

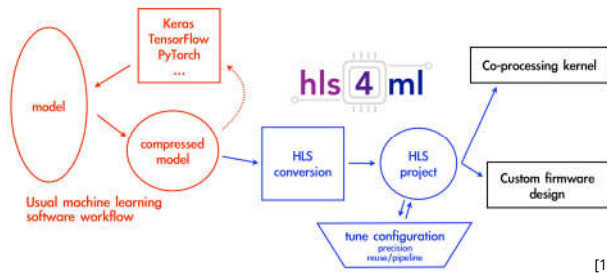
Benchmarking High Level Synthesis for Machine Learning Implementations versus Hand-optimized SystemVerilog

Waiz Khan, Caroline Johnson, Scott Hauck, Shih-Chieh Hsu, Geoff Jones

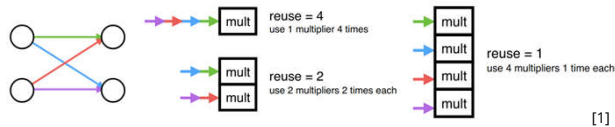


Introduction

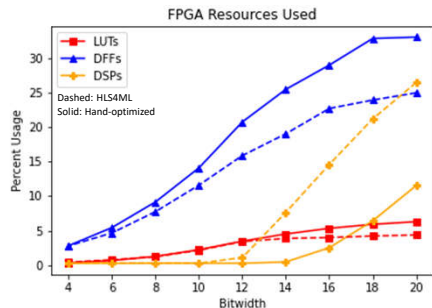
High Level Synthesis for Machine Learning (HLS4ML) enables rapid prototyping of Machine Learning models into hardware designs.



Performance can be optimized by adjusting parameters such as Compression, Precision, and Resource Reuse factors.



Are HLS4ML's implementations efficient compared to lower-level implementations?



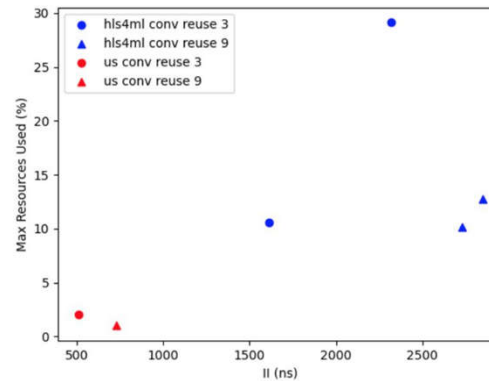
2D Conv Layer, Stride 2

Reuse Factor of 9:

PERFORMANCE	Min Period (ns)	Latency (cycles)	Latency (ns)	II (cycles)	II (ns)
HLS4ML - 256	6.2	592.5	3673.5	459.75	2850.45
HLS4ML - 128	6.9	463.5	3198.15	395.25	2727.225
Us	6.9	219	1511.1	105.75	729.675

Model	RESOURCES	LUTs	Total LUTs	DFFs	RAM	DSP
HLS4ML - 256		256	707	7073	11031	2
HLS4ML - 128		128		6502	8799	3.5
Us			231	2318	865	0.5
Available On Chip			693120	86640	1470	3600
HLS4ML 256				1.02%	12.73%	0.14%
HLS4ML				0.94%	10.16%	0.24%
Us				0.33%	1.00%	0.03%

Percentage of Max Resource Utilization vs. Iteration Interval



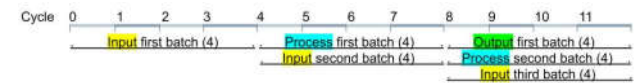
Hand-implemented Conv2D Layer (stride of 2) with Reuse Factor of 9 achieves better performance than the HLS4ML implementations.

- > 52.8% lower latency
- > 73.2% faster iteration interval
- > 64.3% fewer total LUT's used
- > 90.2% fewer DFFs used

Batch Normalization Layer

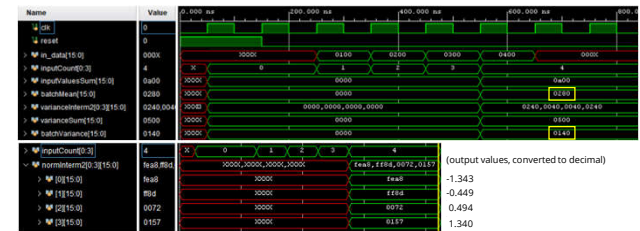
Currently under development, small scale model functional

Batch Norm Layer algorithm:



- > Batch size of 4, pipelined to three major stages, each taking four cycles
- > Values processed in the following order: (each step takes one cycle) Batch Mean, Batch Variance, Normalize value, Scale & Shift
- > Pipelined for efficiency to allow for parallel usage of resources

Small Scale Model Results: (values are in fixed point, 8 integer bits, 8 fraction bits)



Conclusion

- > The possibility to improve CONV2D implementation in HLS4ML to be faster or efficient is demonstrated. The lower-level implementation required fewer resources to produce a model with lower latency.
- > Batch Normalization layer can be implemented efficiently in hardware but will require large LUTs to accelerate some parts of the computation.

Next Steps

- > Implement an HLS4ML-inspired SystemVerilog implementation of Conv2D, stride 2 to improve performance
- > Implement a scaled-up batch norm layer, to compare with HLS4ML

References

[1] J. Duarte et al 2018 JINST 13 P07027

