UNIVERSITY OF Southampton
School of Electronics and Computer Science

Hardware-Level Security for the IoT

Mark Zwolinski

March 2017

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Outline

- Background, IoT, Hardware/Software, Threats/Risks
- Hardware-level security
- PUFs
- Anomaly detection
- Summary

UNIVERSITY OF Southampton
School of Electronics and Computer Science

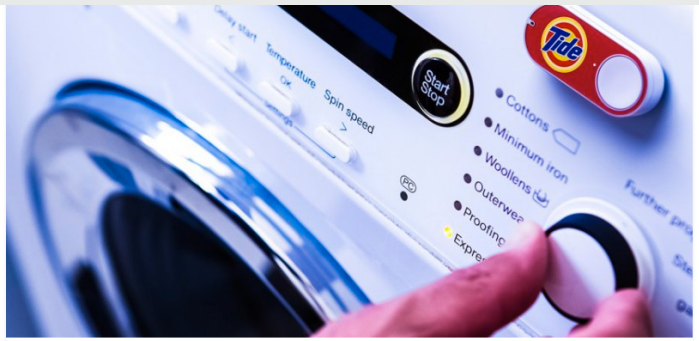
IoT / Embedded Systems

- Not desktop / server systems:
  - 20-30 year lifetimes
  - May be safety-critical
    - automotive
    - medical
  - Access to private networks
  - Limited resources


ANDY GREENBERG SECURITY 07.21.15 6:00 AM

HACKERS REMOTELY KILL A JEEP ON THE HIGHWAY—WITH ME IN IT

wired.com



Amazon Dash Button: IoT Risk in Your Home or Not?

July 10, 2016 Christopher Isak

techacute.com

IoT Security Foundation

iotsecurityfoundation.org

UNIVERSITY OF Southampton
School of Electronics and Computer Science

How is hardware different to software?

- Hardware exists in the real world
  - Physical access allows side-channel attacks
- Implementation is not the same as design
  - Timing
  - Energy
- Every device is unique
  - Variability
- But "Hardware is the root of trust"

