27th Asia and South Pacific Design Automation Conference (ASP-DAC) 2022

FORTIFY: Analytical Pre-Silicon Side-Channel Characterization of Digital Designs

Lakshmy A V, Indian Institute of Technology Madras, avlakshmy@gmail.com

Chester Rebeiro, Indian Institute of Technology Madras, chester@cse.iitm.ac.in

Swarup Bhunia, University of Florida, swarup@ece.ufl.edu

Power Side-Channel Attacks (PSCA)

The instantaneous power consumption patterns of an electronic device may indirectly reveal the data being processed or the operations being performed by the device.

Steal passwords	Reverse engineer software	Record key presses	Monitor web browsing activity

Power Side-Channel Vulnerability Estimation

Post-Silicon Techniques

Manufactured devices

Collect power traces from device → Analyze using statistical metrics

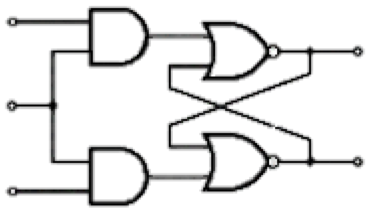
- Meets security requirements
- Does not meet security requirements

Provide accurate side-channel vulnerability estimation

Too late in the design cycle to take any corrective measures

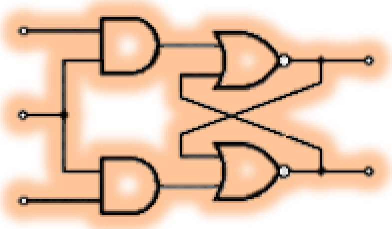
Power Side-Channel Vulnerability Estimation

Pre-Silicon Techniques



H/W design (RTL/netlist)

Behavioral simulation → Power models → Statistical metrics



Side-channel leakage scores

- Provide an early & fine-grained estimate of side-channel leakage
- Less accurate than post-silicon; Require large no. of simulations

Our Key Idea

A signal leaks more information if its values have a high correlation with the reference signal.

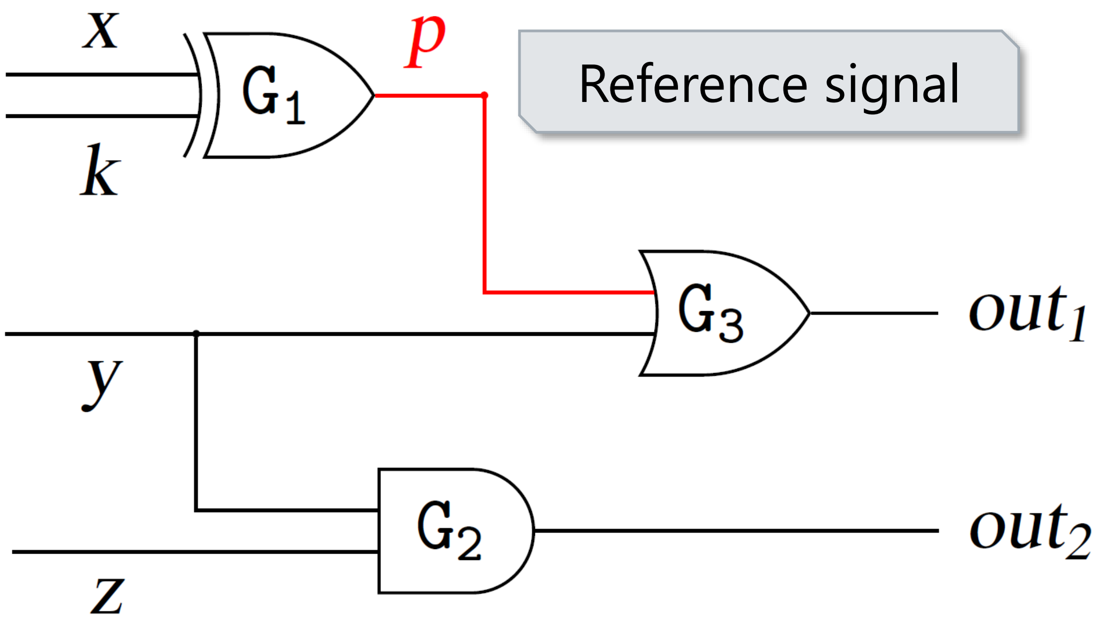
Signal Probability

$$SP(sig) = Pr(sig = 1)$$

Conditional Signal Probability

$$SP_0(sig, A) = Pr(sig = 1 \mid A = 0)$$
$$SP_1(sig, A) = Pr(sig = 1 \mid A = 1)$$

where A is the reference signal



Our Key Idea

A signal leaks more information if its values have a high correlation with the reference signal.

