Framework for creating performance model of AI algorithms for early architecture exploration

Amit Dudeja, Sr R & D Engineer, Synopsys India Pvt. Ltd.
Amit Tara, Sr R & D Engineer, Synopsys India Pvt. Ltd.
Amit Garg, Principal Engineer, Synopsys India Pvt. Ltd.
Tushar Jain, R & D Engineer, Synopsys India Pvt. Ltd.

accellera SYSTEMS INITIATIVE™

SYNOPSYS®

2019 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION INDIA

Agenda

- AI Market and Technology Trends
- AI SoC Design Challenges
- Shift Left Architecture Exploration and Optimization
- AI Exploration Framework
- NN driven Deep Learning Architecture Optimization
  - Case Study Resnet18 with NVDLA

AI Market and Technology Trends

New Neural Network algorithms

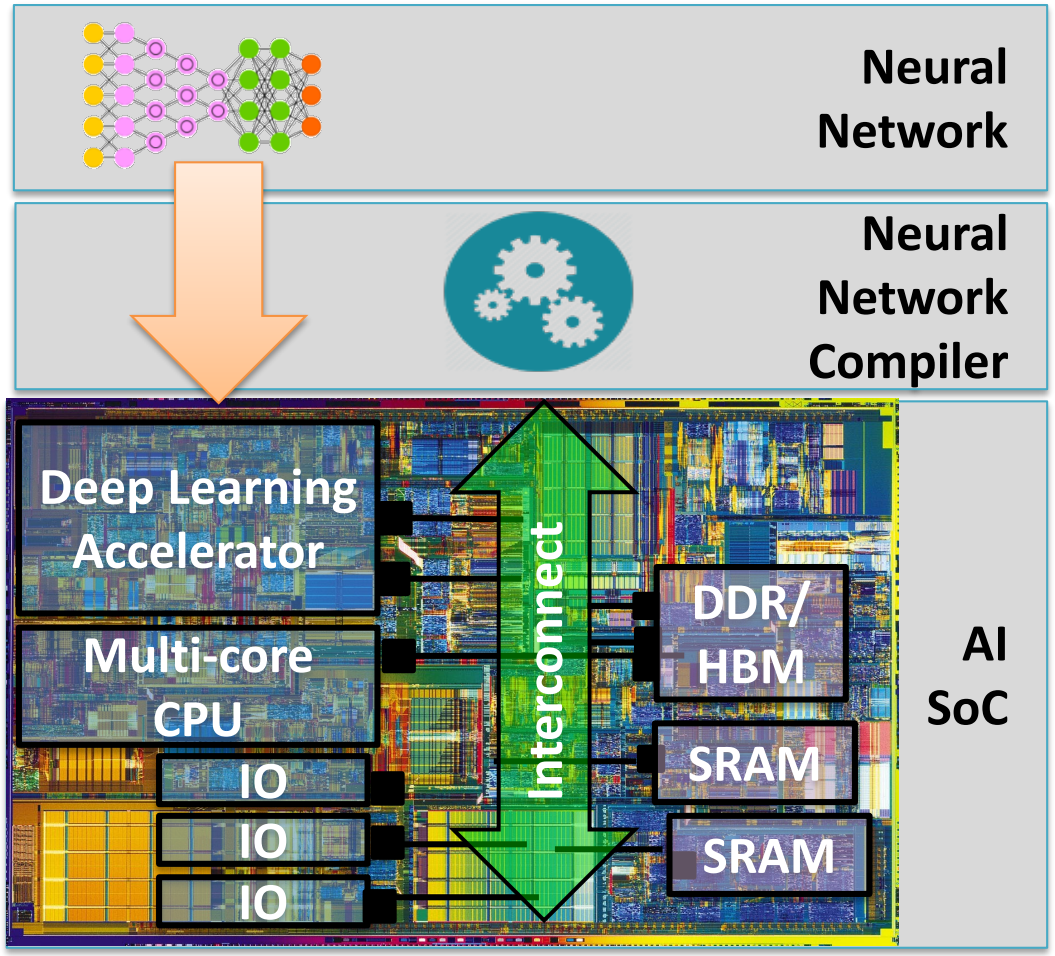
- Higher accuracy, lower size and less processing
- But: less data re-use, less cycles per byte

Neural Network Compiler optimizations

- Loop-tiling, -unrolling, and –parallelization
- Splitting and fusing of Neural Network layers
- Memory layout optimization across layers
- Optimized code generation to utilize available hardware accelerators

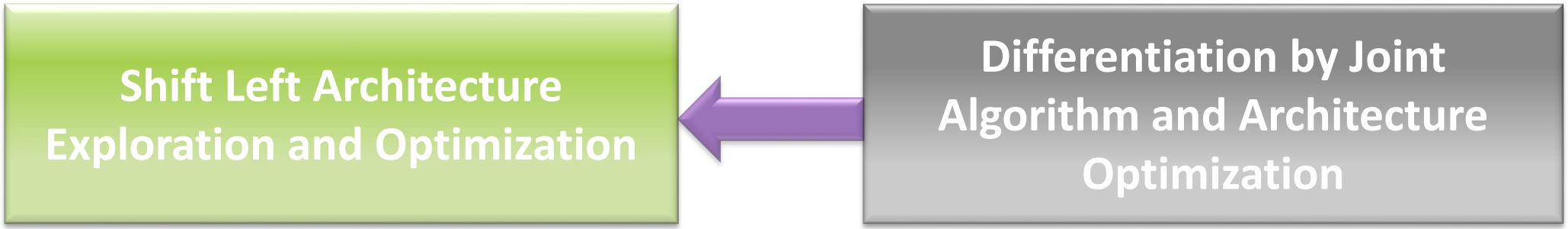
Deep Learning Accelerator optimizations

- Schedule workload on parallel hardware engines
- Optimize/reduce data transfers to & from memory