UNIVERSITY OF Southampton
School of Electronics and Computer Science

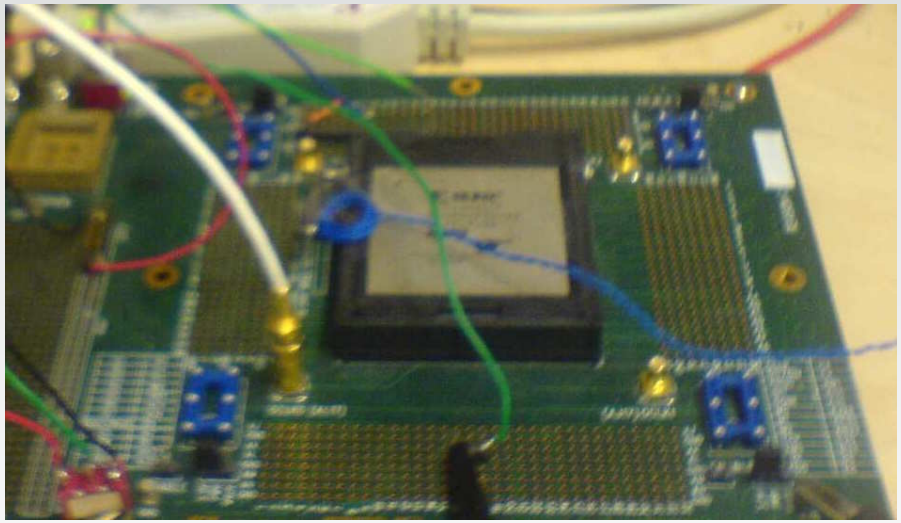
Threats / Risks

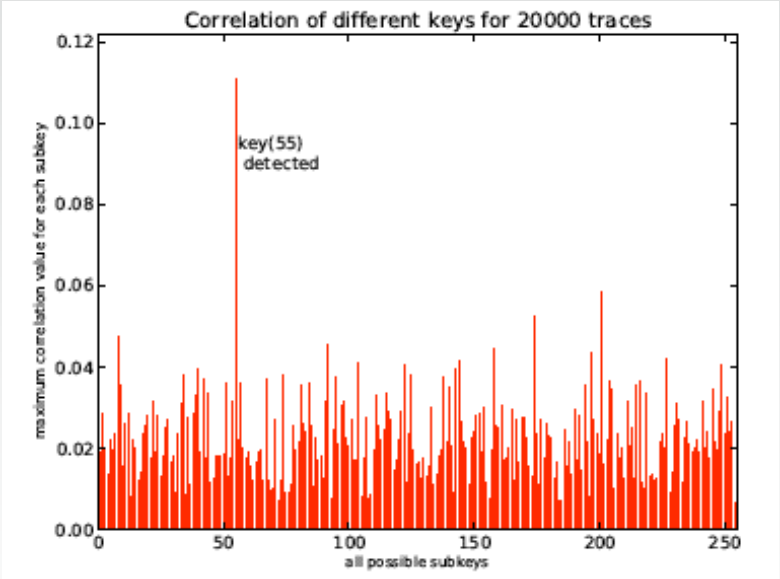
- Physical access
  - Power supply monitoring
  - Changing environment
- Trojans
  - The supply chain is long and not well understood
  - What *exactly* is on your chip?
- Remote hacking
  - Buffer overflow -> root shell
- Loss of data
- Privacy
- Remote control
- Denial of service

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Side-Channel Attacks on Crypto

- Example: Differential Power Analysis
  - AES on FPGA
  - Simple probe





UNIVERSITY OF Southampton
School of Electronics and Computer Science

Security at Hardware-level

- Physical Uncloneable Functions (PUFs)
  - Exploit variability between ICs to give a "fingerprint"
  - Key generation
  - Authentication
- On-chip monitoring
  - Anomaly Detection

UNIVERSITY OF Southampton
School of Electronics and Computer Science

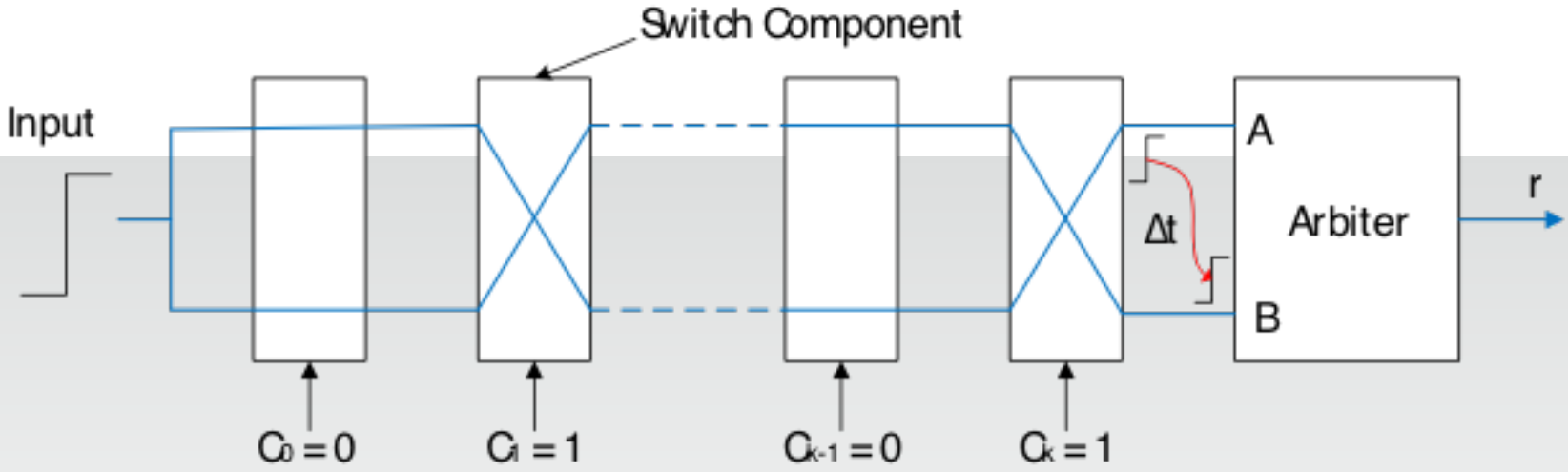
PUFs

- Ring Oscillators
  - Exploit variability in frequency
- SRAM
  - Use start-up values – random, but repeatable
- Need to be on-chip (CMOS)
- Need ECC
- Long term reliability
- Can be hacked by Machine Learning attacks

