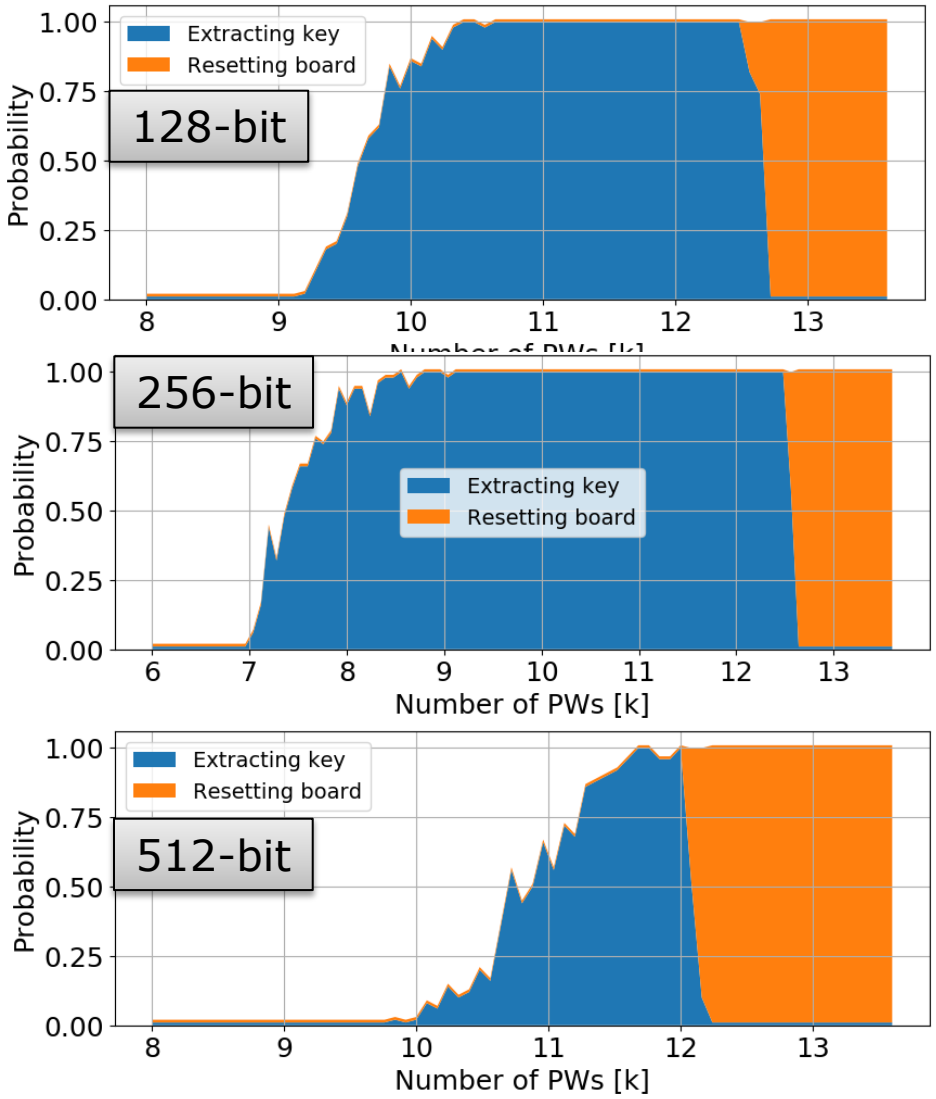
Arria 10

Loc.	PWs [k]	F_{max} [MHz]	F_{break} [MHz] (**w/ wasters**)	F_{break} [MHz] (w/o wasters)
A1, B1	20	179	250	270
A1, B1	24	162	262	281
A1, B1	28	165	240	268
D3, D4	30	103	123	137

How many wasters are required? (cont'd)

- Vary the number of wasters and find the probability of extracting the key
- 11K-12K PWs: high chance of extracting the key undetected
- F_{max}: 94.74MHz, F_{break}: 166MHz (w/o wasters)

~4.5ns margin!



Summary

- Multi-tenant FPGAs
  - Logical next step for cloud computing
- Voltage based attacks
  - Easy to create power wasting circuits that induce faults or crash FPGA
- Characterizing voltage attacks on Arria 10
  - 15% core voltage drop within 8 us
  - Induces faults throughout device
- RSA attack
  - Single fault sufficient to expose key
  - Effective for Cyclone V (even defeats built in timing margin)
  - Effective for Arria 10 if design is overclocked