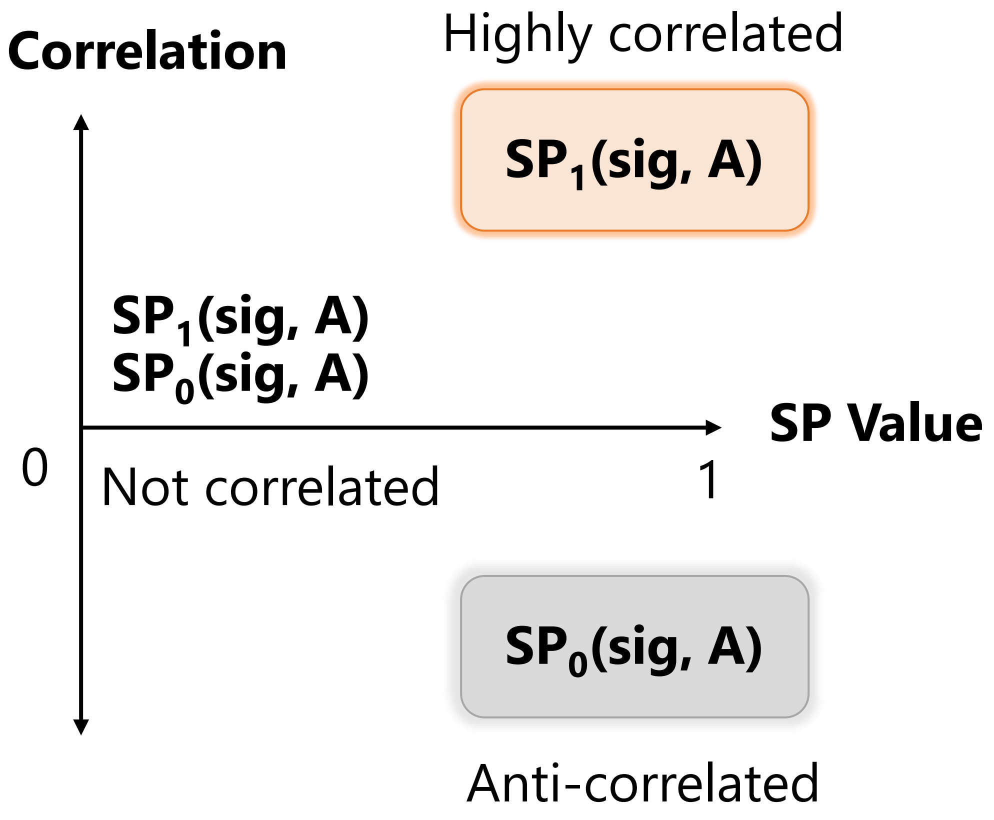
Signal Probability

$$SP(sig) = Pr(sig = 1)$$

Conditional Signal Probability

$$SP_0(sig, A) = Pr(sig = 1 \mid A = 0)$$
$$SP_1(sig, A) = Pr(sig = 1 \mid A = 1)$$

