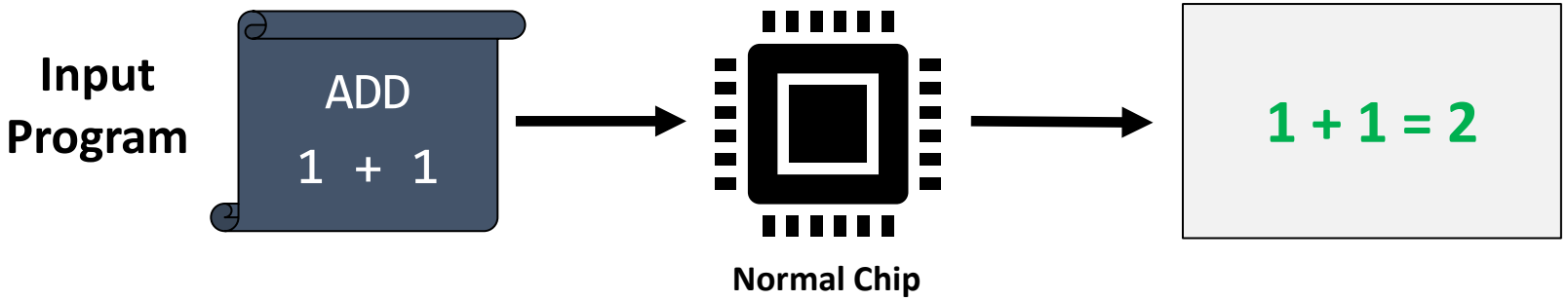
ML Accelerator Reliability: Assessing Faults During Training

Peter Deutsch (MIT), Vincent Ulitzsch (MIT/TU Berlin),
Sudhanva Gurumurthi (AMD), Joel Emer (MIT), Mengjia Yan (MIT)

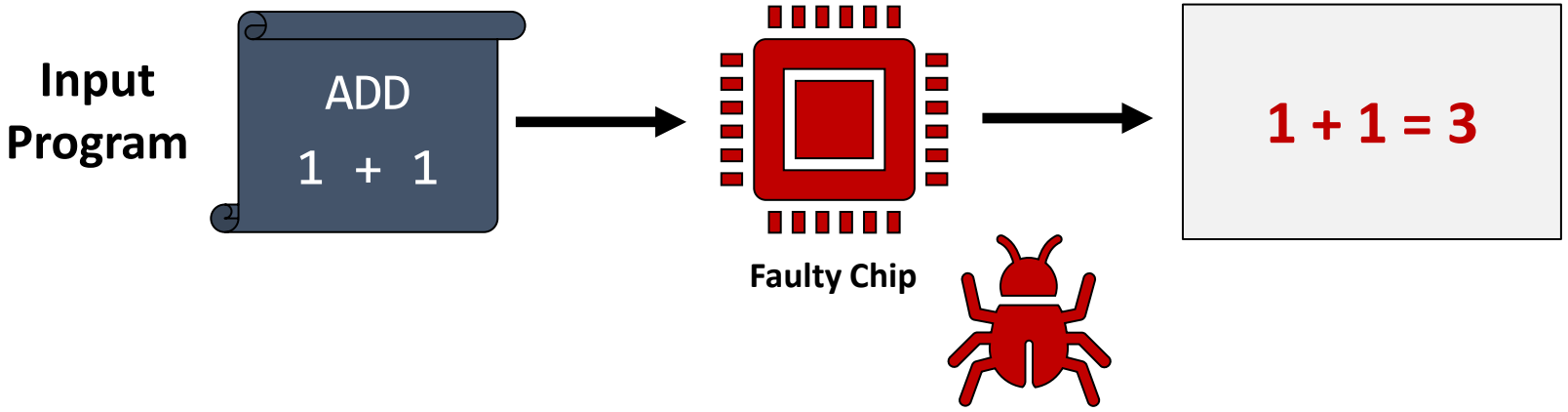
MIT AI Hardware Program

March 31st, 2025

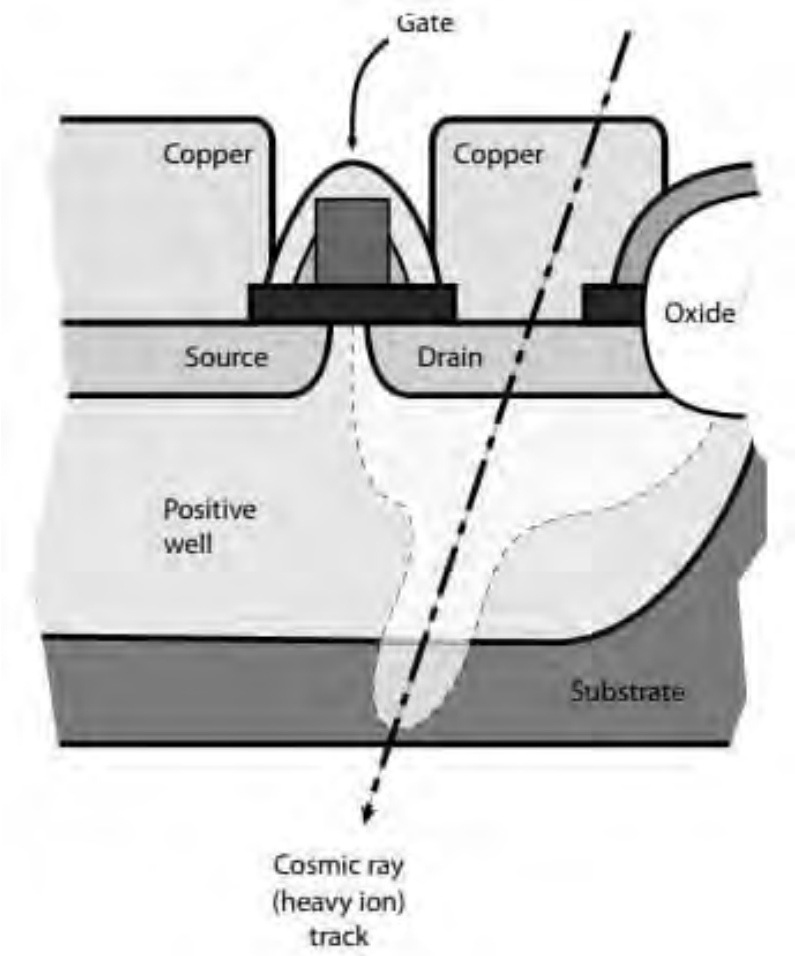
Chips can miscompute, resulting in Silent Data Corruptions (SDCs)



