On-chip FPGA Debug Instrumentation for Machine Learning Applications

Daniel Holanda Noronha[1] , Ruizhe Zhao[2] , Jeff Goeders[3] , Wayne Luk[2] and Steven J.E. Wilton[1]
[1]University of British Columbia, [2]Imperial College London, [3]Brigham Young University
danielhn@ece.ubc.ca, ruizhe.zhao15@imperial.ac.uk, jgoeders@byu.edu, w.luk@imperial.ac.uk, stevew@ece.ubc.ca

FPGA 2019 – Monterey, California

THE UNIVERSITY OF BRITISH COLUMBIA

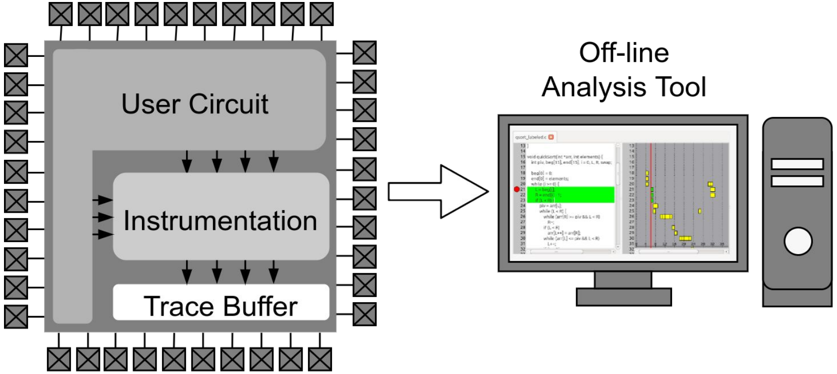
Imperial College London

BYU BRIGHAM YOUNG UNIVERSITY

Previous Work

On-chip debug

- Records the behaviour of the design as it runs at speed for later interrogation
- Is necessary:
    - Simulation is usually too slow
    - Environment can often not be adequately described
- Challenge:
    - Record enough information on-chip to understand the problem

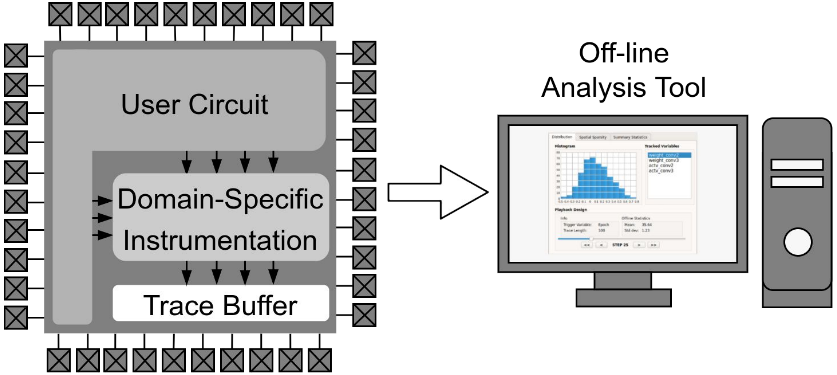


Our approach

A flow to accelerate the debug of machine learning applications on FPGAs

- Previous work is not ideal for debugging ML circuits
  - Even longer run-times; “Correctness” hard to determine; Commonly designed at a high level.

- This work uses domain-specific characteristics of ML circuits to:
  - Maximize the utilization of trace buffer space
  - Provide information that is meaningful to an engineer

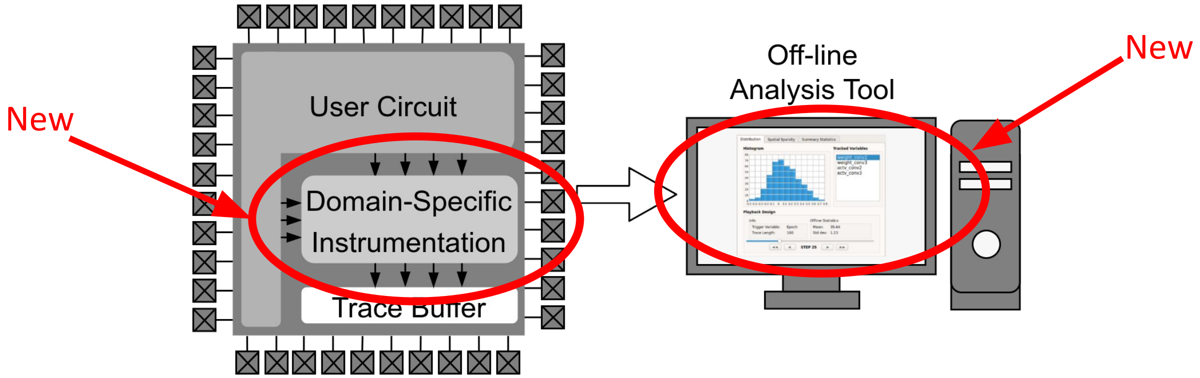


Our approach

A flow to accelerate the debug of machine learning applications on FPGAs

- Previous work is not ideal for debugging ML circuits
    - Even longer run-times; “Correctness” hard to determine; Commonly designed at a high level.