where A is the reference signal



FORTIFY

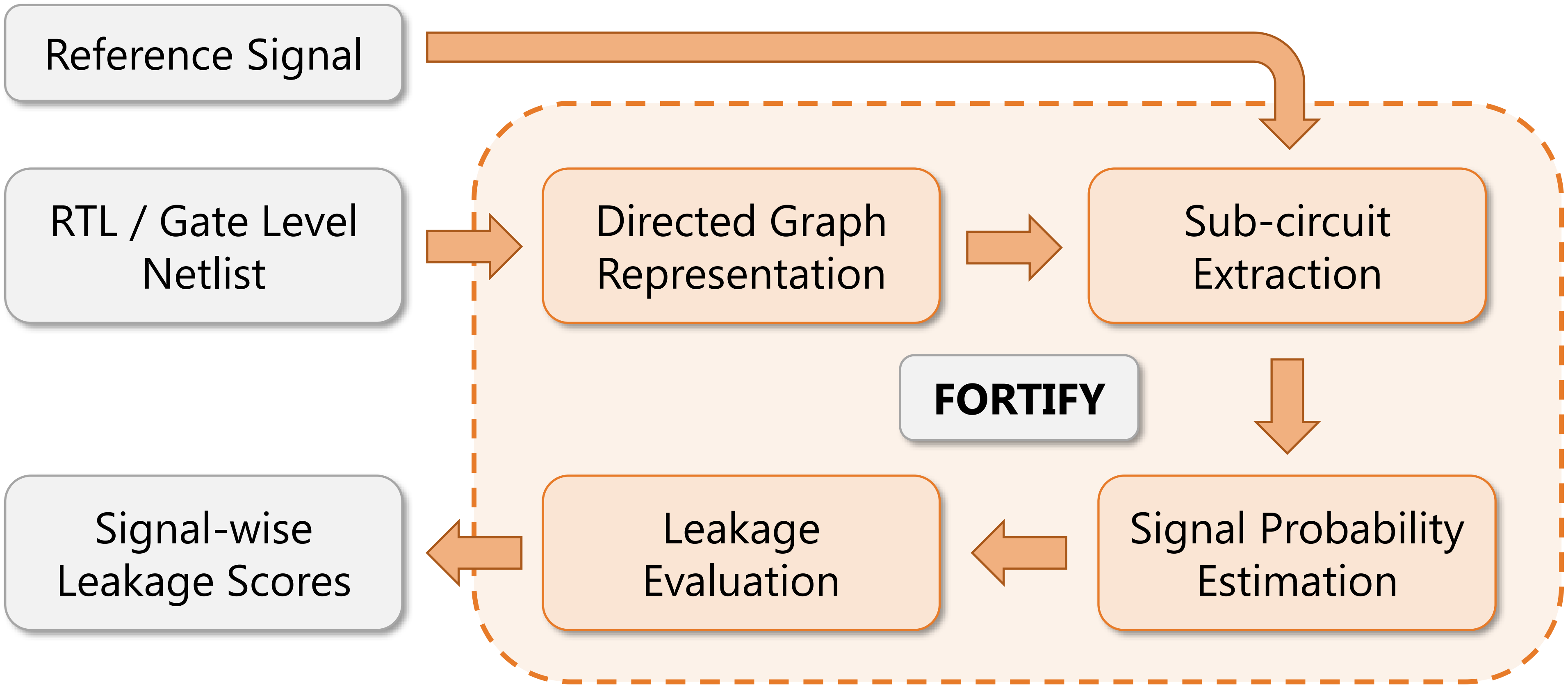
To provide a **quick, fine-grained estimation** of the **power side-channel vulnerability** of **pre-Silicon digital circuit designs**