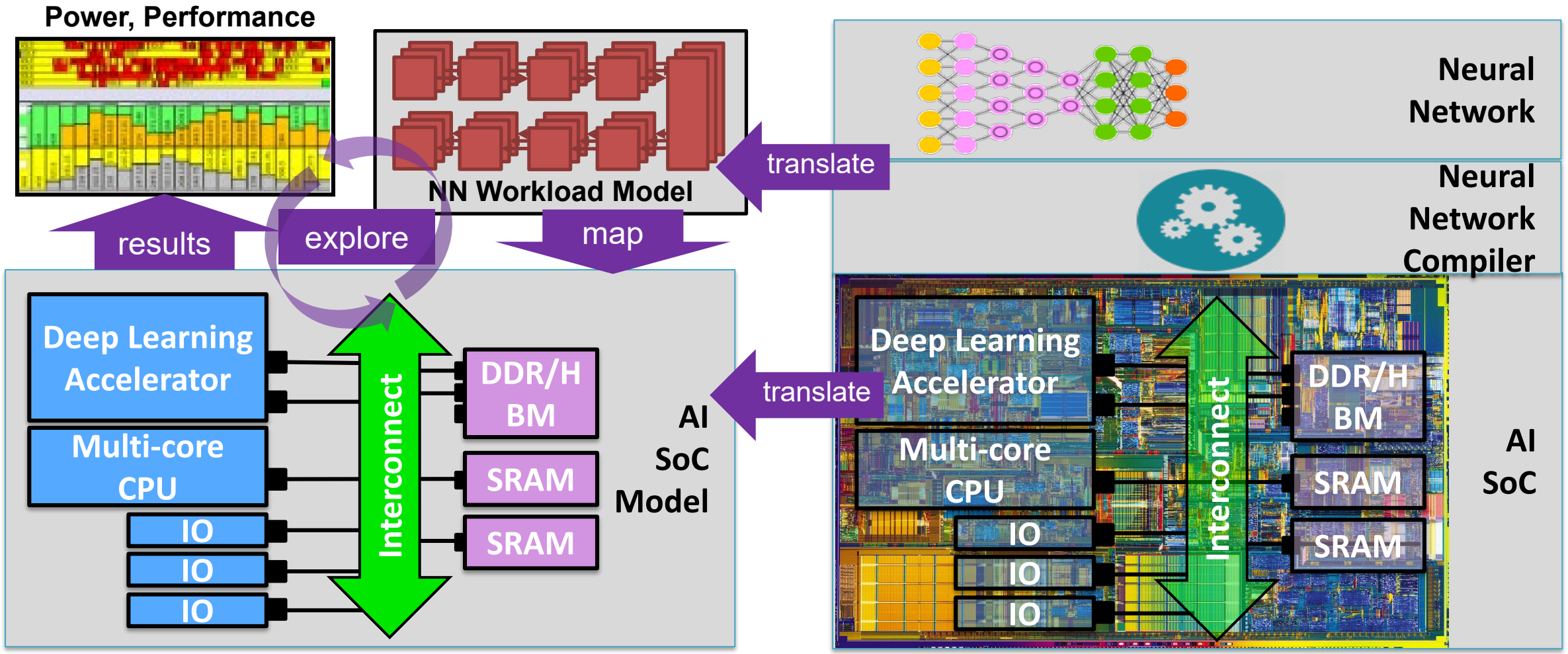


AI SoC Design Challenges

- **Choosing the right algorithm and architecture: CPU, GPU, FPGA, vector DSP, ASIP**
  - CNN graphs are evolving fast, need short time to market and cannot optimize for one single graph
  - Joint design of algorithm, compiler, and target architecture
  - Joint optimization of power, performance, accuracy, and cost
- **Highly parallel compute drives memory requirements**
  - High on-chip and chip to chip bandwidth at low latency
  - High memory bandwidth requirements for parameters and layer to layer communication
- **Performance analysis requires realistic workloads to consider dynamic effects**
  - Scheduling of AI operators on parallel processing elements
  - Unpredictable interconnect and memory access latencies



Shift Left Architecture Exploration and Optimization



Objective

- Systematic creation of AI performance model in an automated manner.
  - Model AI operations like convolution, batchNorm etc
  - Generate complete topology of AI algorithm.
- Construction of generic and configurable AI Centric Hardware Subsystem
  - Model scalable AI engines for compute and memory load.
  - Build hierarchical subsystems for complex designs.

PA Ultra AI Exploration Framework

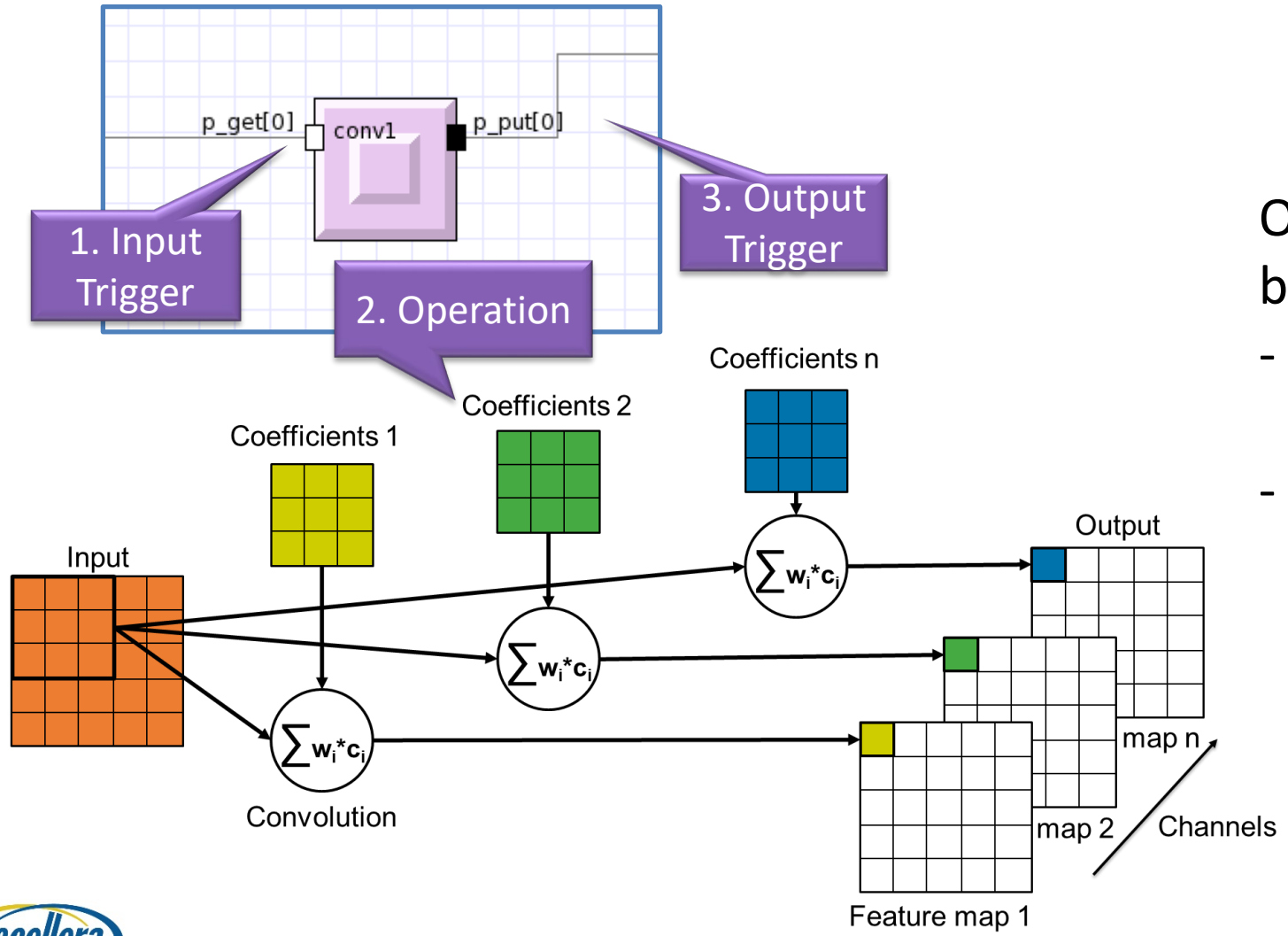
PA Ultra AI Exploration Framework

- AI Operator Library
  - Workload models for different operators (layers) for AI CNN algorithm development

