UNIVERSITY OF CALIFORNIA SANTA CRUZ

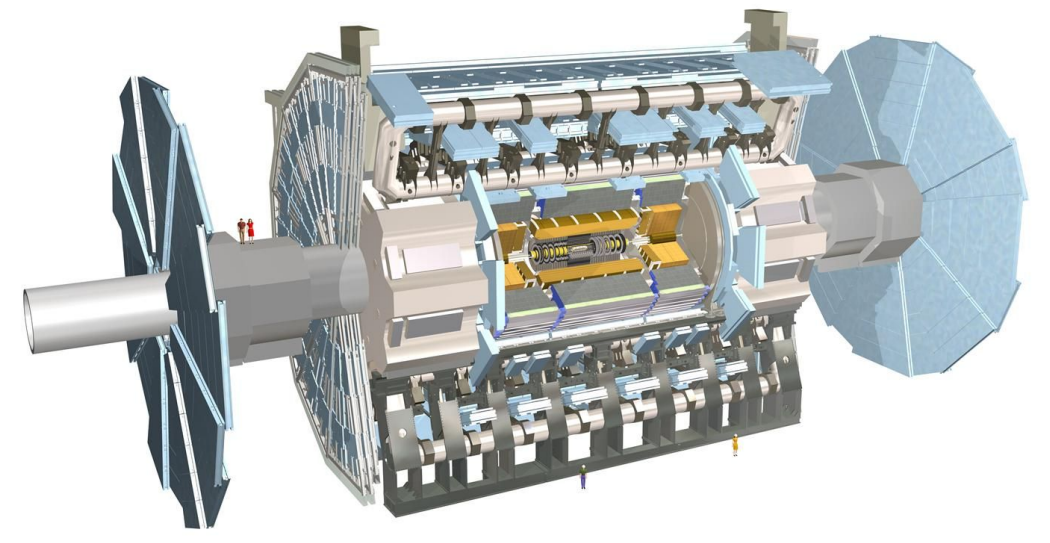
Convolutional Neural Networks for Track Reconstruction on FPGAs

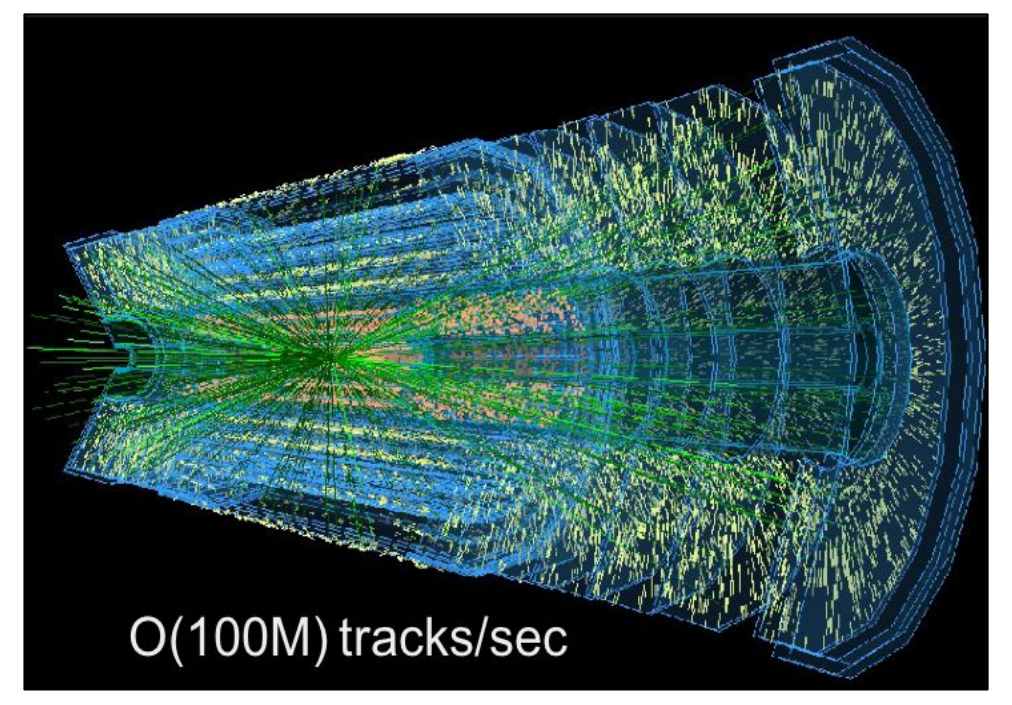
Thomas Boser[1]; Paolo Calafiura[2]; Ian Johnson[2]
[1]University of California, Santa Cruz, [2]Lawrence Berkeley National Laboratory