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Arbiter PUF

- Exploit differences in signal paths to get unique bit patterns

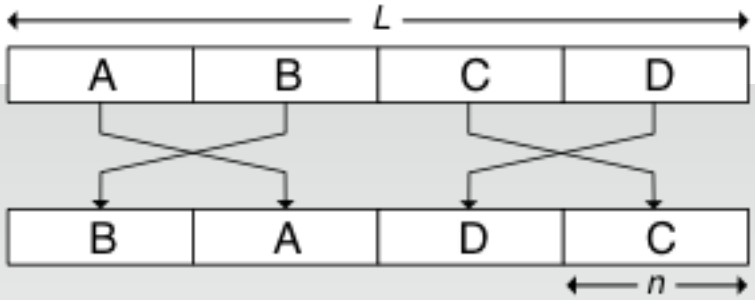


- C is a key – apply different values to get a set of responses
- Low cost (power, area), but vulnerable to Machine Learning

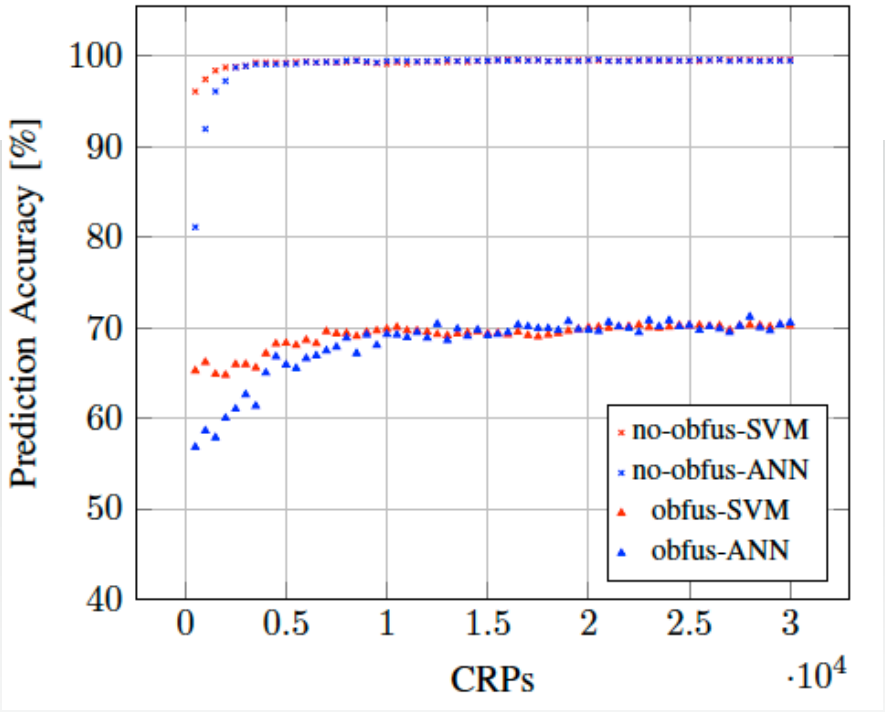
UNIVERSITY OF Southampton
School of Electronics and Computer Science

Arbiter PUF Obfuscation

- Simple permutations can significantly reduce predictability.



- 50% predictability is ideal



UNIVERSITY OF Southampton
School of Electronics and Computer Science

On-Chip Anomaly Detection

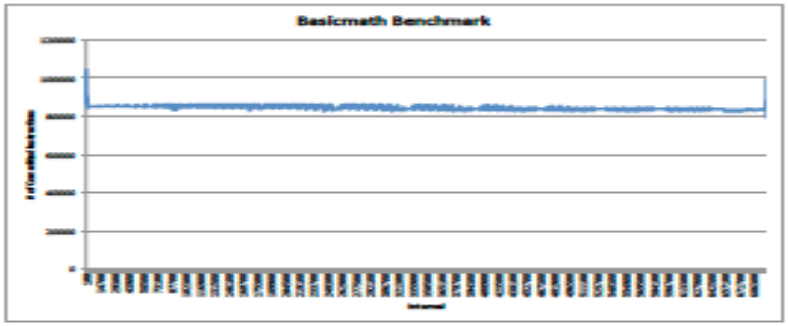
- Hypothesis: Embedded systems do predictable things
- Therefore anomalous behaviour occurs because something bad has happened
  - Reliability problem
    - One-off (radiation) or gradual (ageing)
  - Security problem
    - Sudden, sustained
- May be able to react much more quickly in hardware than in software