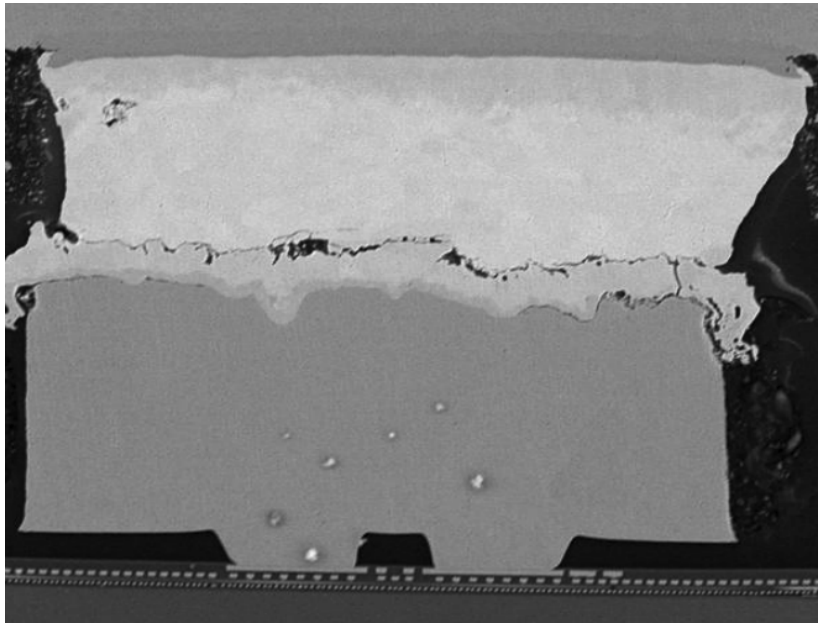
Silent data corruptions occur when faulty chips silently miscompute, resulting in incorrect results



What can cause faults?

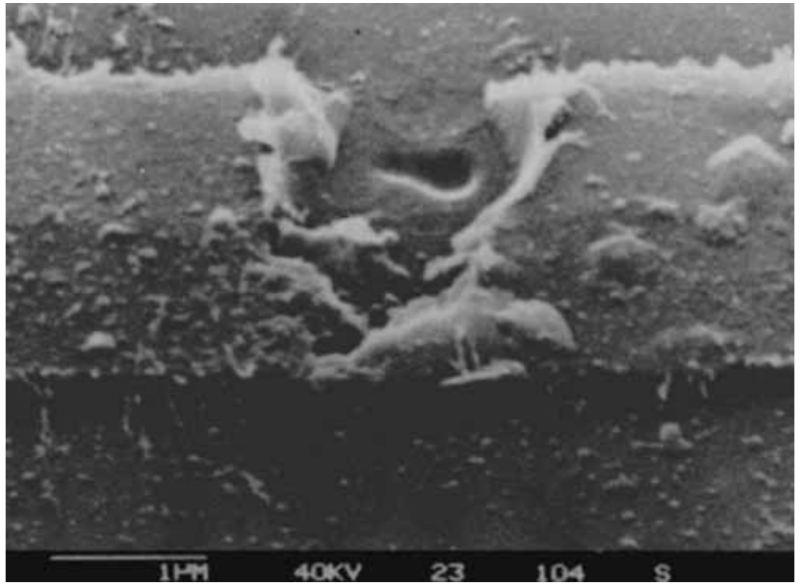


Particle Strikes
(Transient)