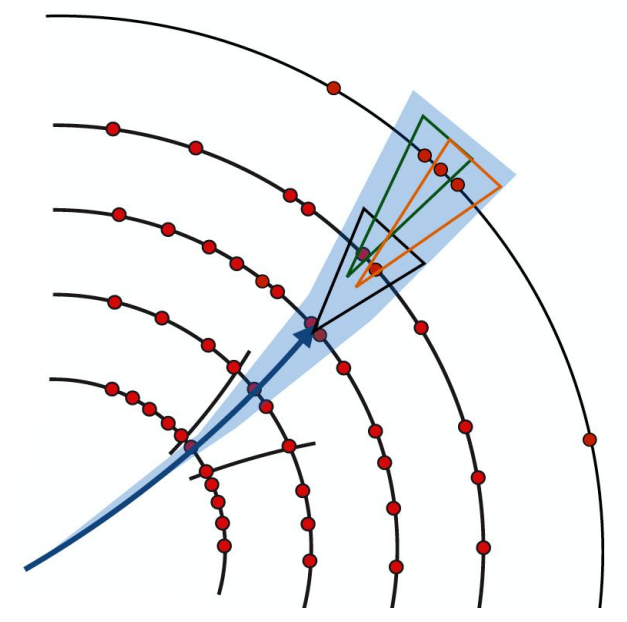
BERKELEY LAB

Motivation

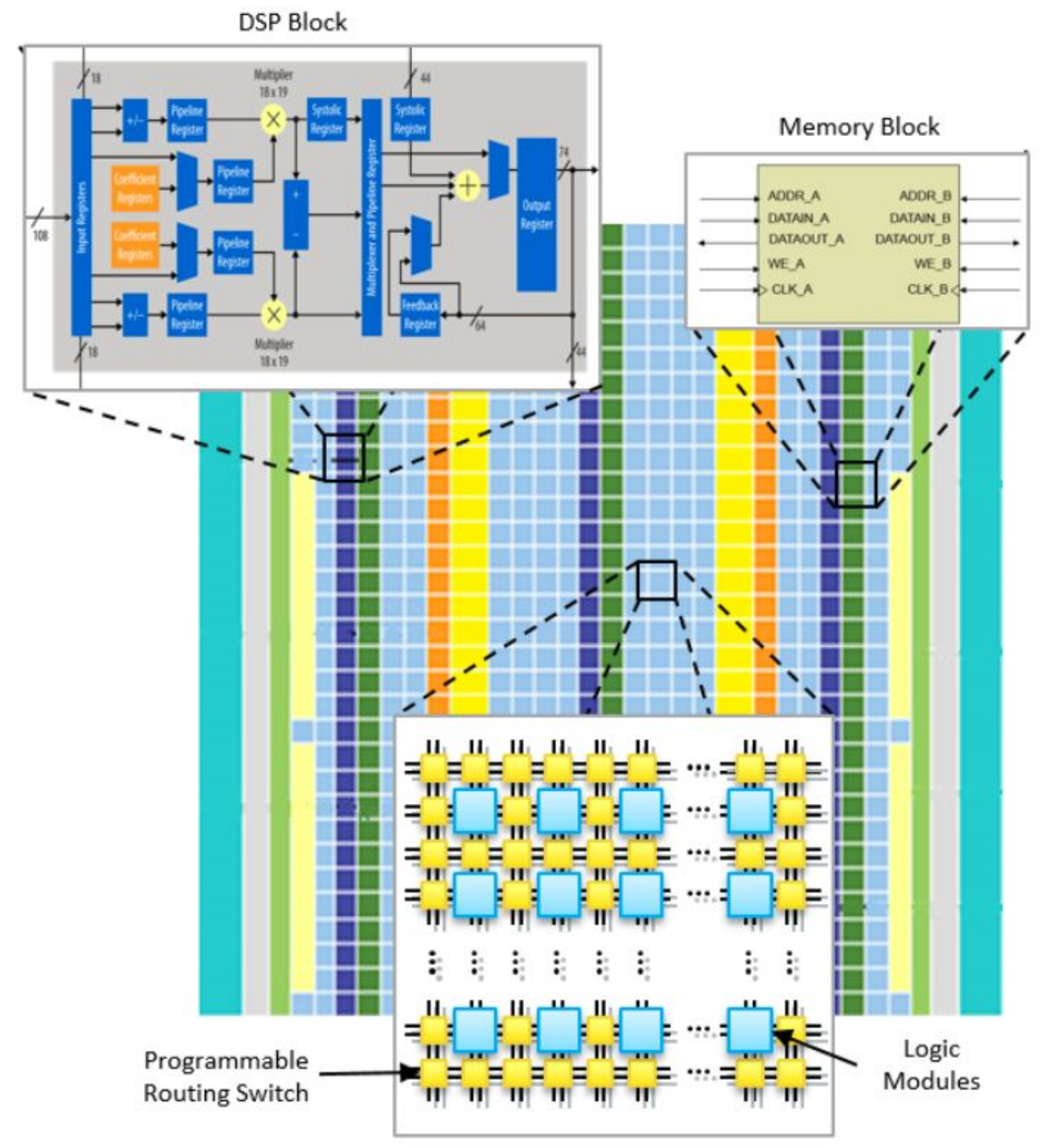
- LHC Particle Tracking as a “Connecting the dots” problem:
  - Given dataset with O(10^5) 3D space-points belonging to O(10^3) particle tracks, predict which space-points belong to the same track.
- Performance requirements:
  - 100KHz rate, with ~5 μs latency per prediction.
- FPGA good match:
  - guaranteed latency, high throughput
  - already used by LHC experiments for similar applications






