**2020.2.12**

**1. Redesign the post process logic**

In the original design, the DLA computes the post process, like bias, batch normalization, element-wise addition and multiplication, after the convolution or FC produced. The PE array (for convolution and FC) and post-process elements are pipelined for real-time ASR. However, this design requires sufficient sources.

In wavenet, the number of multiplication is detailed in Eq.1

 (1)

Where Wout is the length of output time sequence. K is the Kernel Size. Cin and Cout are the numbers of input and output channel, respectively.

However, the number of multiplication in BN or Element-wise operation or Activation Function, such as sigmoid and tanh, is calculated in Eq.2 as

 (2)

Hence, the computation in post process only accounts for a relatively modest portion of the whole layer.

Similar result can be obtained in FC/LSTM.

Therefore, I make a compromise between the one-time pipeline logic and one-after-another operation flow in reference [1]. The bias or BN can be computed at once when the convolution or FC is finished. However, the features should be accessed again from the Feature Buffer to do the element-wise operation. In this way, 32 x 2 floating-point multipliers and adders are saved and the post-processed logic are fully loaded when convolution or FC is under calculation.

Reference:

[1] Laika: A 5uW programmable LSTM accelerator for always-on keyword spotting in 65nm CMOS.

**2. Transmit the DLA SoC slave interface to AXI bus and the DLA DDR master interface to PHI bus for DDR.**

This work is undergoing. As my current laptop can not P&R the relative Xilinx IPs because of the meager CPU and DDR capacity, I am reviewing the previous work in Tanji3. I will get a more power computer in Friday. Hope it can run the Vivado P&R.

**3. If anyone can enter the lab, please reset my computer to UNIX and start the teamvierwe.**

**2020.2.19**

**1. SoC/CPU Core to DLA interface**

In the current SoC design, the CPU can only access the DDR through the SoC-DLA interface and DLA-DDR interface. In another word, the DDR is an exclusive slave of the DLA.

In order to achieve the abovementioned function, the RTL for SoC<->DLA on-chip memory <->DDR is added in DLA RTL design.

**2. Fix the clock-gating bug in DLA computation logic.**

****

In the above diagram, if only multiplication output is selected (out1 enable), the changing data of Mul\_Out\_R would introduce dynamical power in the adder. Another register should buffer the addition operand.

**2020.2.24**

**1. Fix the PE\_EN (MUL\_EN, ADD\_EN) control signal of the PE array when processing the FC with Column-Combine sparsity.**

****

When processing the convolution without sparsity awareness, the MUL\_EN and ADD\_EN in the entire PE array are synchronous (set high or low at the same time). Because the operation in each PE is regular. However when processing the FC with sparsity, the MAC1s in different PEs will receive asynchronous control signals because the CRS coding for Column-Combine conflict requires asymmetric MAC1 activation.

**2. Simplify the CRS decoding logic.**



The CRS coding for Column-Combine conflicted weights consists of three parts, PE IDX, Feature IDX and Weight. The PE IDX indicates which PE in 32-PE row receives the conflicted weight. The Feature IDX points out the paired feature of the compressed weight.

In the 256-bit weight interface, one weight access can fetch 16 conflicted weights at one time. These 16 weights are from 1 weight row to 16 rows at most. In order to distinguish the weights to each uniform row, lots of shifters, n to 2^n decoders are required.

In the original design, the PE IDX is decoded to 32-bit one-hot MAC1 enable signals. And lots of 5-32 decoders are required as mentioned above. In the improved design, the 16 encoded PE IDX vector is shifted firstly according to the number of conflicted weights in each row. Then the divided PE IDX is restored to the one-hot PE enable signals.

**2020.3.4**

1. Modify the DLA2DDR interface as 64 bits and inout type.

2.The teamviewer in the LAB computer went offline. If someone can enter the LAB, please call me.