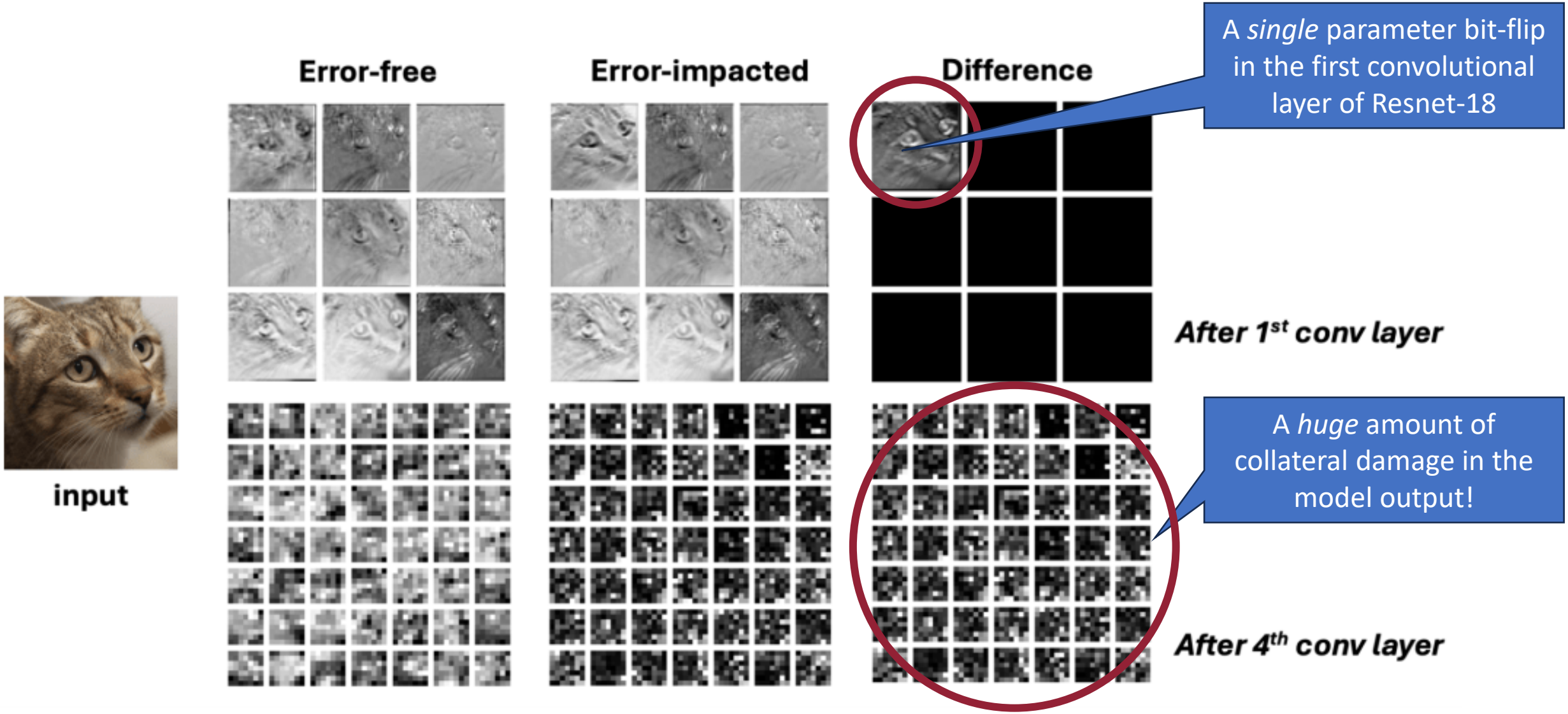
Increased resistance
(*Intermittent)*

Electromigration
(Permanent)



https://www.scienceabc.com/innovation/what-are-bit-flips-and-how-are-spacecraft-protected-from-them.html; https://ieeexplore.ieee.org/document/4925824; https://onlinelibrary.wiley.com/doi/abs/10.1002/maco.19930440405

Errors During Machine Learning Inference



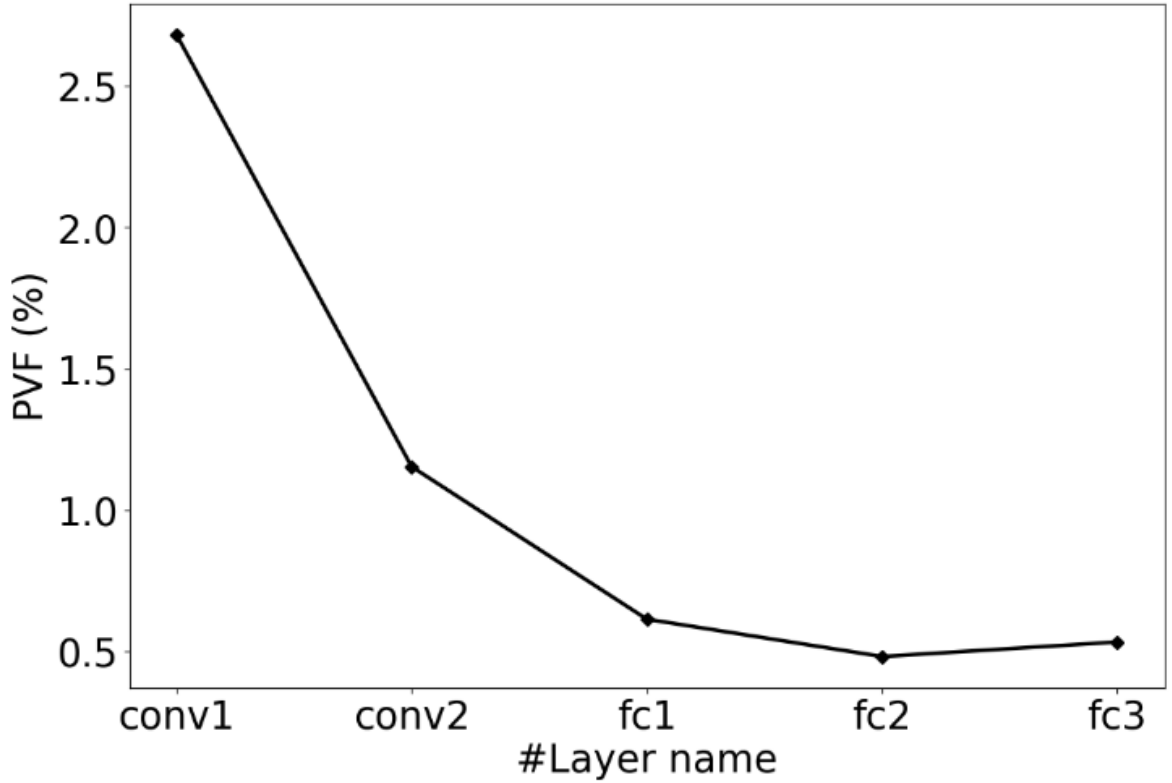
Jiao, Xun, et al. "PVF (Parameter Vulnerability Factor): A Quantitative Metric Measuring AI Vulnerability and Resilience Against Parameter Corruptions. ArXiv.

Do bit-flips actually matter? It's possible!