Credit: Andy Salzburger

Methods and Materials



- FPGAs are reprogrammable integrated circuits.
- Logic blocks can be used to configure low level operations such as bit masking, shifting, and addition.
- Support highly parallel and pipelined algorithm implementations with guaranteed latency.

How are FPGAs programmed?

- Hardware Description Languages

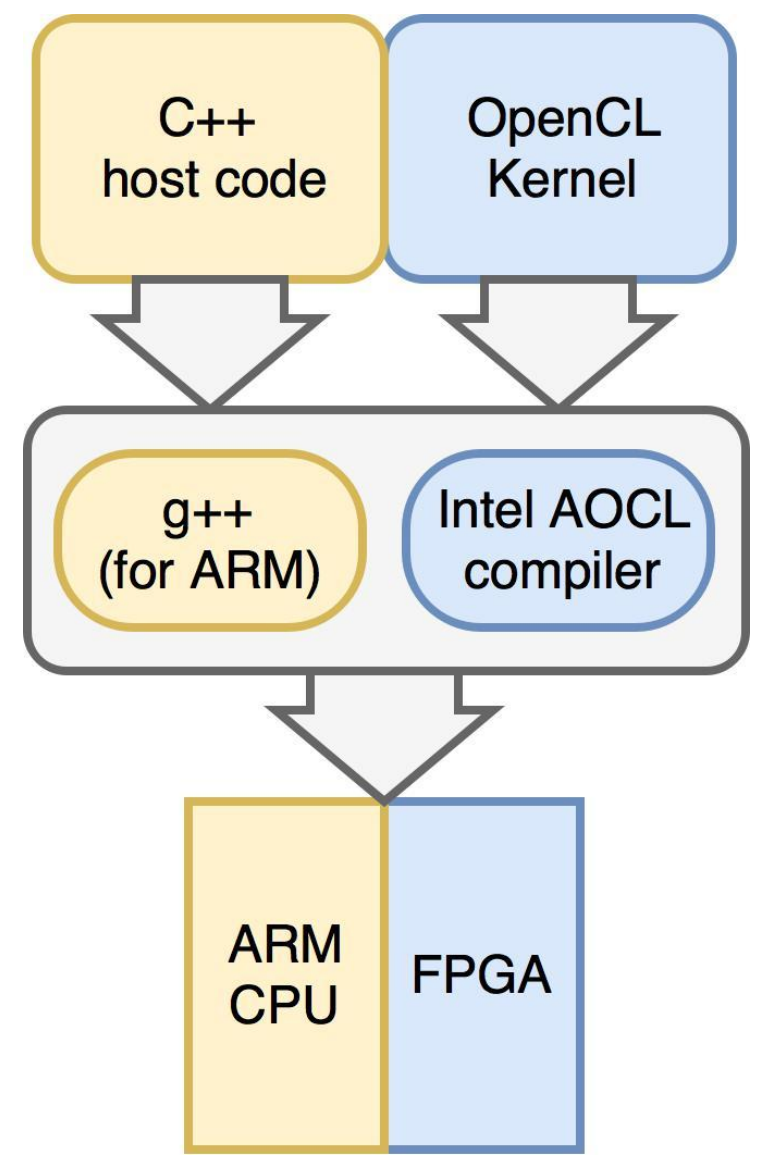
HDLs are programming languages which describe electronic circuits.