Analytical approach, without involving lengthy simulations

Signal Probability Correlation Factor (SPCF) metric

Accurate; scalable to evaluate large designs

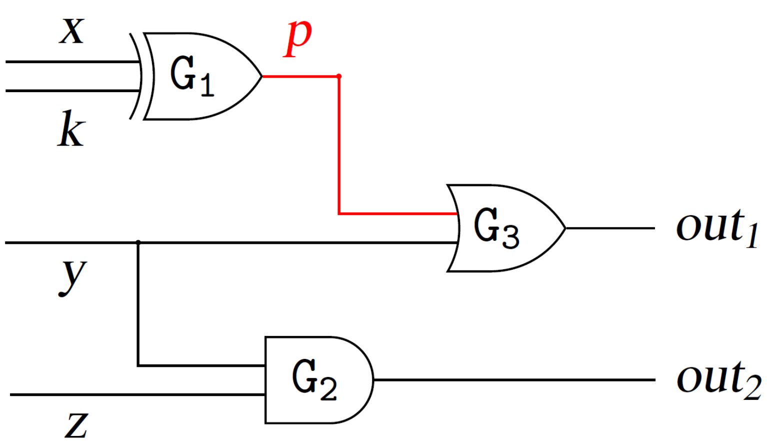
An Overview of FORTIFY

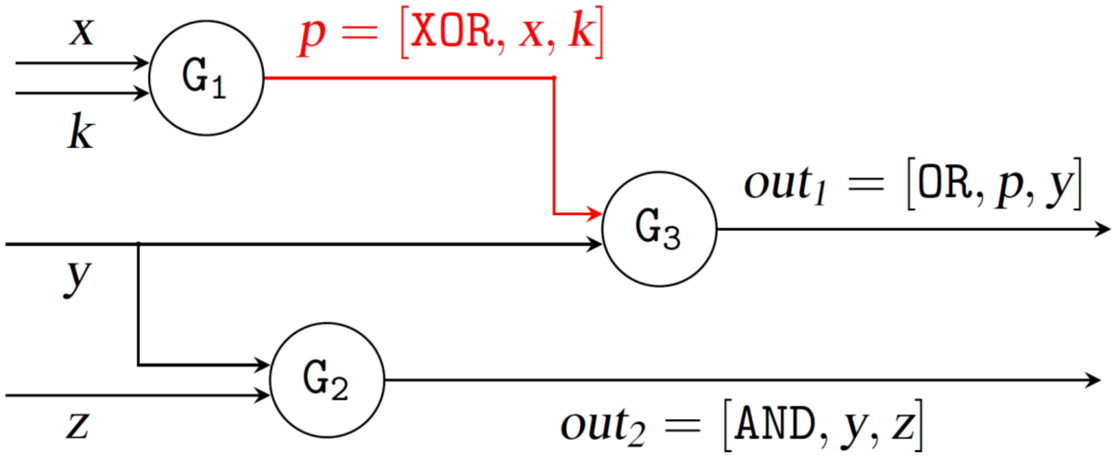


Directed Graph Representation

Convert the input digital circuit design into a directed graph representation

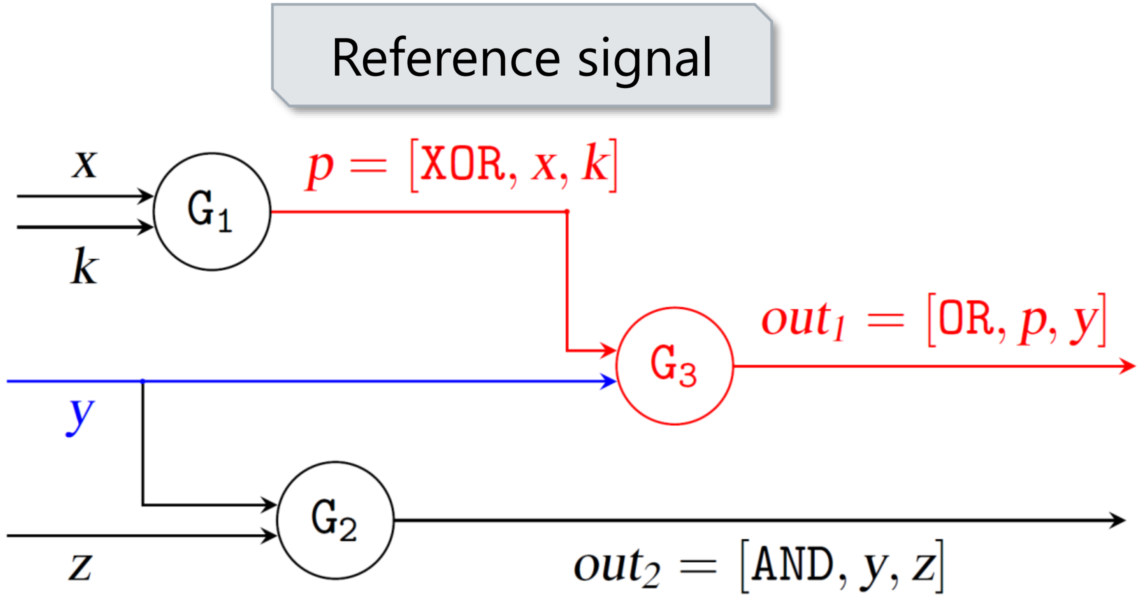
- Nodes: Logic Gates
- Edges: Signals
- Edge labels: Logical expressions





