Homework 4

Deadline: December 19th, 23:59; Submit to the WEB LEARNING in .zip/.rar format

Goal of this homework: Get ready for your on-board Demo (teamwork!). Get familiar with DNN accelerator architecture, instruction, and simulator.

Design a DNN accelerator with its corresponding simulator.

Constraints:

1) MAC is no less than 64;
2) End-to-end Computing Utilization should be more than 80%.
3) You are allowed to store all the weights on chip (Buffer). But the total capacity of BRAM should be less than 200KB.

Requirements:

1) Draw your designed architecture diagram with as much detail as possible (we allow the use of classical architectures such as TPU, NVDLA, Eyeriss, and so on).
2) Illustrate your designed/selected dataflow in a for-loop manner.
3) List the instructions and introduce the corresponding operations being executed.
4) Draw the architecture block diagram of your **fine-grained event-driven simulator**.
5) Display the instruction diagram for a 7-layer network simulation.
6) Try changing the hardware configurations and analyze the impact on performance

Workload:

Layer	**Type**	**IFM Size**	**Input Channel**	**OFM Size**	**Output Channel**	**Kernel Size**	**Stride**	**Padding (zero)**	**ReLU**	**Q_{IN}**	**Q_{OUT}**	**Q_{W}**
Layer-1	Conv	32^2	1	32^2	32	5×5	1	2	√	7	5	7
Layer-2	Max Pooling	32^2	32	16^2	32	2×2	2	0	×	5	5	/
Layer-3	Conv	16^2	32	16^2	64	3×3	1	1	√	5	5	8
Layer-4	Conv	16^2	64	8^2	64	3×3	2	1	√	5	5	8
Layer-5	Conv	8^2	64	4^2	128	3×3	2	1	√	5	5	8
Layer-6	Average Pooling	4^2	128	1^2	128	4×4	1	0	×	5	5	/
Layer-7	FC	1	128	1	10	1×1	/	/	×	5	5	6

Tips:

1) It's best to accompany the diagrams with concise textual explanations. Avoid lengthy discussions—just make sure everything is clearly presented. There's no need to add unnecessary

words in an attempt to compete excessively.

2) The requirements we've provided are consistent with the final project, so I strongly recommend using this simulator for the final project to reduce some workload. However, it's not mandatory to use the exact same one—you can identify issues with the simulator during the final project and make improvements or iterations. This is something we encourage.
3) The items you need to submit are: **Code**: Submit the simulator code files and an executable script. The script should at least support configuring hardware parameters, output simulation results, and generate the instruction diagram. **Document**: Include your design, introduction, and result analysis as specified in the requirements.

Grading Criteria: Each of Requirements 1-4 can earn up to 3 points for clear and reasonable illustrations. Requirement 5 awards 5 points for accurately drawing the instruction dependency graph and providing analysis. Analyzing/tuning the architecture and scheduling with the simulator can also earn up to 5 points (for addressing 1-2 aspects). An additional 3 points will be given based on the ease of executing the code—ideally, a single-step script that generates the report and instruction dependency graph (we won't be overly strict).

The score calculation above represents the **base score**.

We encourage designing larger-scale accelerators and improving utilization through co-design of hardware and software. Therefore, your final score will be calculated as:

$$\text{score} = \max\left(25,\ \text{base score} \times \frac{\text{actual utilization}}{80\%} \times \sqrt{\frac{\text{MAC number}}{64}}\right)$$