- High Level Synthesis

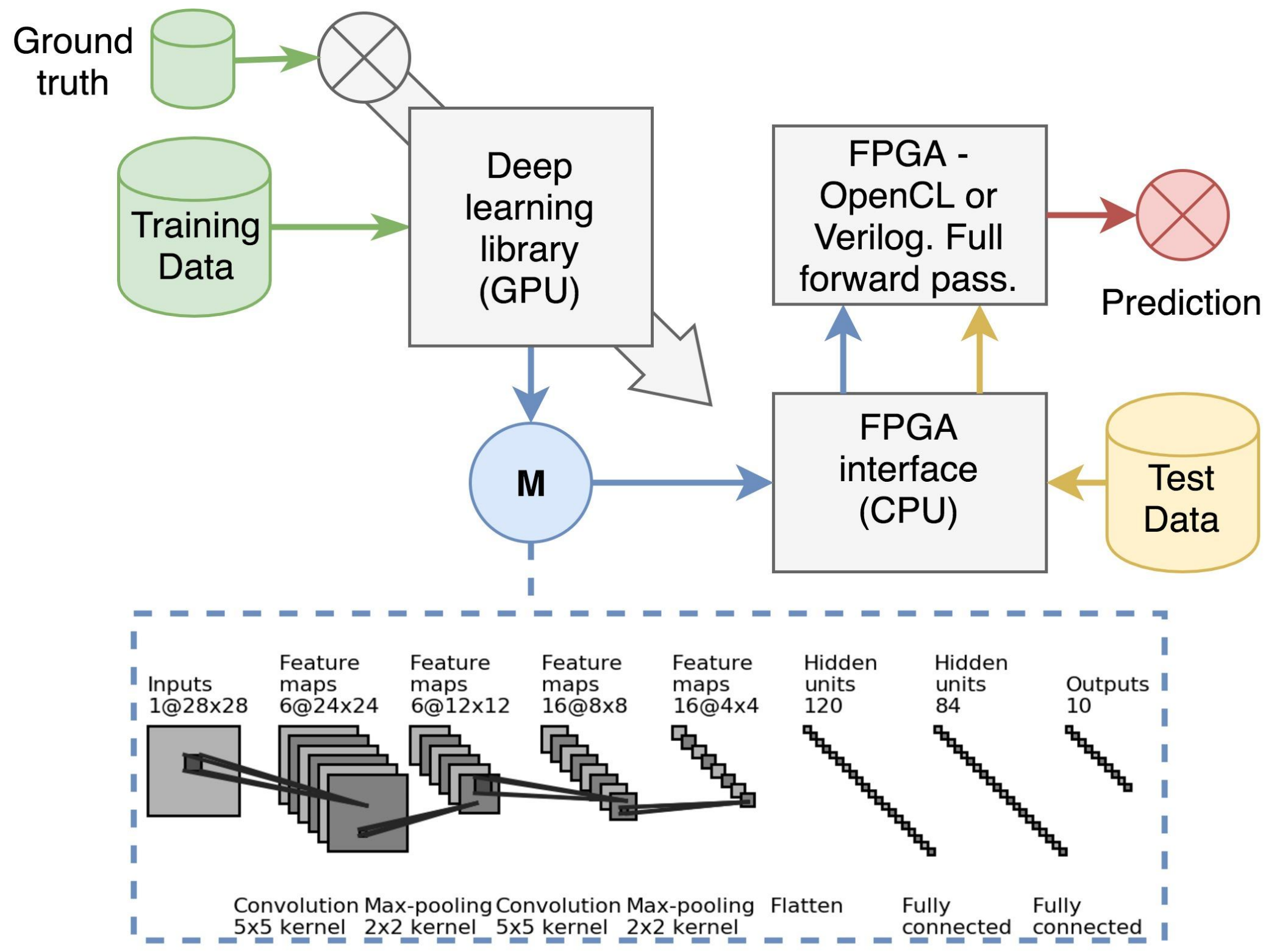
Generation of HDL from a higher level language, often C or C++.

- OpenCL

Computing framework similar to CUDA which is supported by some FPGAs.