TABLE I
CNN LAYERS W/ WEIGHT COUNTS

Layer Name	Weight Size	Weight Count
conv1	[6, 1, 5, 5]	150
conv2	[16, 6, 5, 5]	2400
fc1	[120, 256]	30720
fc2	[84, 120]	10080
fc3	[10, 84]	840

Upwards of 2.5% of bits in a LeNet convolutional layer will result in a final model misclassification if they're flipped!



Jiao, Xun, et al. "PVF (Parameter Vulnerability Factor): A Quantitative Metric Measuring AI Vulnerability and Resilience Against Parameter Corruptions. ArXiv.

Silent Data Corruptions During ML Training

"Despite best efforts, given the scale of ML systems and the size of ML training jobs, hardware errors can occur, and sometimes incorrect computations from one buggy chip can spread and infect the entire training system"

-- Jeff Dean, Chief Scientist for Google Research and Deepmind
(MLSys Keynote 2024)

What makes ML training vulnerability different than inference?

Inference

- A single fault can affect a single classification.
- **Cost of a failure:** Potentially incorrect classification

- Well studied by literature (including both reliability and security angles)

Training

- A single fault can affect the training state of the model and affect many downstream classifications.
- **Cost of a failure:** Degraded model accuracy

- **Unknown: Is it robust enough?**

Effects of Faults During Training

What's the impact of a transient/permanent bit-flip during training, either in the forward or backwards pass?

We identify three possible outcomes

#1

Nothing happens!

Training/test accuracy is not meaningfully affected.

#2

Crash!

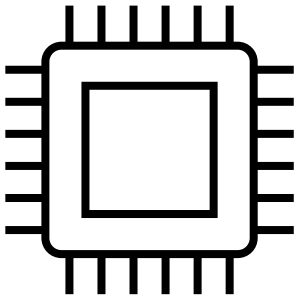
Training experiences an INF or NaN result.

#3

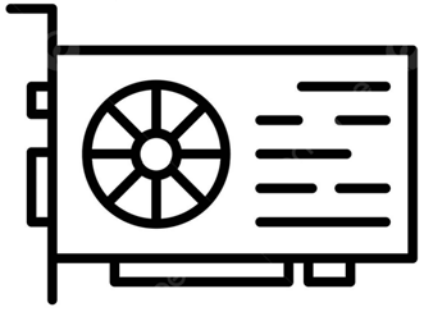
SDC!