UNIVERSITY OF
Southampton
School of Electronics
and Computer Science

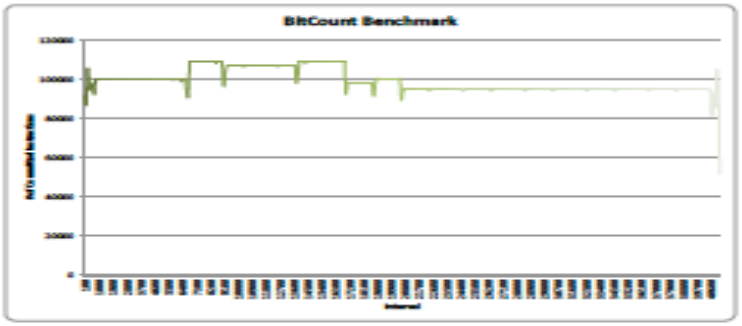
Normal Behaviour

- Different programs have patterns

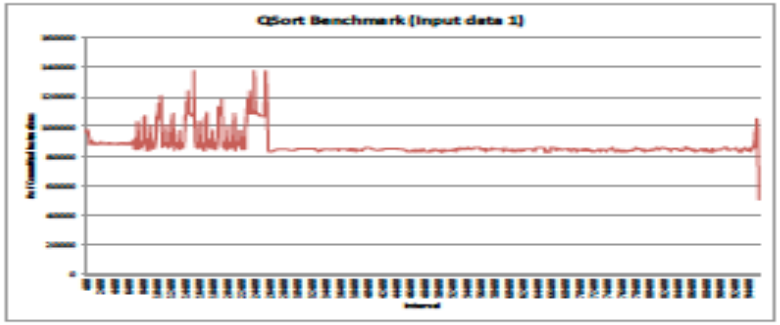
Committed Instructions:



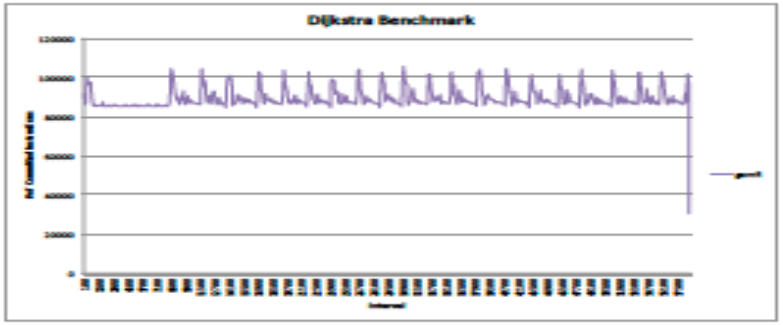
(a)



(b)



(c)



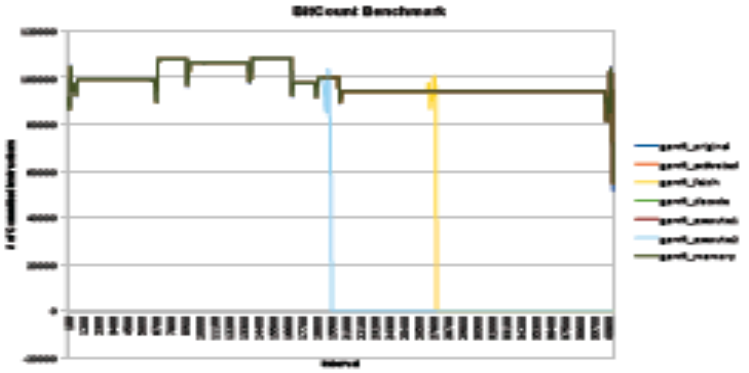
(d)