- Automated generation of workload model for the Neural Networks

- AI centric HW architecture Engine

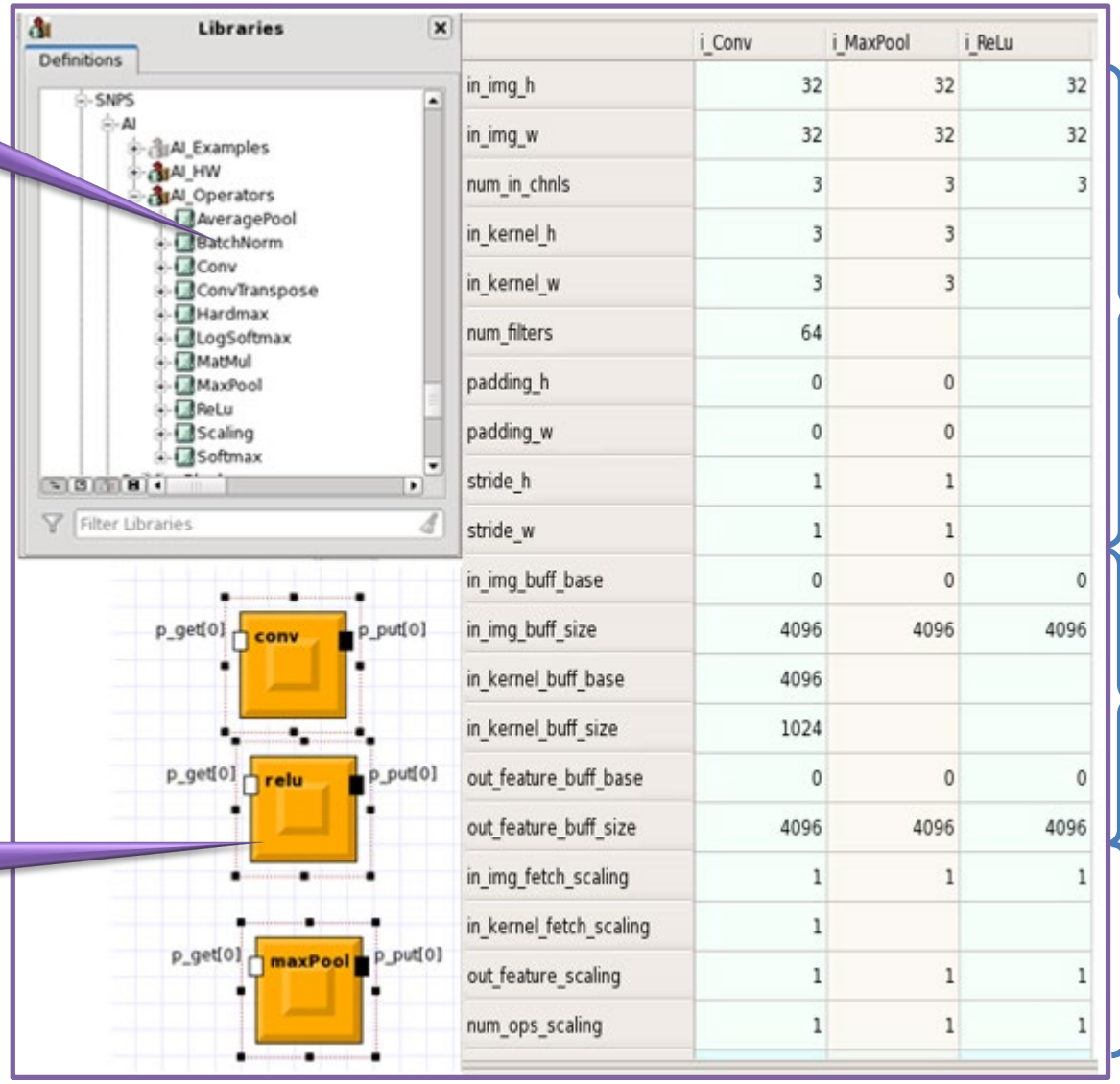
Workload Model of a Convolution Layer



Once the Convolution Operator block gets triggered:

- Input frame and coefficient are read in parallel.
- Once the necessary input data is read, block process the convolutions and write backs the resulting feature maps.

AI Operator Library



	i_Conv	i_MaxPool	i_ReLu
in_img_h	32	32	32
in_img_w	32	32	32
num_in_chnls	3	3	3
in_kernel_h	3	3	
in_kernel_w	3	3	
num_filters	64		
padding_h	0	0	
padding_w	0	0	
stride_h	1	1	
stride_w	1	1	
in_img_buff_base	0	0	0
in_img_buff_size	4096	4096	4096
in_kernel_buff_base	4096		
in_kernel_buff_size	1024		
out_feature_buff_base	0	0	0
out_feature_buff_size	4096	4096	4096
in_img_fetch_scaling	1	1	1
in_kernel_fetch_scaling	1		
out_feature_scaling	1	1	1
num_ops_scaling	1	1	1

Library of Operators

Design Canvas

User Parameters

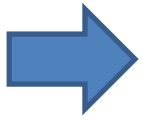
Exploration Parameters

S/W Optimization Parameters

PA Ultra AI Exploration Framework

- AI Operator Library
  - Workload models for different operators (layers)

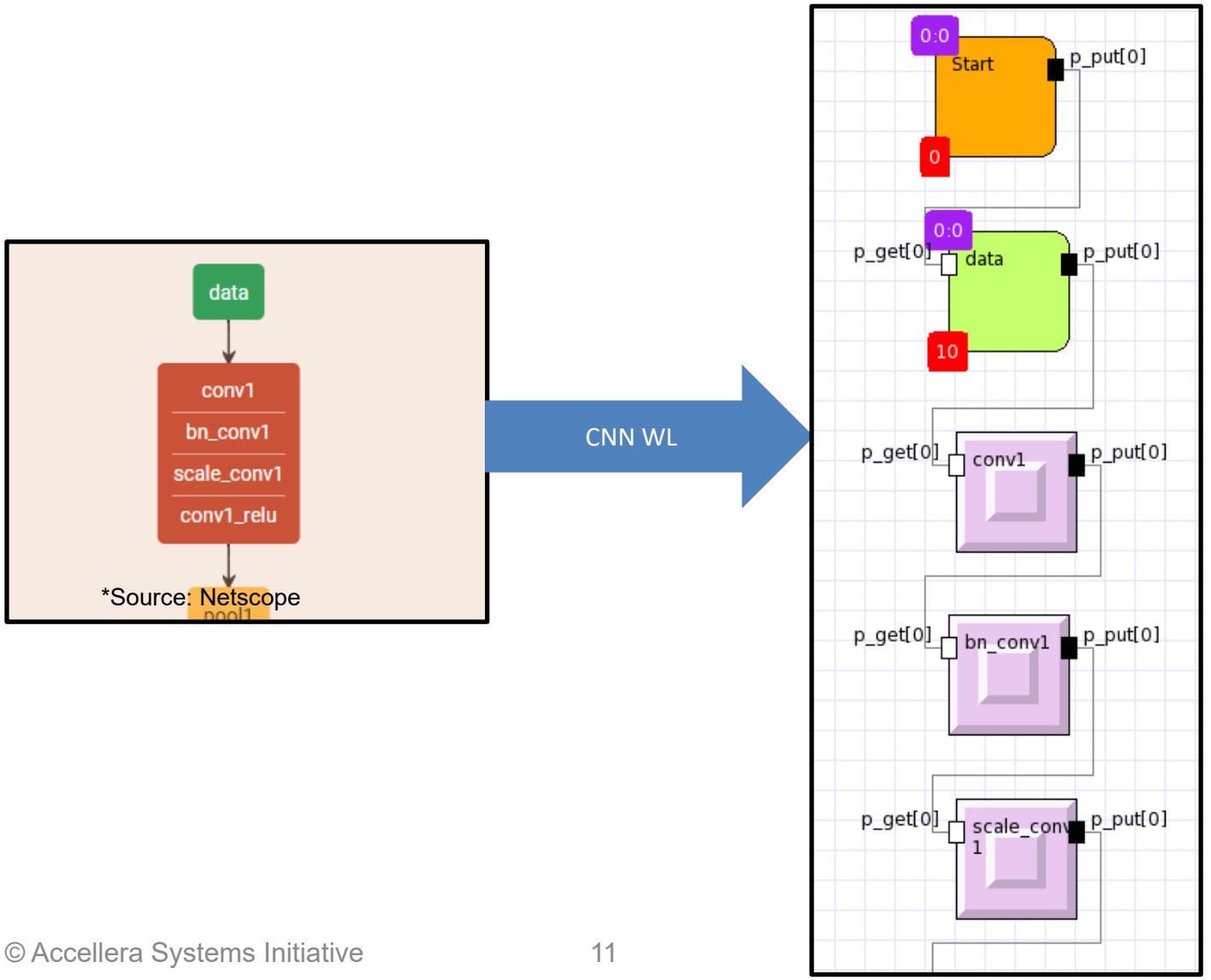
  for AI CNN algorithm development



Automated generation of workload model for Neural Networks

- AI centric HW architecture Engine

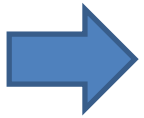
Creation of performance model of AI algorithms



PA Ultra AI Exploration Framework

- AI Operator Library
  - Workload models for different operators (layers) for AI CNN algorithm development

- Automated generation of workload model for Neural Networks



AI centric HW architecture Engine

PA Ultra Hardware IP Models Library

Traffic, Processors, RTL

- Task-based and trace-based workload models
- Cycle accurate processor for ARM, ARC, Tensilica, CEVA
- RTL Co-simulation/emulation

arm tensilica ARC Synopsys CEVA

Interconnect Models

Generic:
- SBL-TLM2-FT (AXI)
- SBL-GCCI (ACE, CHI)