The training is affected in a strange way…

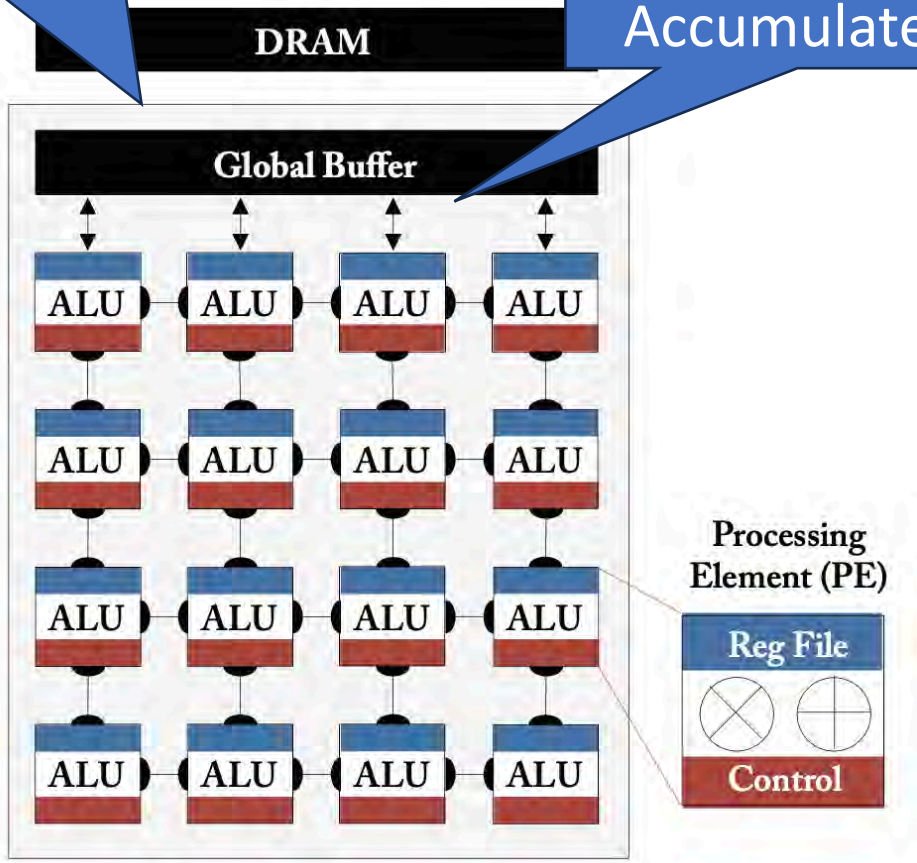
Accelerating DNNs via Hardware



CPU



GPU



DNN Accelerator

Speed-up

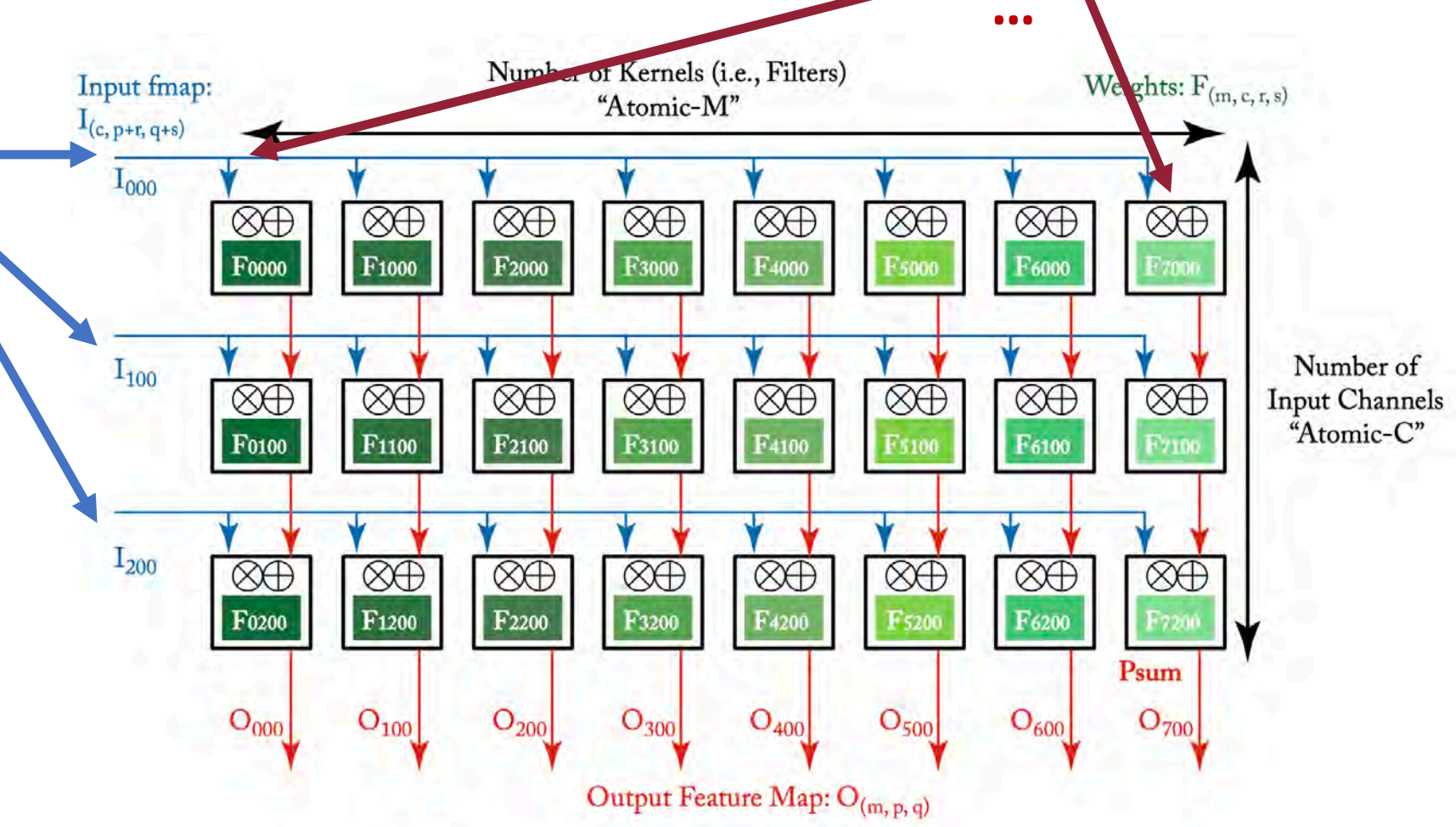
To understand the impact of HW faults, we need to take the following steps:

1. We choose a particular fault model.
2. We derive a software error model for that fault model, corresponding to the underlying hardware architecture.
3. We inject errors into a DNN model according to the software error model (e.g., via Tensorflow/Pytorch simulations).
4. We evaluate the impact of these errors on the training and validation accuracies.

NVDLA Convolution Core Architecture

Each column corresponds to a single output channel

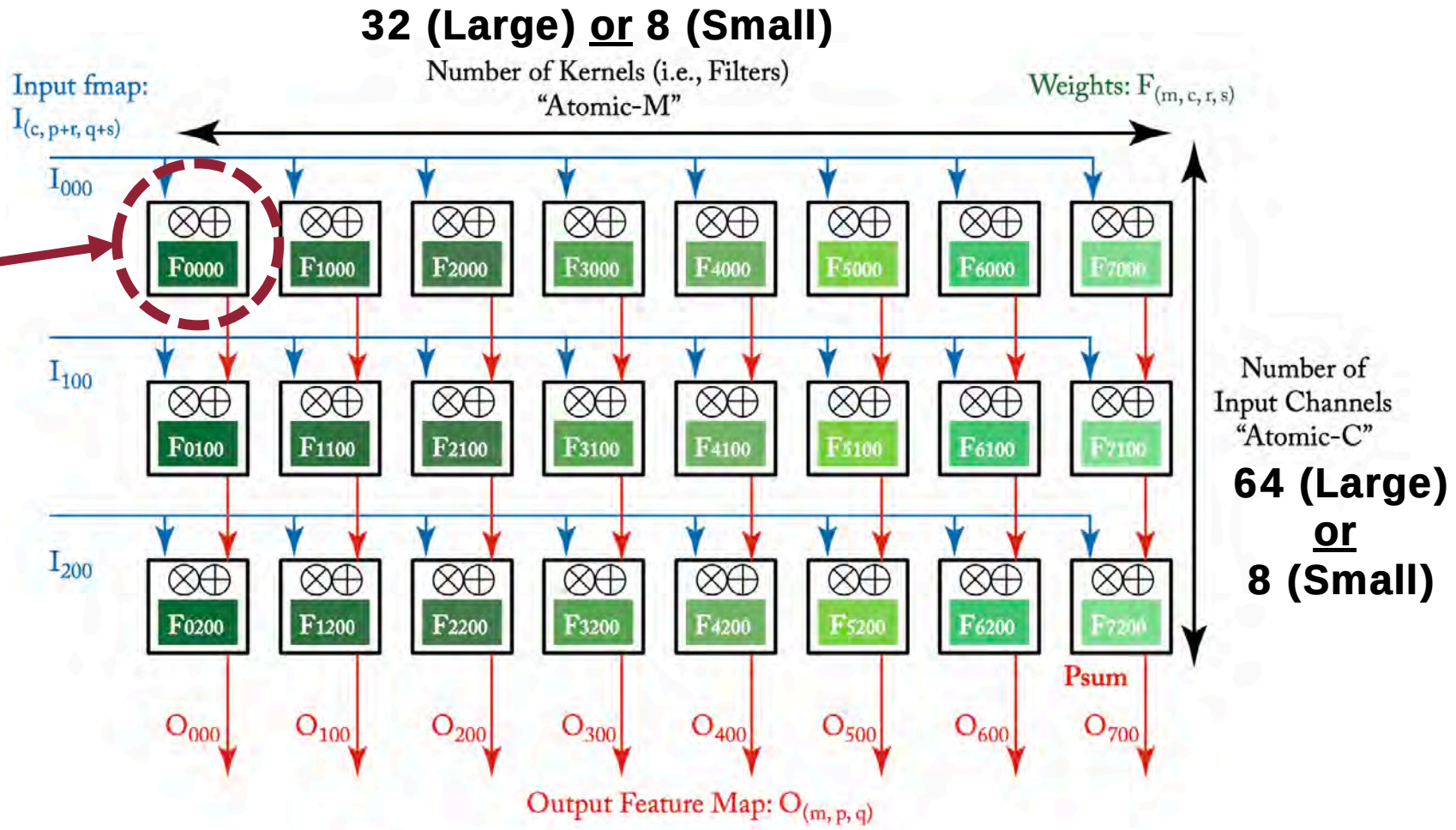
Each row corresponds to a single input channel