UNIVERSITY OF
Southampton
School of Electronics
and Computer Science

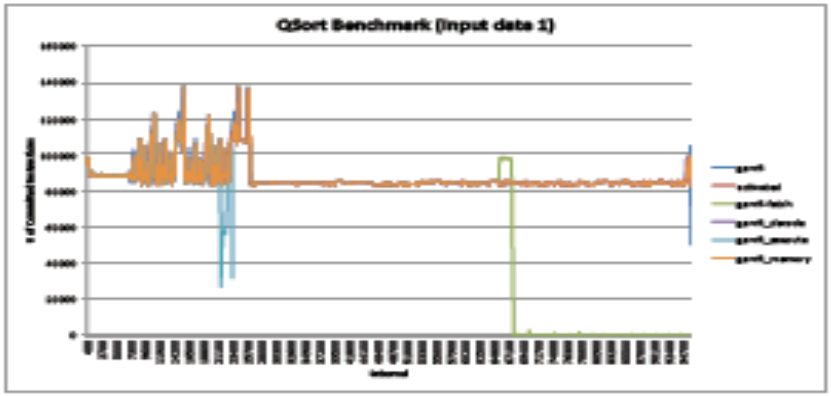
Anomalous Behaviour

- Injected faults (not attacks)

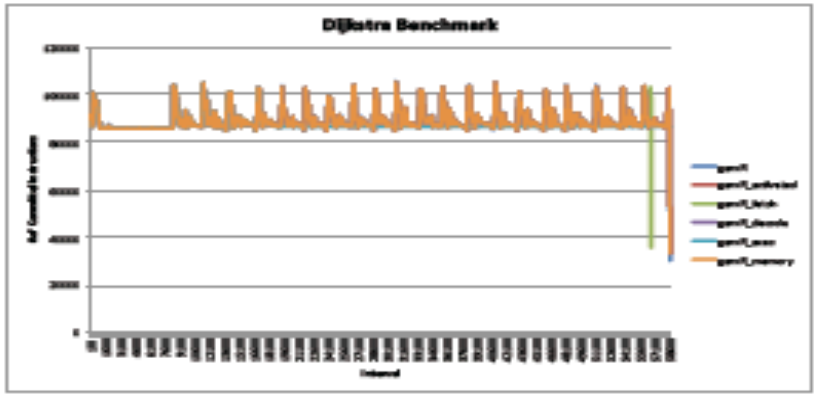
(a)



(b)



(c)



(d)

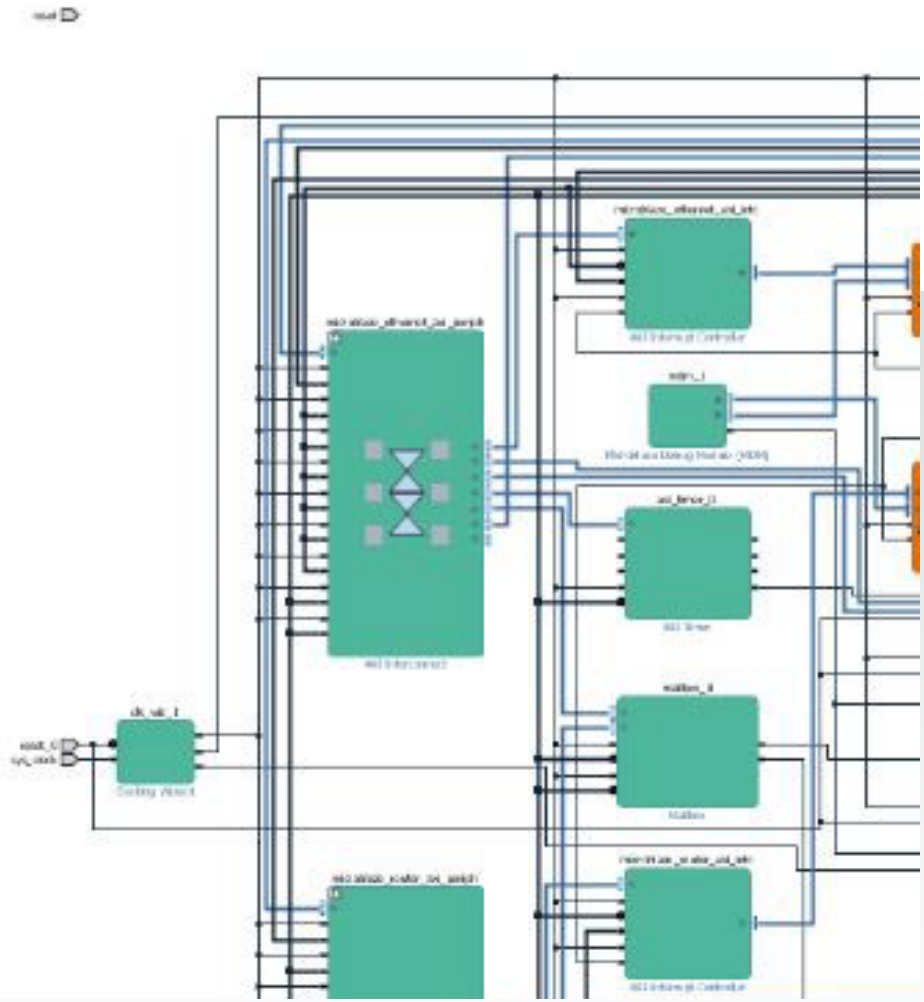
UNIVERSITY OF Southampton
School of Electronics and Computer Science