IP Specific:
- Arteris FlexNoC & Ncore
- Arm AHB/APB
- Arm PL300
- Arm SBL-301
- Arm SBL-400
- Synopsys DW AXI

SYNOPSYS arm ARTERIS IP

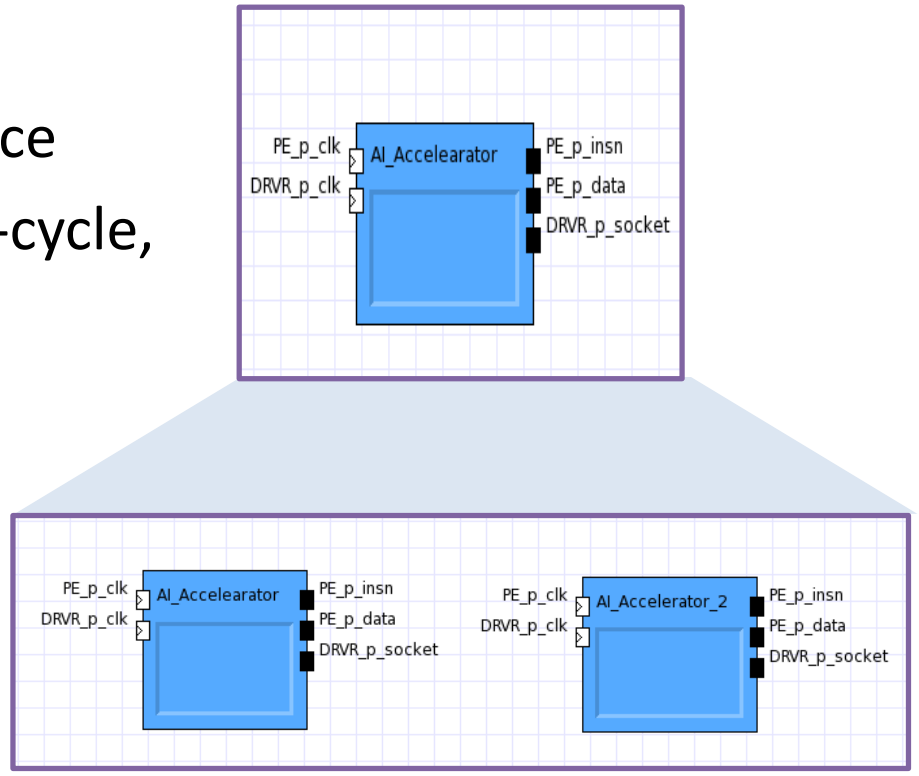
Memory Subsystems

- Generic multiport memory controller (GMPMC)
- DesignWare uMCTL2 memory controller
- DesignWare LPDDR5 memory controller
- Co-simulate with RTL

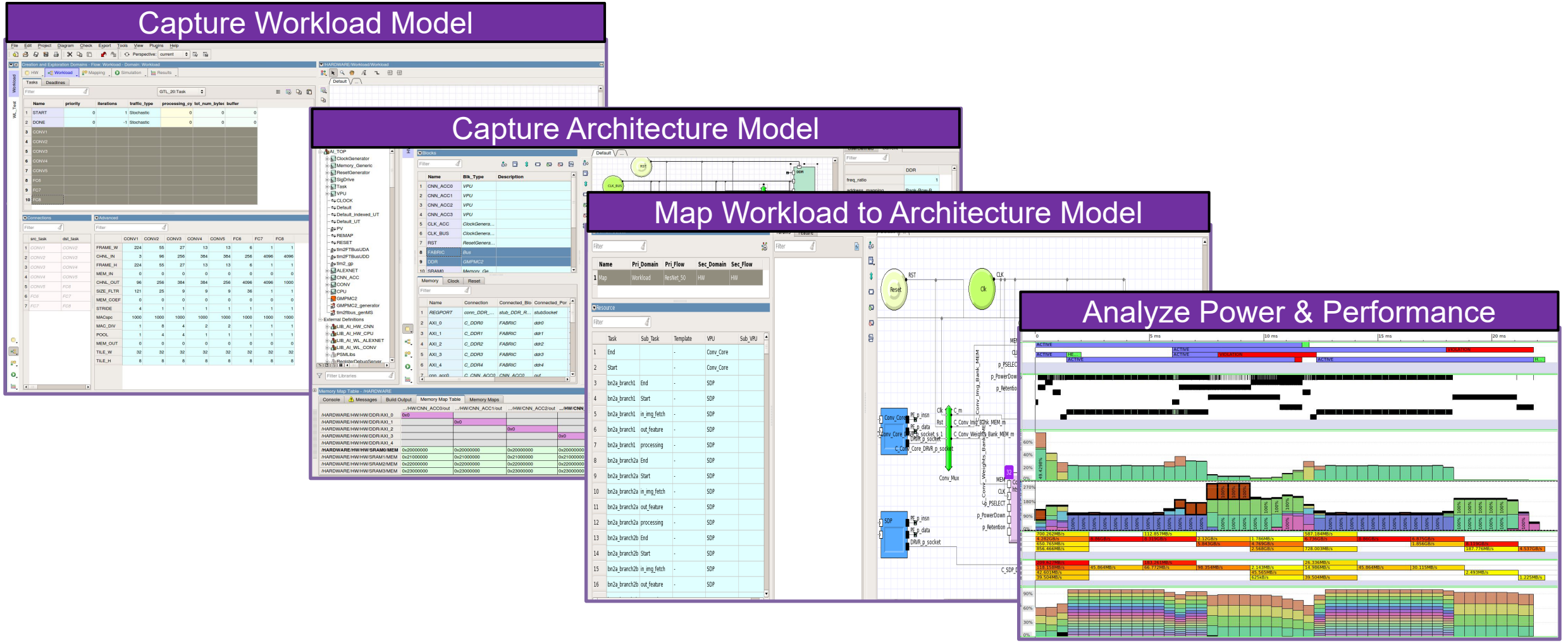
SYNOPSYS

Creation of AI Centric Hardware Engine

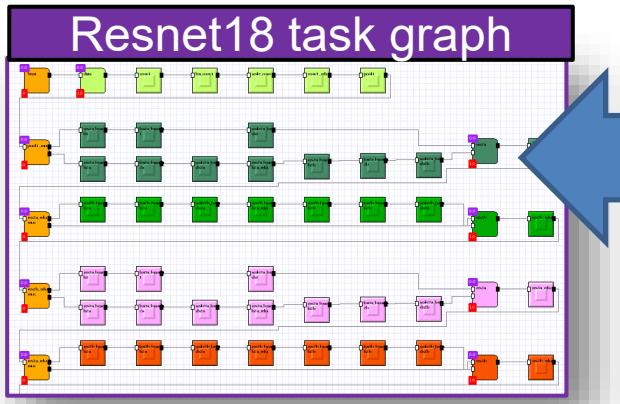
- AI-centric hardware engine caters to the compute and memory requirements of AI operations.
- Can be constructed hierarchically by encapsulating n-sub engines underneath.
- Can be characterized with performance related attributes like operations-per-cycle, stochastic cache, branch prediction, pipeline depth etc.