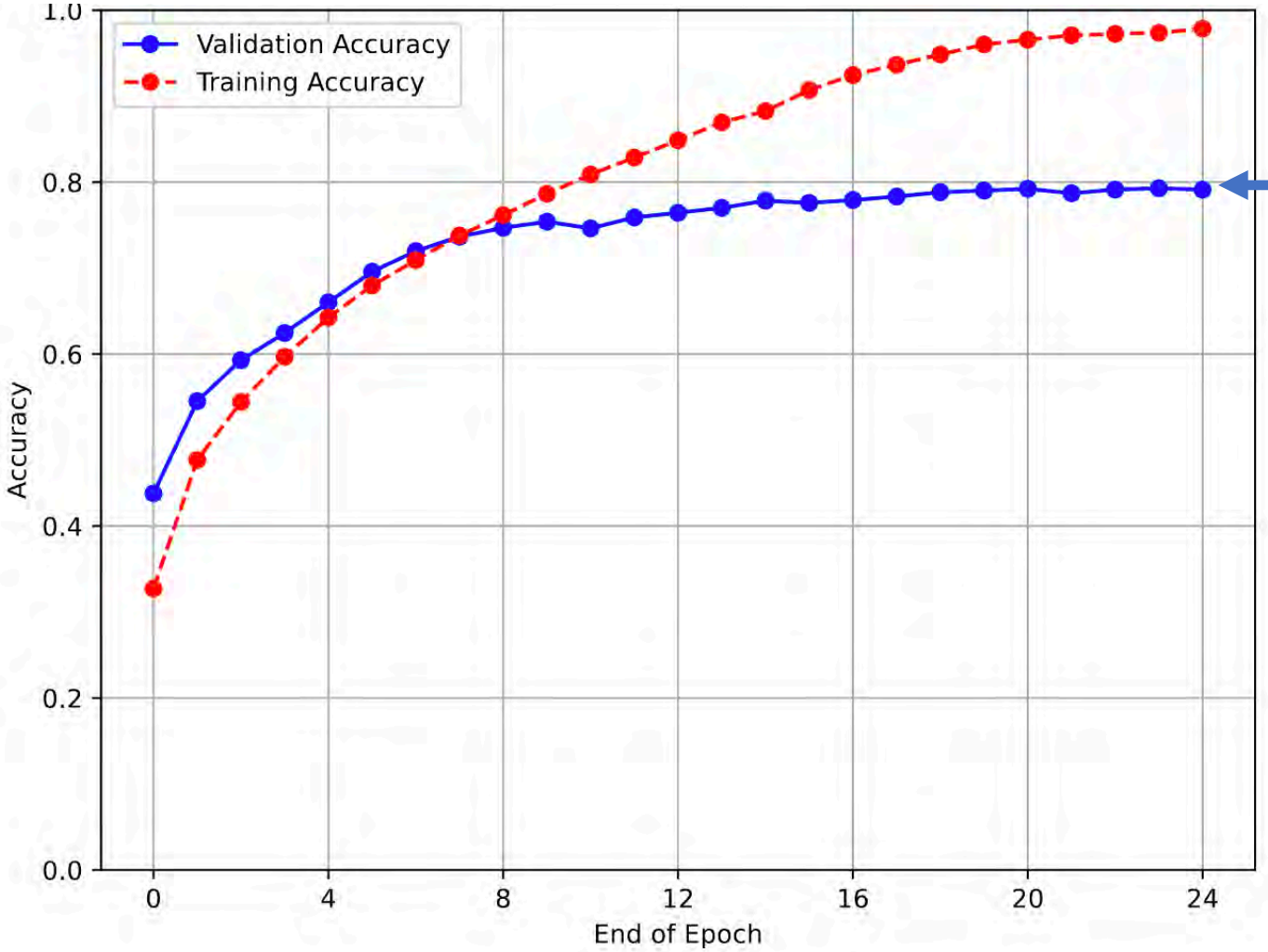
Weight Faults in NVDLA

Assume a stuck-at bit in the weight computation logic of a single PE

Software Error Model:
All computations involving weights in input channel 0, filter/output channel 0 will be incorrect!