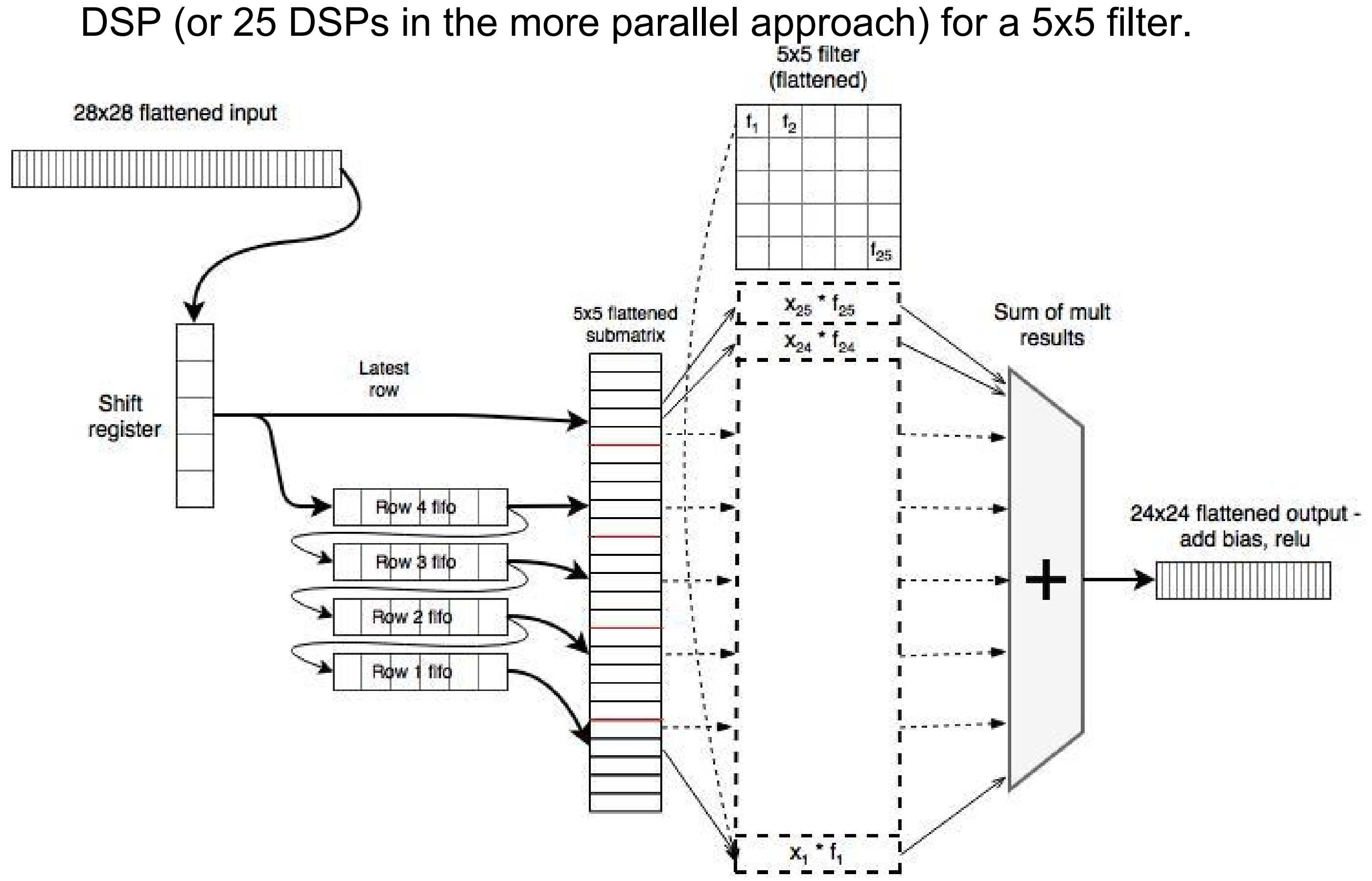
Workflow and Implementation



- Approach:
  - Design and train model using a deep learning library.
  - Perform inference using the FPGA.
- Implemented LeNet5 on FPGA natively (VHDL) and via OpenCL.

Firmware convolution:

- Input matrix streamed linearly into the convolution module
  - Feeds into shift register which then sends multiple row values into FIFO’s which store values on the same row, ‘iterating’ through input.
  - the FIFOs push 5 values each (25 for 5x5 filter) into a DSP which multiplies input values by filter values and sums them, producing the output for one pixel.
- Each instance of a filter convolving on an input matrix allocates 1 DSP (or 25 DSPs in the more parallel approach) for a 5x5 filter.