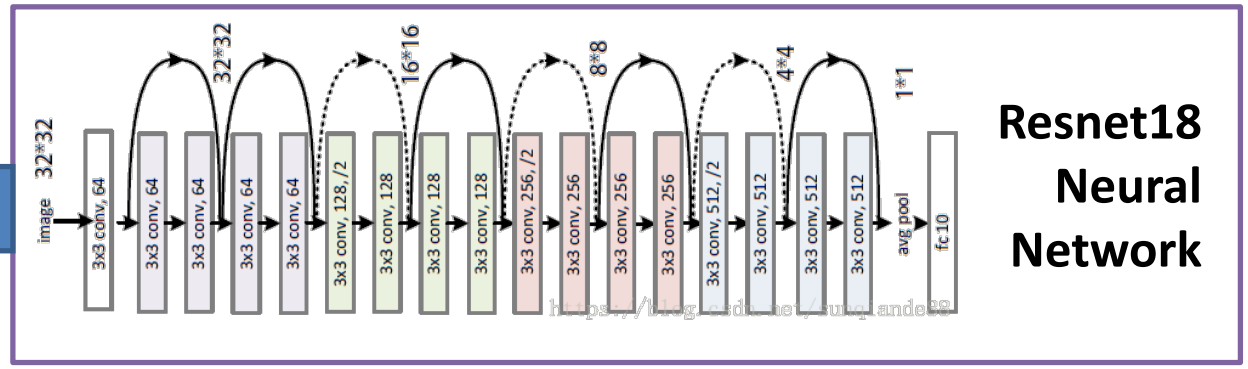
Platform Architect Ultra



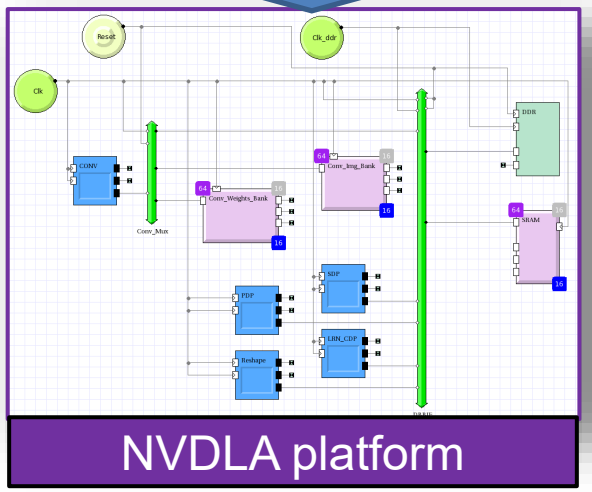
Case Study: Resnet-18 with NVDLA



generate



map



Goals:

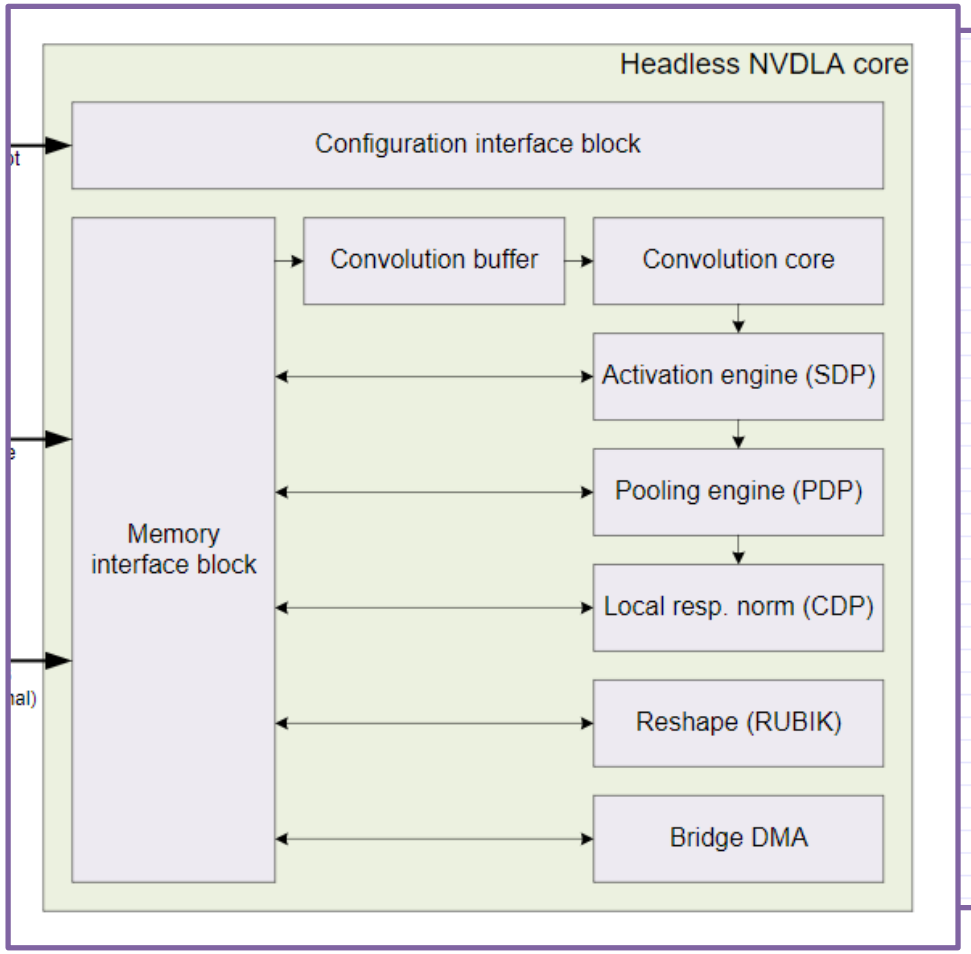
① 5 ms latency, ② minimize power, ③ minimize energy

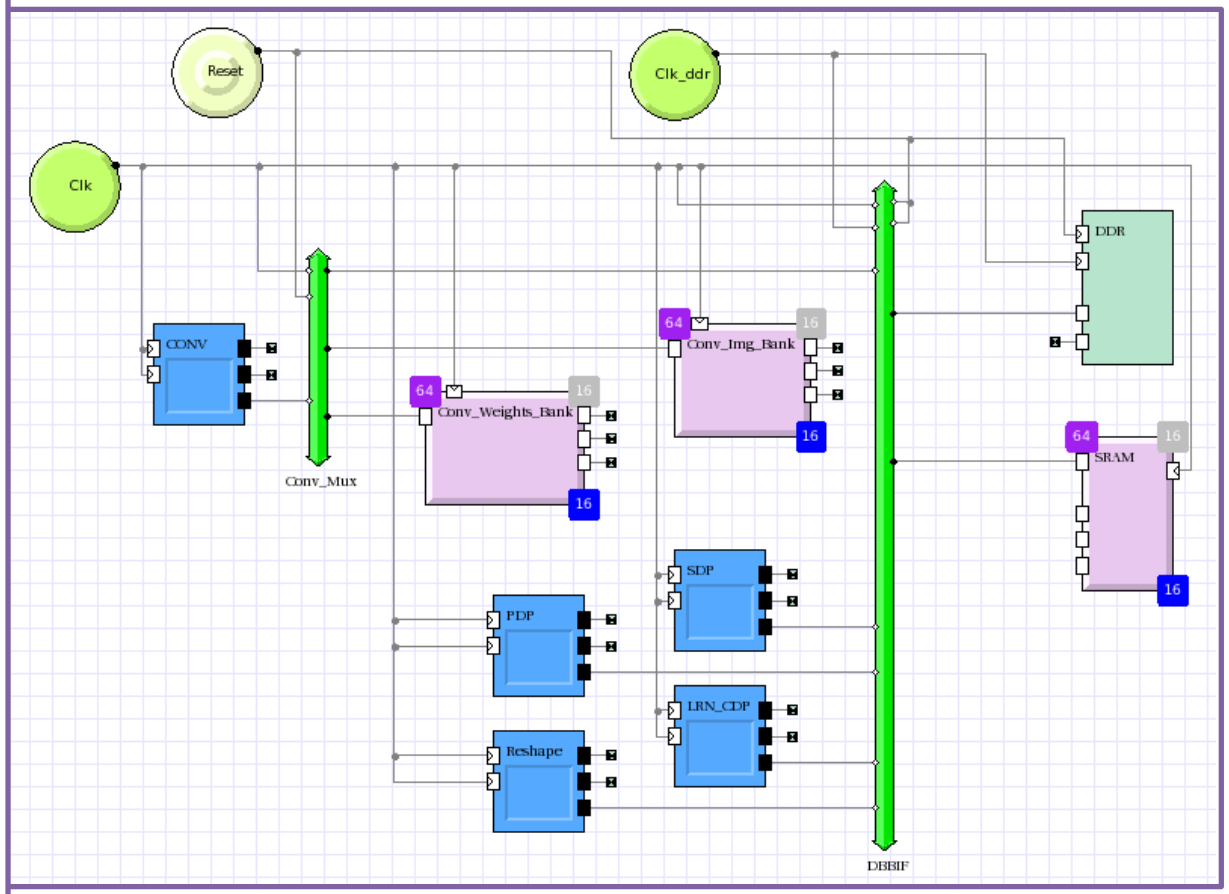
Optimize Hardware configuration:

- SIMD width
- Burst size, outstanding transactions
- speed of DDR memory and of data path

ResNet-18 Workload model generated with AI-XP

NVDLA

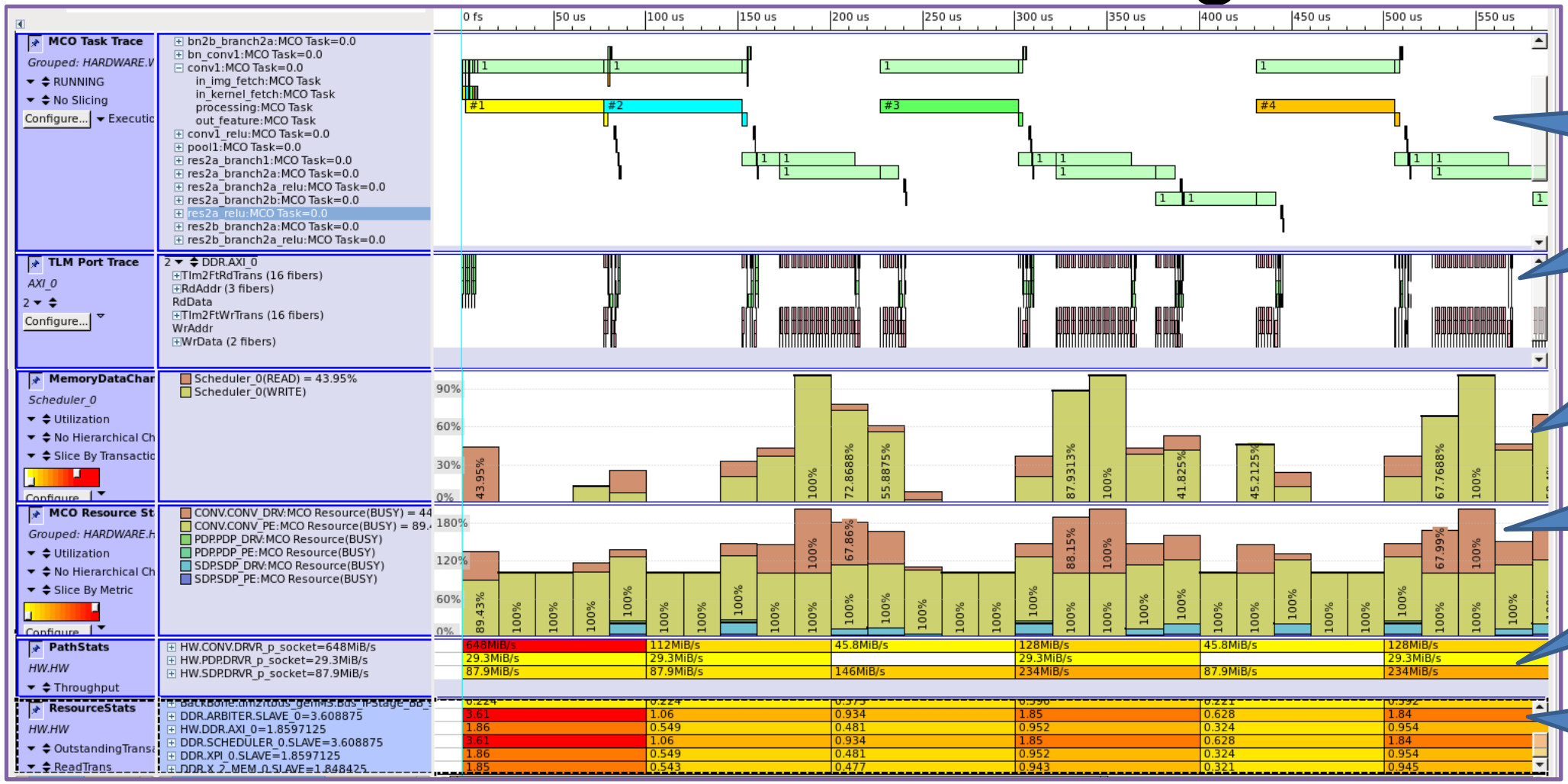

Headless NVDLA core
Configuration interface block
Memory interface block
Convolution buffer
Convolution core
Activation engine (SDP)
Pooling engine (PDP)
Local resp. norm (CDP)
Reshape (RUBIK)
Bridge DMA


Reset
Clk_ddr
Clk
CONV
Conv_Mux
Conv_Weights_Bank
Conv_Img_Bank
DDR
SRAM
SDP
PDP
LRN_CDP
Reshape
DBBIF

NVDLA Initial Configuration

- Burst size: 16
- Max Outstanding transactions: 8
- Clock frequency of data path: 1GHz
- SIMD width: 16 operations per cycle

Results of Initial Configuration



Task trace

Transaction trace

DDR utilization

Resource utlization

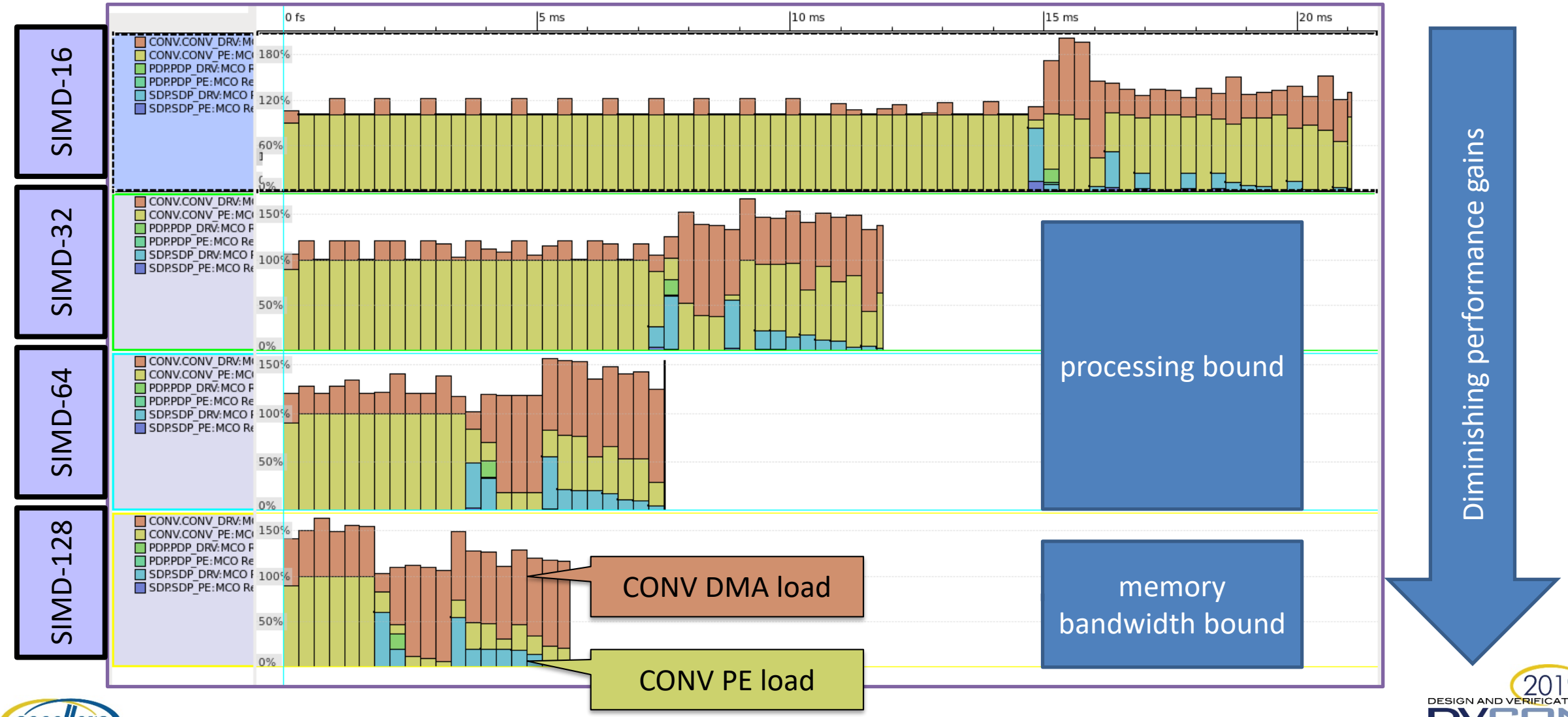
Throughput

Outstanding transactions

Performance limited by processing, use wider SIMD data path

Impact of SIMD Width on Performance

Resource Utilization of CONV Datapath (yellow), CONV DMA (red) and other components