Sub-circuit Extraction

Extract the sub-circuit of the input design influenced by the reference signal

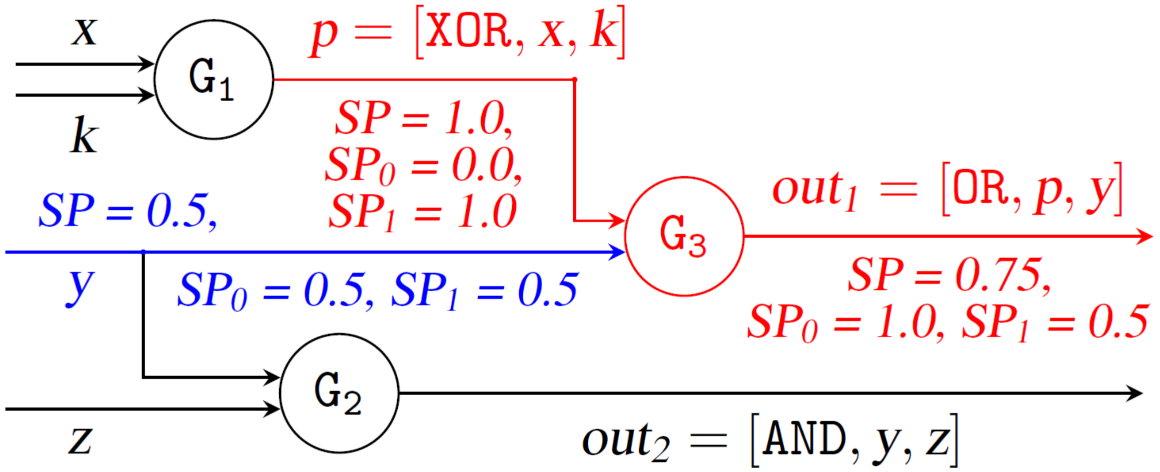


Gates and signals reached by reference signal

Other inputs feeding into the reachable gates

Signal Probability Estimation

Estimate signal probabilities, conditional signal probabilities w.r.t reference signal



Incremental Signal Probability Calculation

Logical Expression	Signal Probability
Input A	a
Input B	b
NOT (A)	1 – a
AND (A, B)	ab
OR (A, B)	a + b – ab
XOR (A, B)	a + b – 2ab

Leakage Evaluation

Signal Probability Correlation Factor (SPCF): Metric to estimate leakage

For a 1-bit signal

$$L_A(sig) = \frac{[\mathbf{SP}_1(sig, A) - \mathbf{SP}_0(sig, A)]^2}{2 \cdot \sqrt{\mathbf{V}(sig) \cdot (1 - \mathbf{V}(sig))}}$$