FPGA Resources

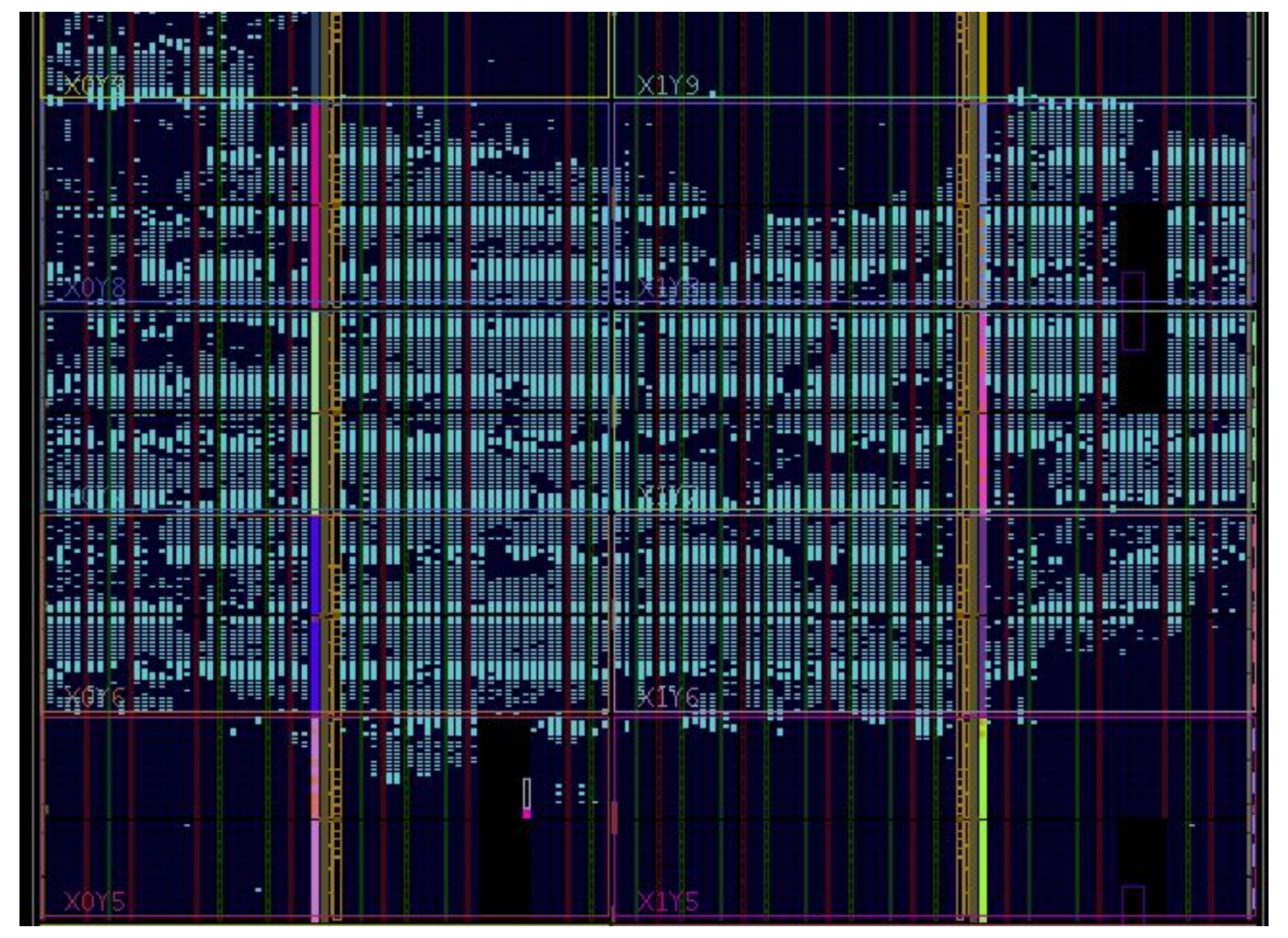


Diagram showing example resource usage on an FPGA.