Anomaly Detection

- Security anomaly may cause different types of unusual behaviour
  - Program Counter has unusual pattern
  - Cache Miss rate suddenly increases
  - Temperature suddenly rises

On-Chip Detection

UNIVERSITY OF Southampton
School of Electronics and Computer Science



- Xilinx Microblaze
- Implemented a new Vivado Block
- Features AXI peripherals

Data Model

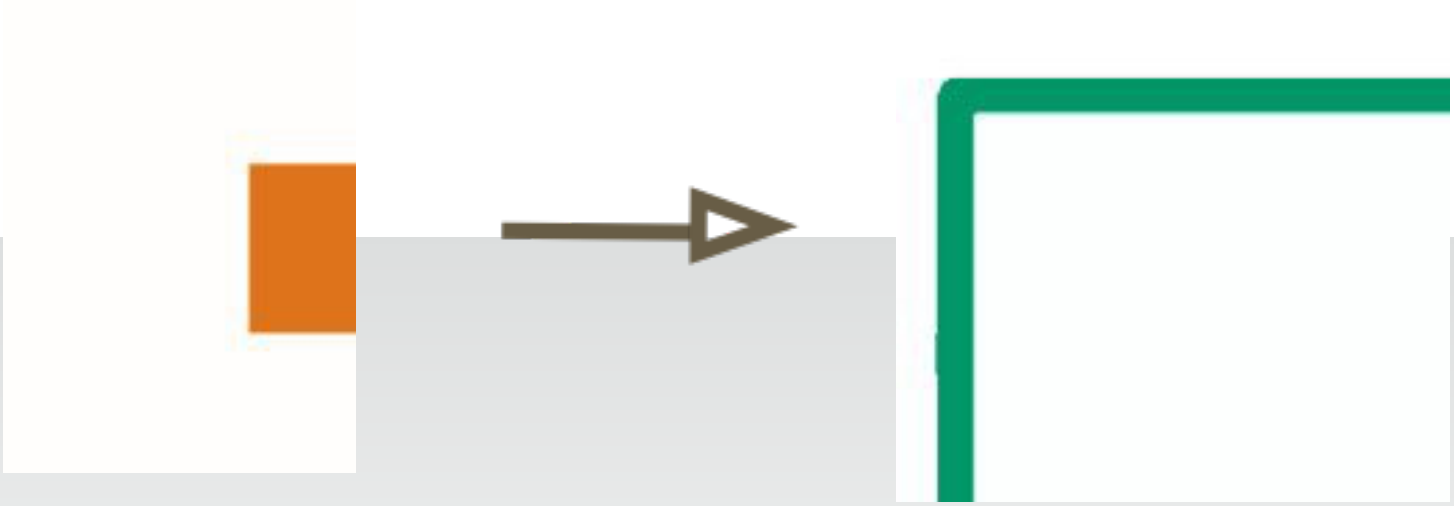
UNIVERSITY OF Southampton
School of Electronics and Computer Science

Implemented as a deterministic alternative to a sparse matrix

- **Advantages**
  - Deterministic
  - Using ‘chunks’ of the program counter which
    the size
  - Implements tally to keep track of how many
    path is accessed which allows ‘unlikely path’ d
- **Disadvantages**
  - Larger space requirement
    - Map can be optimised off-chip with kn
      the program execution
  - Still using the program counter
    - Only map branch instructions

Learning

UNIVERSITY OF Southampton
School of Electronics and Computer Science



On-chip only

- Cannot construct model in real time in learning mode
- Not enough memory on chip to store for later processing