Resnet 18 Example Sweep

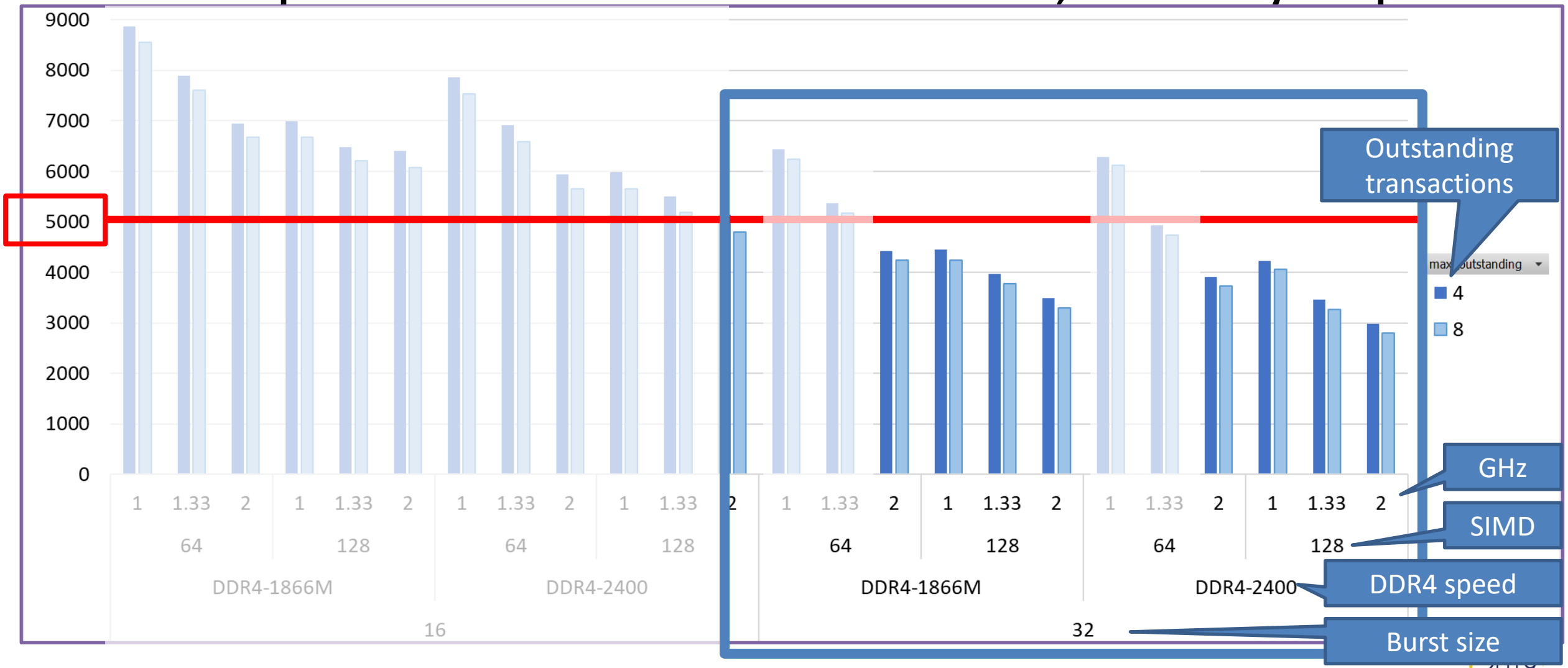
Goal: 5ms latency, minimize power & energy

	Name	simtime_us	...outstanding	..._sz_in_bytes	...speed_bin	...Clk/perioc	...opc
23	run_b32_opc64_clk05_DDR4_1866M_os4	4416.966	4	32	DDR4-18...	0.5	64
24	run_b32_opc64_clk05_DDR4_1866M_os8	4235.459	8	32	DDR4-18...	0.5	64
25	run_b16_opc128_clk10_DDR4_2400_os4	5990.249	4	16	DDR4-2400	1	128
26	run_b16_opc128_clk10_DDR4_2400_os8	5657.129	8	16	DDR4-2400	1	128
27	run_b16_opc128_clk10_DDR4_1866M_os4	6994.128	4	16	DDR4-18...	1	128
28	run_b16_opc128_clk10_DDR4_1866M_os8	6676.136	8	16	DDR4-18...	1	128
29	run_b16_opc128_clk075_DDR4_2400_os4	5509.487	4	16	DDR4-2400	0.75	128
30	run_b16_opc128_clk075_DDR4_2400_os8	5189.458	8	16	DDR4-2400	0.75	128
31	run_b16_opc128_clk075_DDR4_1866M_os4	6485.257	4	16	DDR4-18...	0.75	128
32	run_b16_opc128_clk075_DDR4_1866M_os8	6207.377	8	16	DDR4-18...	0.75	128
33	run_b16_opc128_clk05_DDR4_2400_os4	5141.269	4	16	DDR4-2400	0.5	128
34	run_b16_opc128_clk05_DDR4_2400_os8	4797.354	8	16	DDR4-2400	0.5	128
35	run_b16_opc128_clk05_DDR4_1866M_os4	6402.813	4	16	DDR4-18...	0.5	128
36	run_b16_opc128_clk05_DDR4_1866M_os8	6080.660	8	16	DDR4-18...	0.5	128
37	run_b32_opc128_clk10_DDR4_2400_os4	4228.984	4	32	DDR4-2400	1	128
38	run_b32_opc128_clk10_DDR4_2400_os8	4066.654	8	32	DDR4-2400	1	128
39	run_b32_opc128_clk10_DDR4_1866M_os4	4444.511	4	32	DDR4-18...	1	128
40	run_b32_opc128_clk10_DDR4_1866M_os8	4236.462	8	32	DDR4-18...	1	128
41	run_b32_opc128_clk075_DDR4_2400_os4	3464.214	4	32	DDR4-2400	0.75	128
42	run_b32_opc128_clk075_DDR4_2400_os8	3261.505	8	32	DDR4-2400	0.75	128
43	run_b32_opc128_clk075_DDR4_1866M_os4	3963.731	4	32	DDR4-18...	0.75	128
44	run_b32_opc128_clk075_DDR4_1866M_os8	3769.144	8	32	DDR4-18...	0.75	128
45	run_b32_opc128_clk05_DDR4_2400_os4	2981.385	4	32	DDR4-2400	0.5	128
46	run_b32_opc128_clk05_DDR4_2400_os8	2795.655	8	32	DDR4-2400	0.5	128
47	run_b32_opc128_clk05_DDR4_1866M_os4	3483.204	4	32	DDR4-18...	0.5	128
48	run_b32_opc128_clk05_DDR4_1866M_os8	3301.767	8	32	DDR4-18...	0.5	128