Alexnet Fault-Free Accuracy



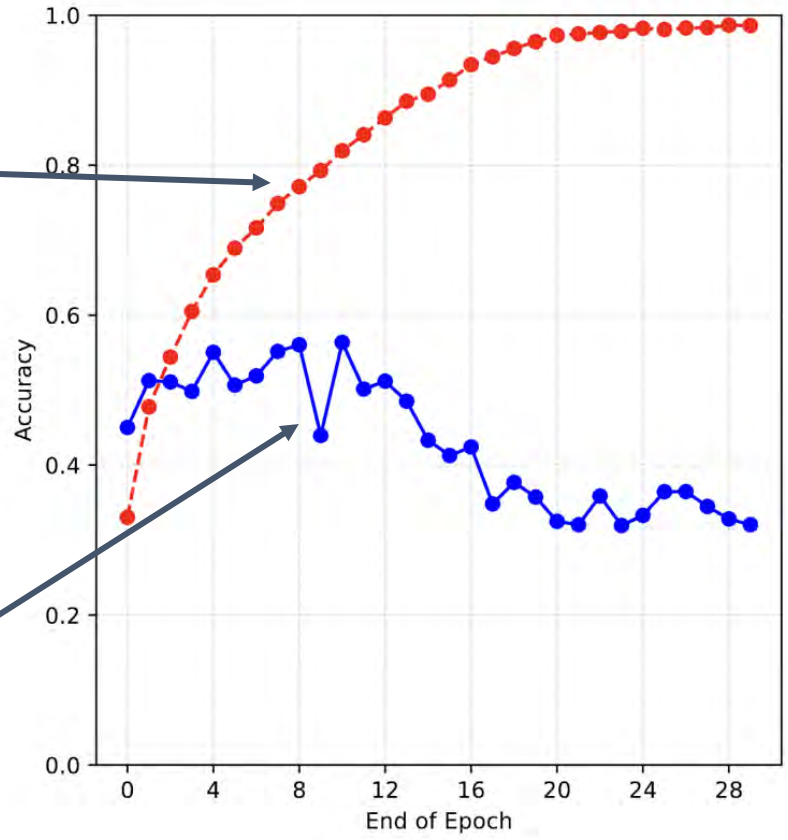
Validation reaches ~79% accuracy

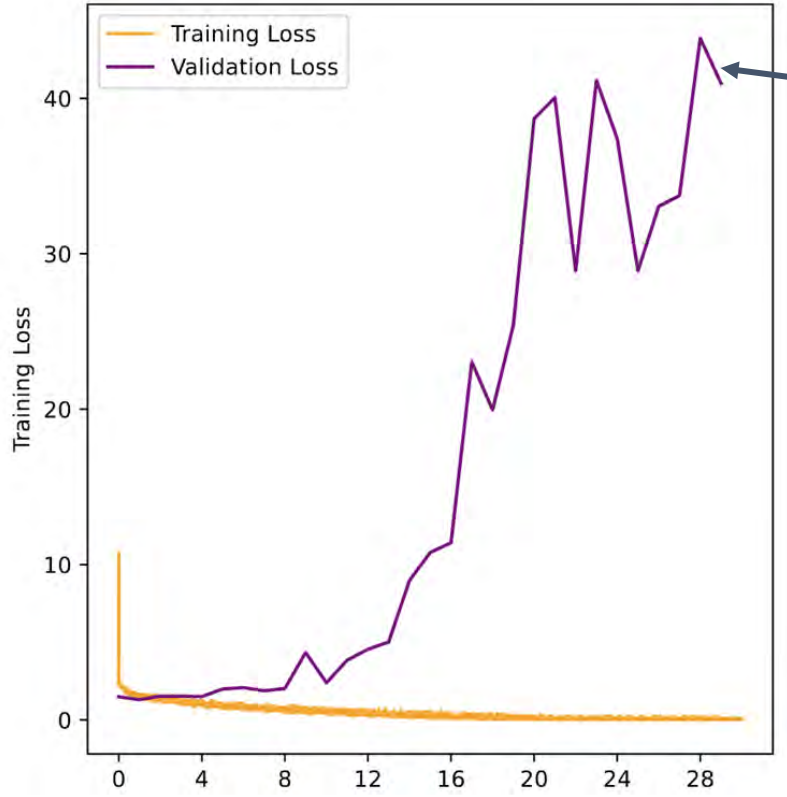
Training Under Stuck-at-0 Fault on Alexnet

Bit 28 Stuck-at 0, Seed 123, Small Config, Permanent Fault

No obvious change in training accuracy (computed with fault)