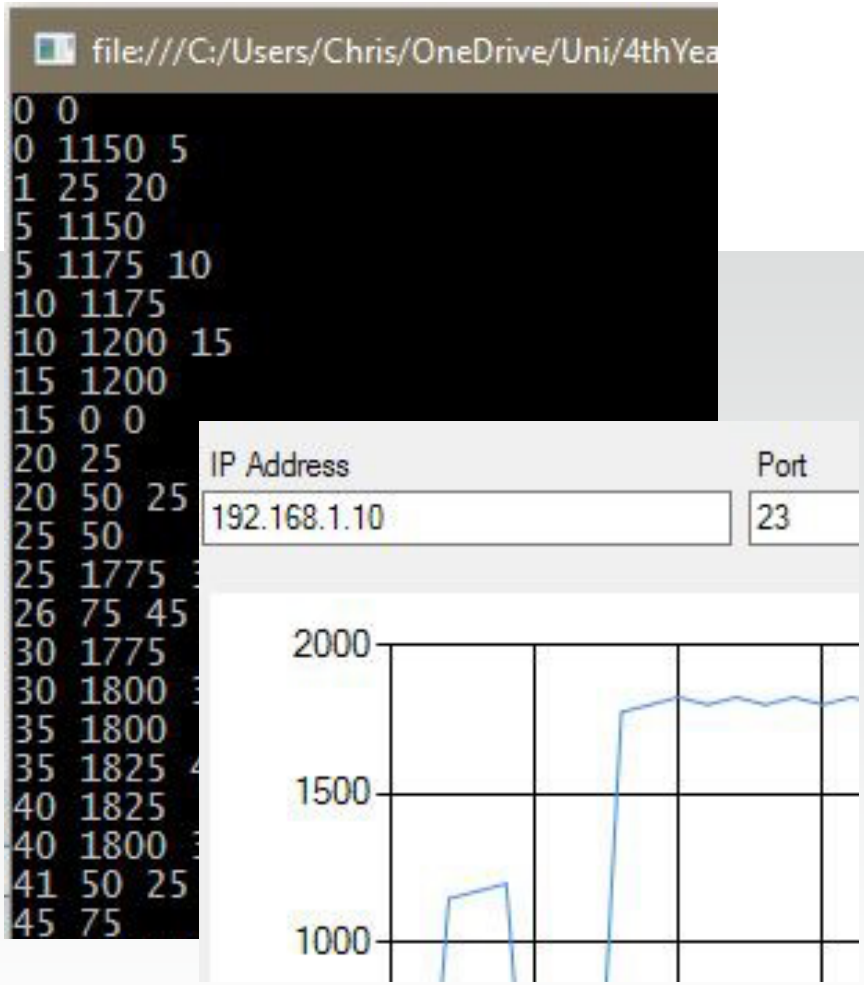
With PC

- Much more memory available
- More processing power to construct model
- Device cannot independently produce a new model to deal with changes in program e.g. updates

Learning

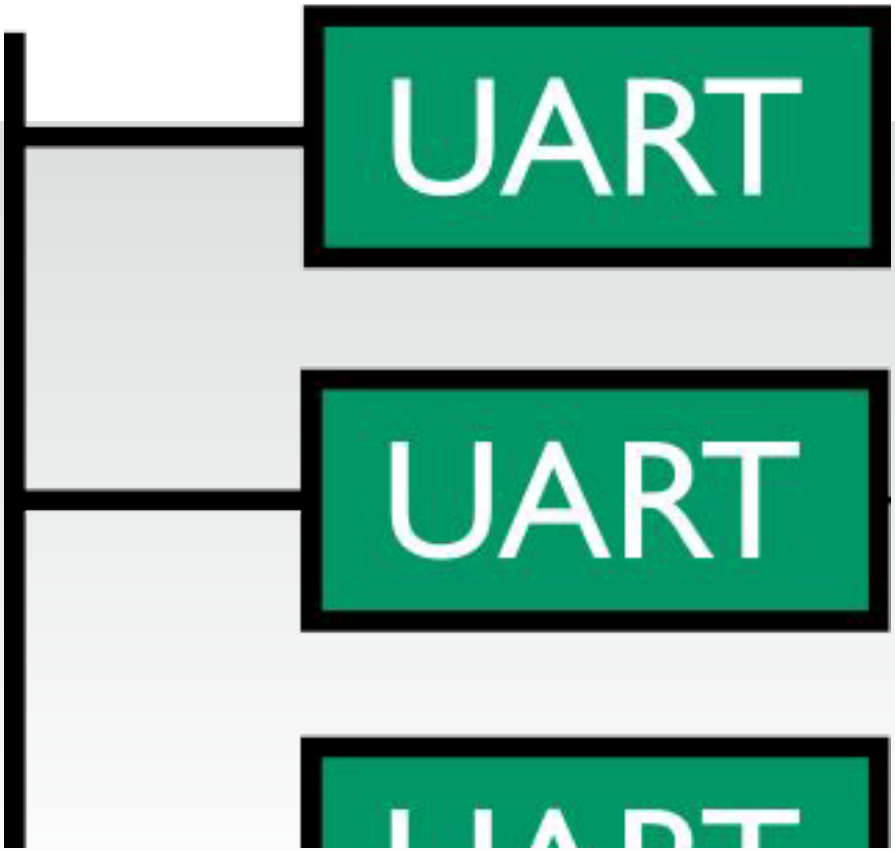
UNIVERSITY OF Southampton
School of Electronics and Computer Science