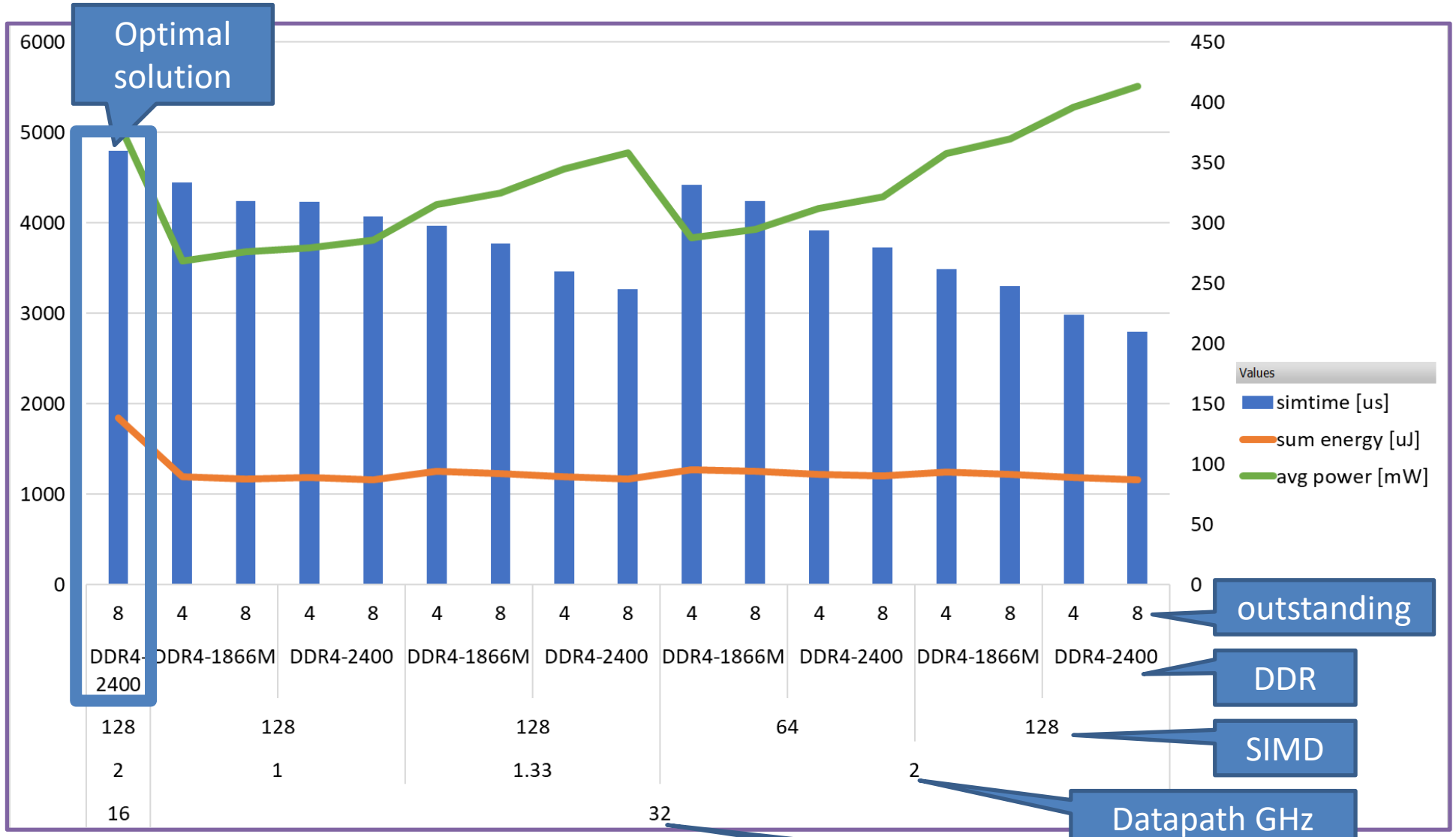
Sweep parameters

- Burst size: 16, 32
- Outstanding transactions: 4, 8
- DDR memory speed: DDR4-1866, DDR4-2400
- Clock frequency of data path: 1, 1.33, 2GHz
- SIMD width: 64, 128 operations per cycle

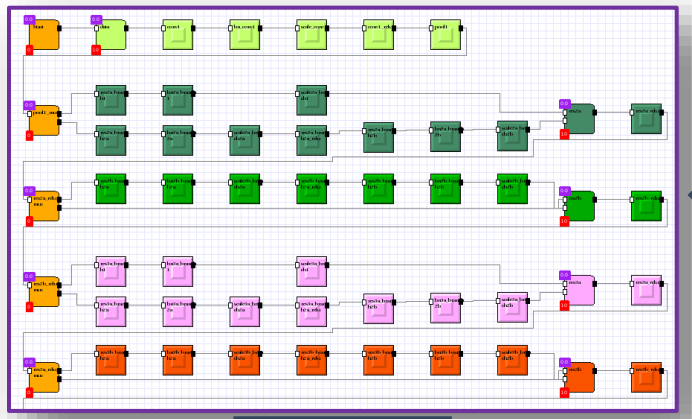
Sweep Over Hardware Parameters, Latency in µs



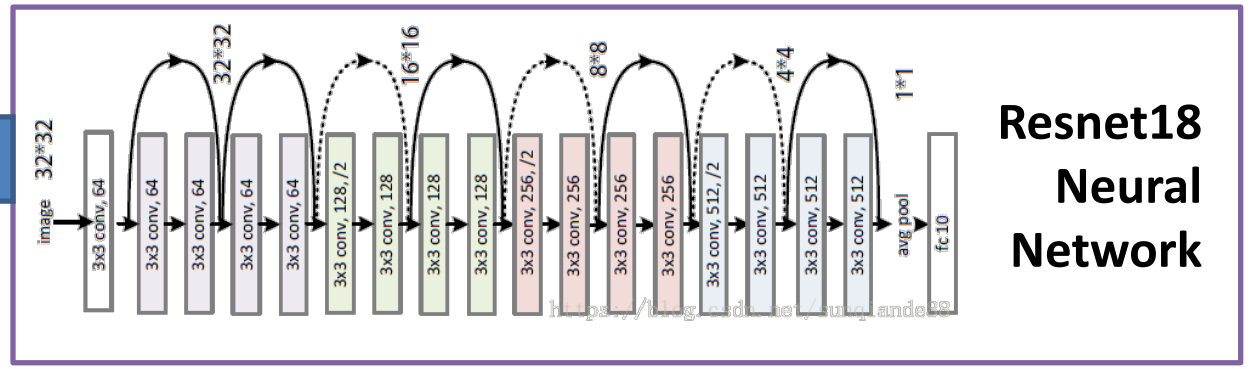
Power/Performance/Energy Trade-off Analysis



Example: Resnet-18 with NVDLA

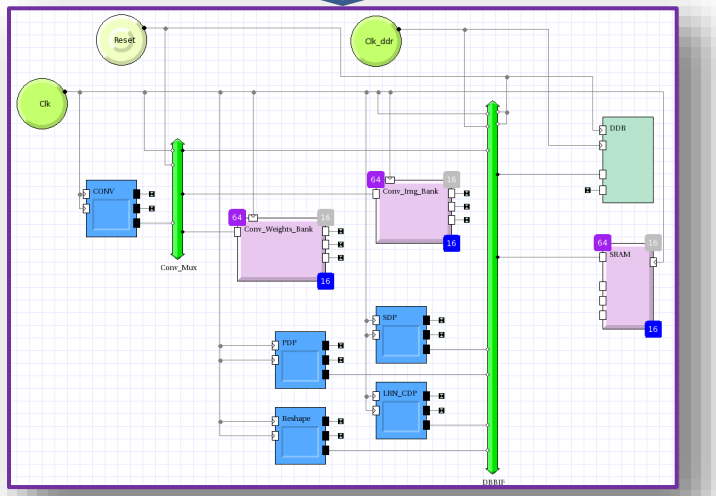


generate



Resnet18 Neural Network

map



Goal:

– 5 ms latency, minimize energy

Optimize Hardware configuration:

– SIMD width: 128 operations per cycle
– Burst size: 32 bytes
– outstanding transactions: 4
– speed of DDR memory: DDR4-1866
– speed of data path: 1GHz

Summary

- Demonstrated a framework to perform exploration studies in the field of AI.
- Ease of automatically creating a workload model out of an AI algorithm.
- Goal directed power and performance study to achieve an inference latency of 5ms with optimized power consumption.

Questions