- This work uses domain-specific characteristics of ML circuits to:
    - Maximize the utilization of trace buffer space
    - Provide information that is meaningful to an engineer

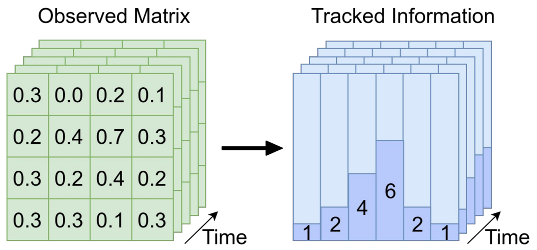


Debug Instruments

Overview of our instruments

- Many machine learning applications consist of large arrays (eg. activations or weights)
- Instruments track large arrays over time

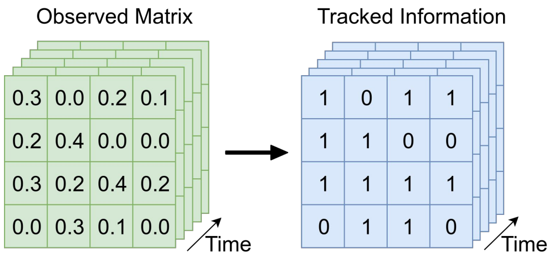
Distribution Instrument



- Creates a history of the distribution of the matrix we are observing over time (over multiple frames)
- In a CNN, a frame may represent all calculations corresponding to a single input image

Debug Instruments

Spatial Sparsity Instrument



- Stores an indication whether each element of the array is zero or non-zero.
- The same logic could also be used to track elements close to 1, another upper bound or NaN.

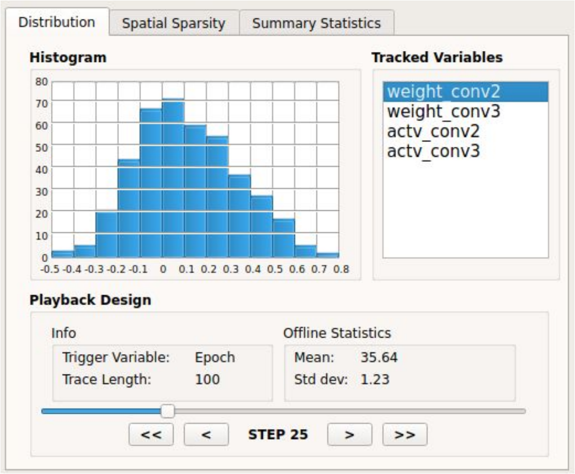
Summary Statistics Instrument

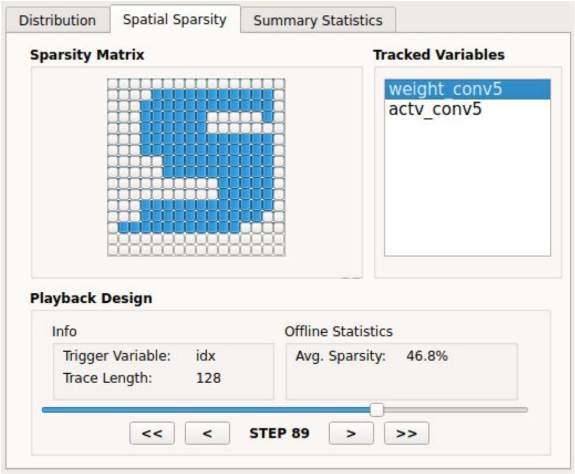
- Tracks only one kind of statistic (sparsity, mean, std. dev) per frame.

User Interface

Main Differences

- Stepping through frames instead of stepping through clock cycles (hardware-oriented debug) or lines of C code (HLS-debug)
- No access to raw values, we can trace the circuit for a longer period





Results

Configuration	Kernel	FMax (MHz)	LEs	# Traced Frames
Previous work	**32x28x28**	213.79	3391	0.124
	8x28x28	260.05	3324	0.498
	1x28x28	287.89	3167	3.985
Distribution Instrument - 32 bins	**32x28x28**	200.48	2867	195
	8x28x28	227.65	2834	223
	1x28x28	229.87	2676	284
Distribution Instrument - 128 bins	**32x28x28**	189.62	3670	48
	8x28x28	225.17	3600	55
	1x28x28	228.98	3488	71
Spatial Sparsity Instrument	**32x28x28**	200.46	2547	3
	8x28x28	211.13	2531	15
	1x28x28	214.70	2393	127
Summary Statistics Instrument - Sparsity	**32x28x28**	213.17	2557	6666
	8x28x28	258.75	2531	7692
	1x28x28	285.30	2390	10000
Proposed instruments combined	**32x28x28**	189.23	2930	3
	8x28x28	206.69	2927	14
	1x28x28	220.51	2786	87

Results

Configuration	Kernel	FMax (MHz)	LEs	# Traced Frames
Previous work	32x28x28	213.79	3391	0.124
	8x28x28	260.05	3324	0.498
	1x28x28	287.89	3167	3.985
Distribution Instrument - 32 bins	32x28x28	200.48	2867	195
	8x28x28	227.65	2834	223
	1x28x28	229.87	2676	284
Distribution Instrument - 128 bins	32x28x28	189.62	3670	48
	8x28x28	225.17	3600	55
	1x28x28	228.98	3488	71
Spatial Sparsity Instrument	32x28x28	200.46	2547	3
	8x28x28	211.13	2531	15
	1x28x28	214.70	2393	127
Summary Statistics Instrument - Sparsity	32x28x28	213.17	2557	6666
	8x28x28	258.75	2531	7692
	1x28x28	285.30	2390	10000
Proposed instruments combined	32x28x28	189.23	2930	3
	8x28x28	206.69	2927	14
	1x28x28	220.51	2786	87

Takeaway:

Domain-specific instrumentation allow us to store more **useful information** on-chip