- The many convolution and matrix multiplications can be resource costly for FPGAs:
  - Smaller FPGAs can quickly be resource starved.
  - Our implementation had to be shrunk in order to fit completely on an Altera Cyclone V.
- Digital Signal Processors (DSPs)
  - Used in order to perform multiplications, because generating multiplication blocks from FPGA logic elements is too expensive.
  - Cyclone V DSPs can perform 2 multiplications per clock cycle (pipelined), so latency becomes #multiplications / 2 * #DSPs.
- Estimating latency assuming DSPs are limiting factor:
  - The number of multiplications per convolutional layer is:

$$m_{if} = w_i * h_i * w_f * h_f * d_i * n_f$$

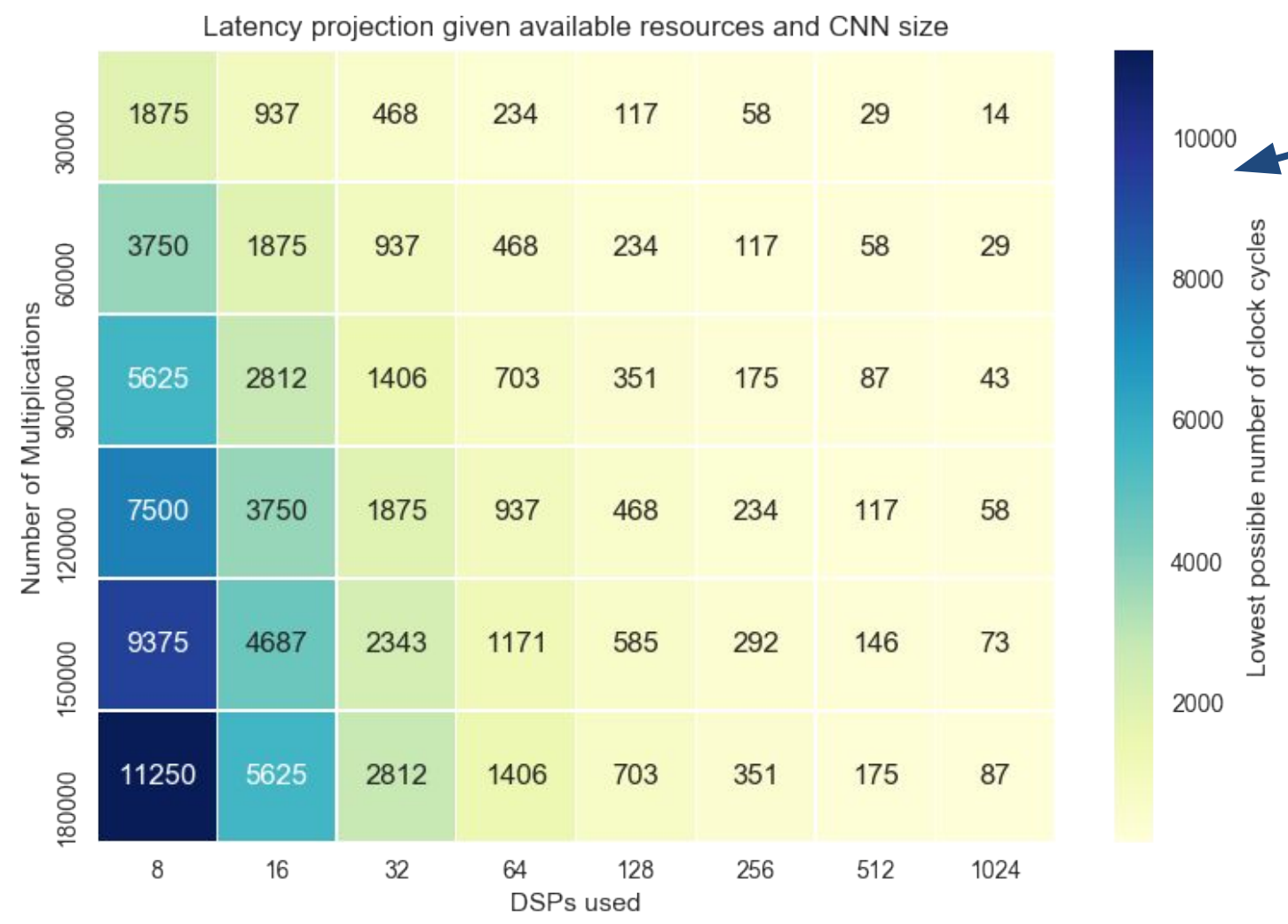
  - DSPs are assigned in chunks proportional in size to factors in equation 1, so for example a 5x5 convolution with 1 filter (n_f=1) could be assigned 25 DSPs to parallelize multiplications across filter multiplications the leaving us with w_i*h_i*d_i non-parallel operations (size of the input).