$$\mathbf{V}(sig) = 2 \cdot \mathbf{SP}(sig) \cdot (1 - \mathbf{SP}(sig)).$$

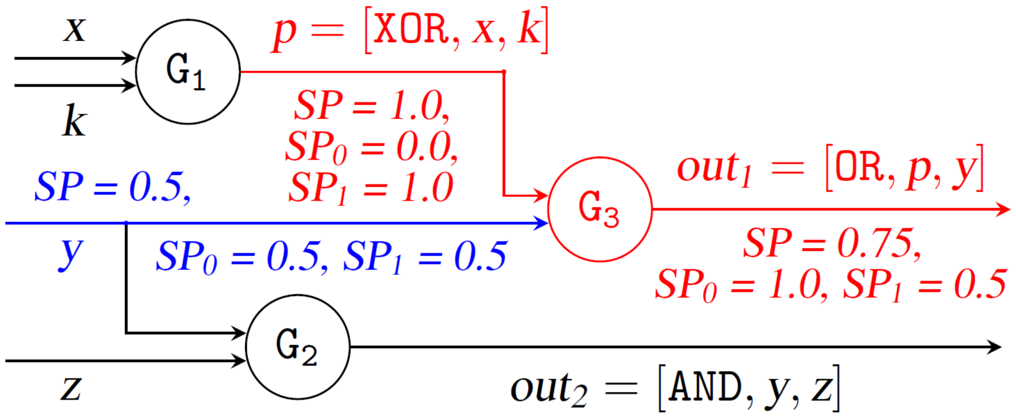
For a w-bit signal

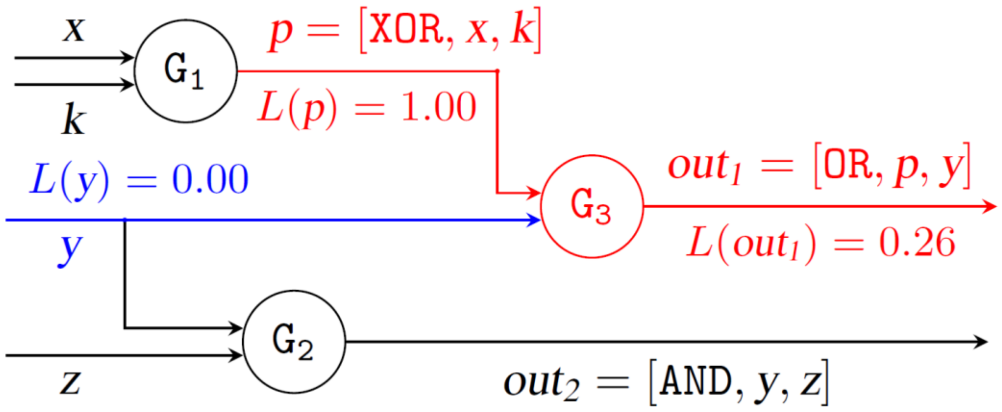
$$L_A(sig) = \sqrt{\sum_{i=1}^{w} L_A(sig[i])^2}$$

$$L_A(sig[i]) = L_A(sig)/\sqrt{w}$$

Leakage Evaluation (ctd)

Calculate leakage from signal probability, conditional signal probability values





FORTIFY: Runtime Complexity

Module in FORTIFY	Runtime Complexity
Directed Graph Representation	$O(G + S)$
Sub-circuit Extraction	$O(G + S)$
Signal Probability Estimation	$O(G' + S')$
Leakage Evaluation	$O(S')$

G = no. of gates in the input design
S = no. of signals in the input design
G' = no. of gates in the sub-circuit
S' = no. of signals in the sub-circuit

