Puspal Chatterjee
ECEN Grid username: 695r43
Abhishek Nayak
ECEN Grid username: 695r48

SoC Design Course Intermediate Project report: Neural Network Inference

18th November 2021

STATUS

We have run the software profiling on CNN and MLP programs.

For quick reference - the below data trace the functions in the CNN program taking up the majority of the time during the execution of the software program -

Index	%time	self	children	called	name
1	100.0	0.00	0.17		main[1]
2	98.1	0.00	0.17	100	kann_apply1[2]
3	97.1	0.00	0.17	100	kad_eval_at[3]
4	97.1	0.00	0.17	100	kad_eval_marked[4]
5	**94.1**	**0.15**	**0.01**	**206**	**kad_op_conv2d[5]**

Pointer to the complete profile data -
https://drive.google.com/file/d/10qj530iJNkCfdxCnwPSOxis4lS7FzUEn/view?usp=sharing

We observe that the function **kad_op_conv2d** accounts for 94.1% of the total execution time.

Similarly, from the profiling data of the MLP program, we found that the function **kad_sgemm_simple** accounts for 98.1% of the total execution time.

Pointer to the complete profile data -
https://drive.google.com/file/d/1nsVkw4I6WYh8h-UTzBaKvXTkbhzcpkoT/view?usp=sharing

Both of the above functions perform a huge amount of floating-point additions and multiplications using the loop functions of **kad_sdot** and **kad_saxpy_inlined**.

DESIGN SPACE EXPLORATION

- Hardware optimizations -
    - Floating-point adder custom instruction module: to add the float type variables.
    - Floating-point multiplier custom instruction module: to multiply float type variables.
    - Both custom instruction modules will have two 32-bit input values.
- On-chip communication architecture -
    - As per our current planning, we will use a DMA controller for on-chip communication between the processor and the hardware accelerator.
- Memory system -
    - We plan to use Block RAM to store the weight, bias, input, and output parameters of hidden layers.
- Software optimizations -
    - We plan to use loop unrolling in many of the loops present in C program files to optimize the program's execution speed.

UPDATED TIMELINE

Milestone	Timeline	Status
Environment setup	10/28/21	Done
Software profiling of MLP and CNN programs	11/12/21	Done
Implementation of acceleration of MNIST-CNN and MNIST MLP (Including RTL Design → integration of communication architecture → possible pipelining of MAC operations → software optimizations)	12/3/21	In progress
Verification and Testing	12/10/21	TBD
Results and Analysis of data	12/10/21	TBD
Project report completion	12/17/21	TBD