	DSPs	Memory (Block RAMs)	Clock cycles
OpenCL LeNet	35	273	1176k
VHDL 5x5 convolution*	25	4	npixels + 10
VHDL LeNet (resource conscious)*	112	300	50k
Available (Cyclone V)[1]	112	557	--
Available (Stratix 10)[2]	5,760	11,721	--

* Assuming single pixel stream in which can be widened to parallelize.
* Cyclone V available resources used as constraints.

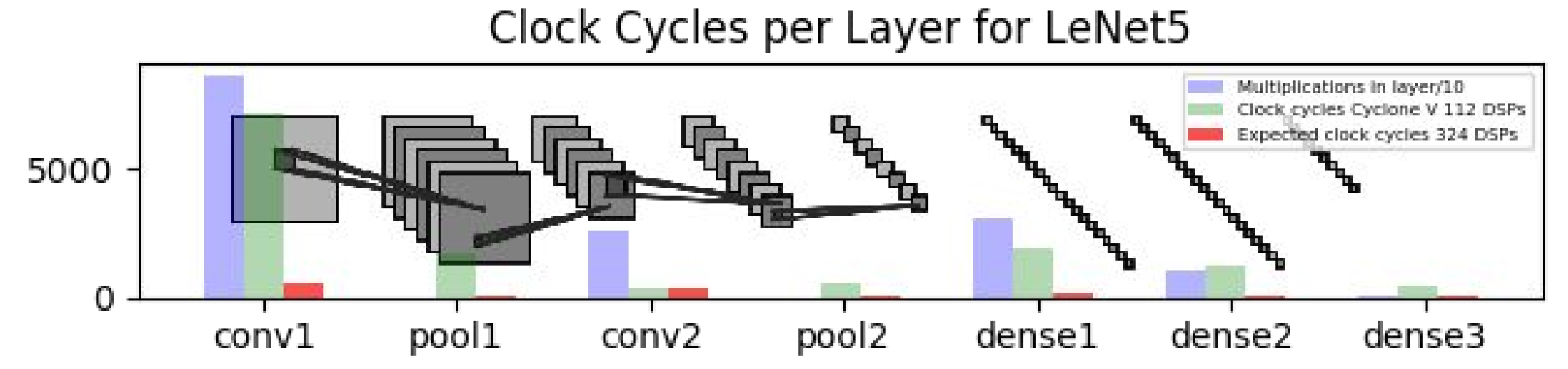
Discussion

Is our latency goal attainable?



A heatmap showing how the number of DSPs allocated to convolution layers impact the number of clock cycles for convolutions of different sizes.

- **YES.** At 400 MHz, 5 μs is 2000 clock cycles.
  - Heatmap shows that with our implementation and a large FPGA (some have 5000+ DSPs) we can predict on a reasonable network.
- We propose a pipelined CNN forward pass which scales with resources available.
  - Assuming multiplications are limiting factor, latency scales linearly with number of DSPs .



Number of cycles required to complete each layer of LeNet using resource available in Cyclone V vs an FPGA with 324 DSPs, demonstrating how much DSPs can constrain latency.

- Using LeNet as an example we see that increasing DSPs by a factor of 3 can dramatically reduce latency of a network.
- Data flow into the FPGA:
  - Use of a general purpose coprocessor for data transfer will increase latency too much.
  - Our VHDL implementation will allow for data to stream directly into FPGA input ports reducing IO caused latency.

Conclusions and future work

- FPGA DNN implementation:
  - Have a predictable real-time latency
  - Implemented in data streaming approach
  - Data can be streamed through the FPGA DNN
  - Convolutions are a good example of the FPGA potential for low latency DNNs
- Optimized DNN implementation to fit into FPGA resources:
  - large FPGAs contain 1000’s of DSPs and clocked at ~400 MHz >4,000,000 multiplications/s. For reference LeNet5 performs ~150,000 multiplications per image in its forward pass.
- Implementing new layers:
  - Most successful approach to the tracking problem has been through a combination of LSTMs and CNNs.

Contact:

TBoser@ucsc.edu

References

[1] https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/pt/cyclone-v-product-table.pdf
[2] https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/pt/stratix-10-product-table.pdf

Resources:

https://github.com/HEPTrkX/NIPS2017-demo
https://heptrkx.github.io/