- Implemented a second microblaze processor on the FPGA which outputs the trace data to a PC via Ethernet.
- An AXI peripheral added to buffer program counter values.
- PC program developed to log the received data
- Data then transmitted back to the device and then processed to save the directed graph in memory.
- PC can also implement more complex model algorithms and enable more rapid prototyping.



IoT Exemplar

UNIVERSITY OF Southampton
School of Electronics and Computer Science



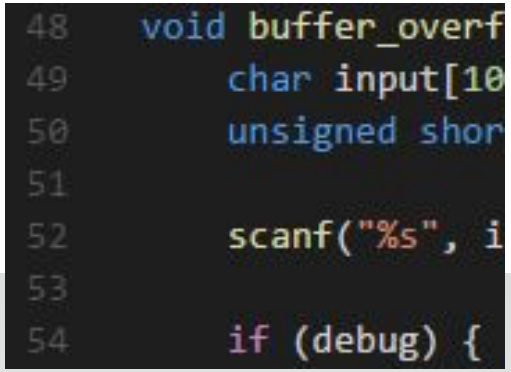
Evaluation

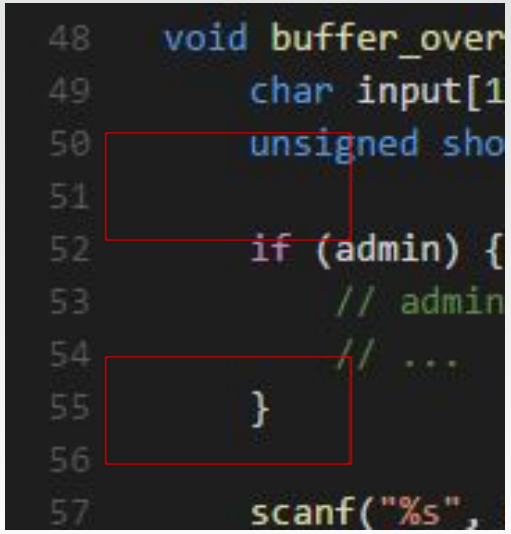
UNIVERSITY OF Southampton
School of Electronics and Computer Science

1. **Timing**
   - Does the algorithm run in real time with the processor?
2. **Hardware Size**
   - How much space on the FPGA does the anomaly detection hardware consume?
3. **Power Consumption**
   - How much additional power is consumed by the extra hardware?
4. **User Complexity**
   - What additional equipment is required to configure and run the detector?
5. **Defensive Capabilities**

Attacks that...

- Modify the execution of the program
  - Entirely new execution
  - Known execution in an unknown pattern
- Crash the program
- Change the output of the program





UNIVERSITY OF Southampton
School of Electronics and Computer Science

Summary

- Hardware has different characteristics to software
- PUFS – Exploit variability in manufacturing
- Anomaly detection – different types of threats; faster, different response

UNIVERSITY OF Southampton
School of Electronics and Computer Science

Acknowledgements

- PUFs – Mohd Syafiq Mispan, Dr Basel Halak
- Anomaly Detection – Elena Woo
- On-chip monitoring project – Jack Cosslett, Emma Curati-Alasonatti, Chris Holbrow & James Middleton-Jones
- Microsoft Research for Azure Computing Resources