Validation accuracy (computed without fault) is severely degraded





Validation curve continuously gets *worse* as training continues

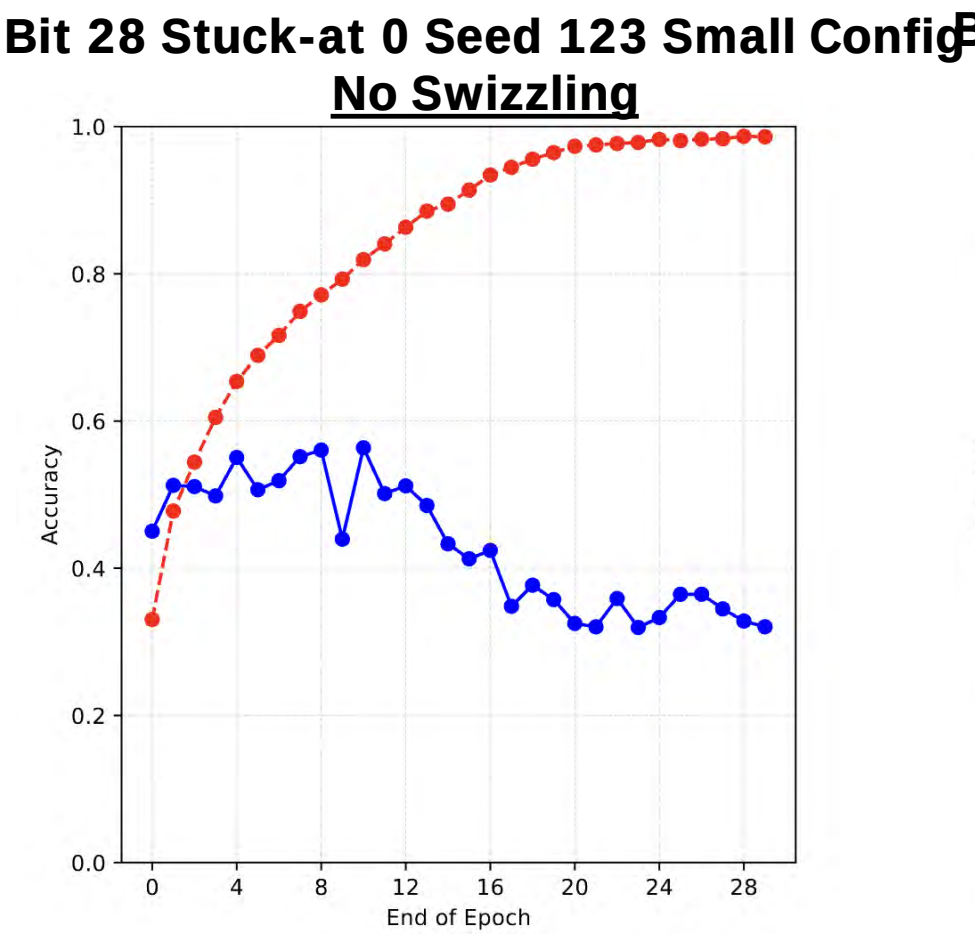
Observation: Despite there being no sharp drops in accuracy, a fault can sometimes make the resultant model nearly useless!

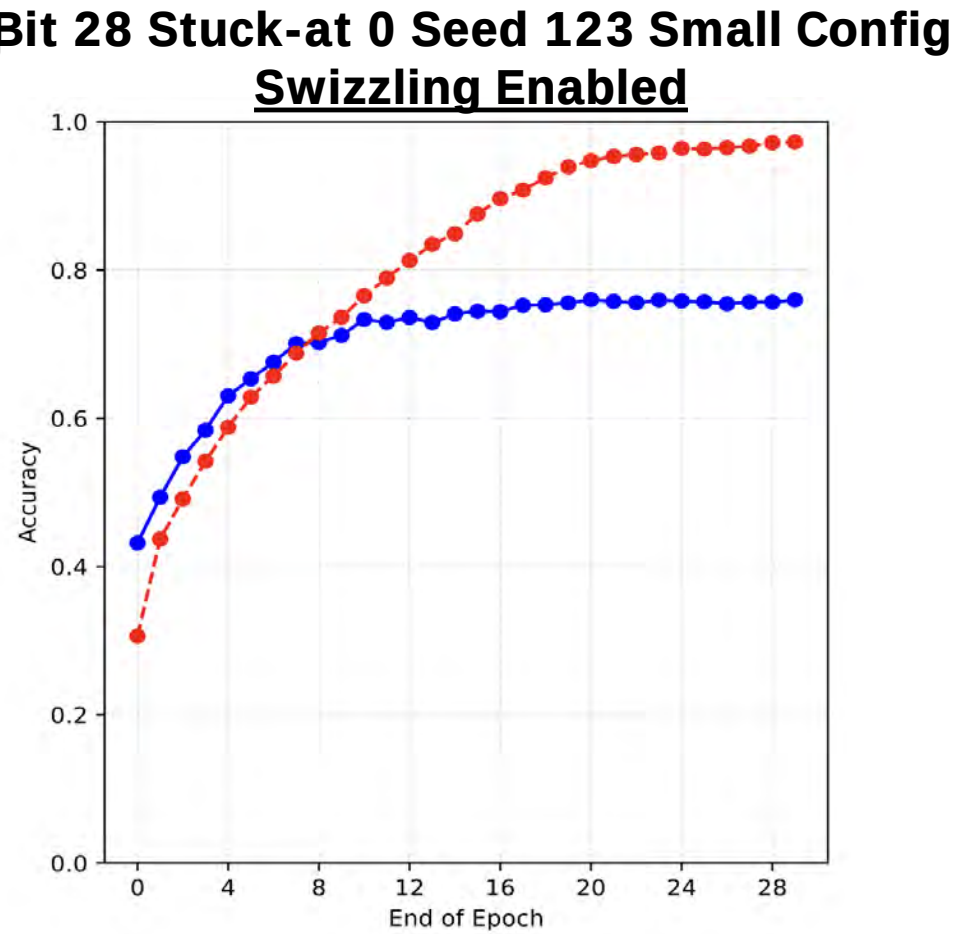
Theory: The model is adapting to the presence of the fault and performs poorly when the fault condition is removed.

Ongoing Research: Under which conditions does this behaviour occur? Can we describe this behaviour analytically?

Channel “Swizzling”

Idea: If we accept that a fault will happen, can we change the underlying hardware to have a software error pattern that is less impactful?





ML Accelerator Reliability: Assessing Faults During Training

Peter Deutsch (MIT), Vincent Ulitzsch (MIT/TU Berlin),
Sudhanva Gurumurthi (AMD), Joel Emer (MIT), Mengjia Yan (MIT)

MIT AI Hardware Program

March 31st, 2025

Technische Universität Berlin

AMD

MIT CSAIL

compute. collaborate. create