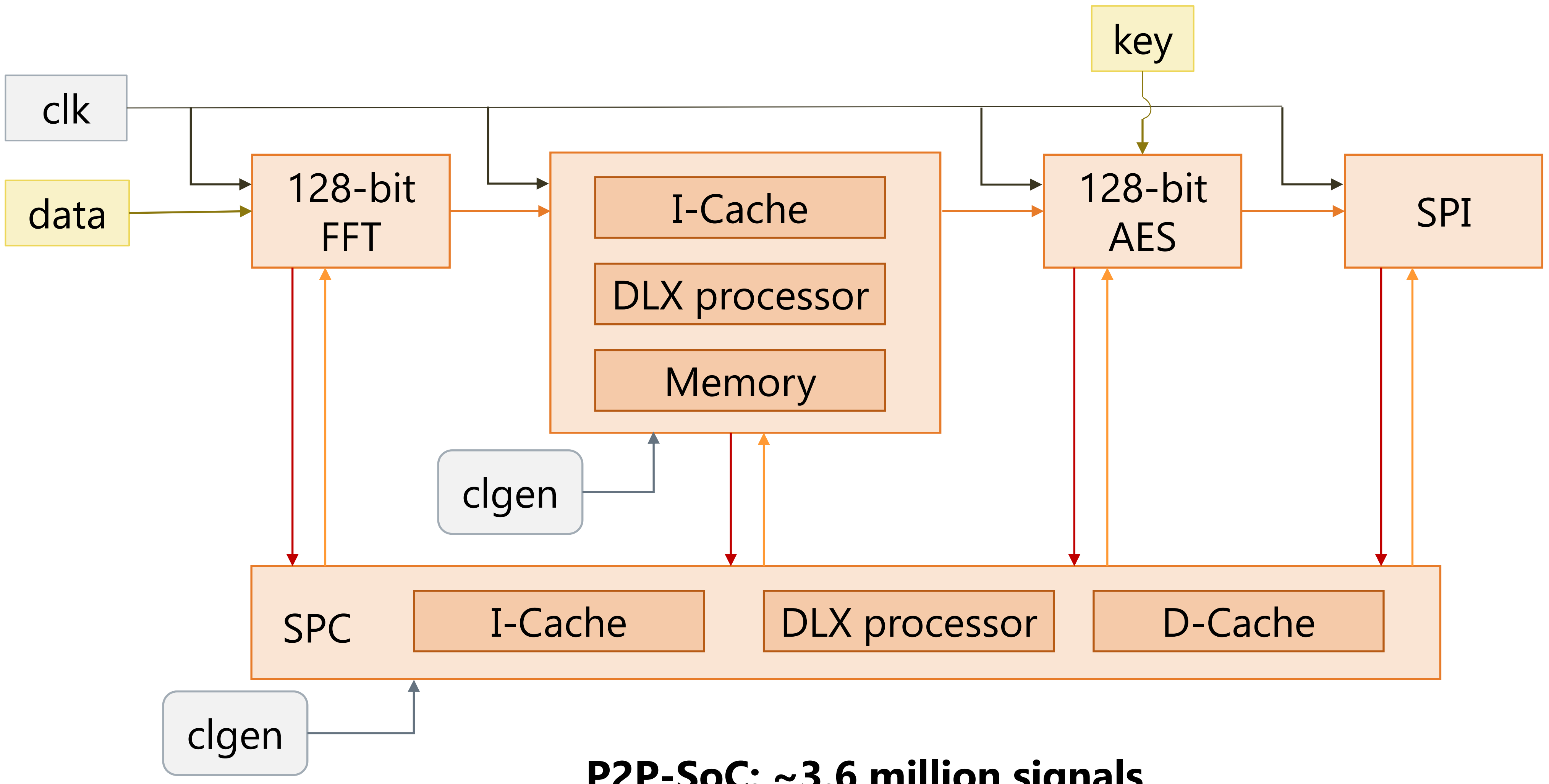
The runtime of FORTIFY is linear in the size of the input design

Results: FORTIFY v/s PLAN*

Design	# Signals	Time Taken		Pearson's Correlation	Spearman's Correlation
		PLAN	FORTIFY		
c17	11	1.6 min	0.88 s	0.975	0.923
FA-2	30	3.7 min	0.82 s	0.992	0.907
FA-4	46	5.5 min	0.75 s	0.995	0.910
FA-8	78	9.3 min	0.88 s	0.995	0.906
c432	276	33 min	0.91 s	0.997	0.652
PRE-Enc-1	6651	12.9 hr	3.62 s	0.989	0.943
PRE-Dec-1	6476	12.7 hr	3.88 s	0.990	0.898
PRE-Enc-2	7986	16.3 hr	4.47 s	0.977	0.806
PRE-Dec-2	7635	15.0 hr	4.83 s	0.984	0.809

* KF, Muhammad Arsath, et al. "PARAM: A Microprocessor Hardened for Power Side-Channel Attack Resistance." 2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST). IEEE, 2020.

Results: Evaluation of P2P-SoC$



P2P-SoC: ~3.6 million signals

$ https://github.com/apdn/P2PSoC

Results: Evaluation of P2P-SoC[$] (ctd)

Module	# Signals	Time Taken by FORTIFY	Estimated Time Taken by PLAN
P2P-SoC	~ 3.6 million	6 hrs	~ 7.5 months
FFT	~ 1.3 million	2.5 hrs	~ 3 months
DLX	~ 0.7 million	5 min	~ 1.5 months
AES	~ 0.3 million	1 min	~ 21 days
SPC	~ 1.3 million	2.5 hrs	~ 3 months
SPI	~ 20,000	8 sec	~ 33 hrs

[$] https://github.com/apdn/P2PSoC

Limitations of FORTIFY

- Does not consider physical sources of leakage
- Assumes that input design is free from reconvergent fanouts
- Assumes that input design does not have cyclic dependencies

FORTIFY: A Summary

- Early and fine-grained side-channel leakage estimation
- Scales up to evaluate very large designs
- Analytical approach using signal probabilities
- Can be incorporated in commercial EDA tools to design for security

Thank you!