SEOUL NATIONAL UNIVERSITY
SYSTEM SEMICONDUCTOR ENGINEERING FOR AI
서울대학교 연합전공 인공지능반도체공학

2025 Deep Learning Hardware 설계 경진대회

주동준, 우상욱, 남현준

전자전시회, 숭실대학교

Contents

1. Outline

- 개요
  - 딥러닝 추론 하드웨어 설계
- 설계 목표
  - On Device에서의 한정적인 자원을 최대한 활용하여 YOLOv3-tiny 모델을 FPGA 보드상에서 효율적으로 사용하고자 한다.

1. Outline

- **구현 결과**
  - Quantization (mAP)
  - H/W inference results (@XXXMHz)
  - Board implementation
    - Power
    - LUT
    - BRAM
    - DSP

1-1. Quantization (mAP)

- mAP: 78.03%

- (80.31%를 얻을 수 있었지만, hardware compatibility를 위해 mAP loss를 감안하더라도 multiplier를 모두 power of 2로 변경)

```
 for thresh = 0.24, precision = 0.63, recall = 0.54, F1-score = 0.58
 for thresh = 0.24, TP = 1604, FP = 953, FN = 1355, average IoU = 45.75 %

 mean average precision (mAP) = 0.780314, or 78.03 %
Total Detection Time: 37.000000 Seconds
```

```
 for thresh = 0.24, precision = 0.66, recall = 0.59, F1-score = 0.63
 for thresh = 0.24, TP = 1750, FP = 887, FN = 1209, average IoU = 48.51 %

 mean average precision (mAP) = 0.803073, or 80.31 %
Total Detection Time: 33.000000 Seconds
```

Figure. 1. mAP Results

1-2. Board Implementation - Power

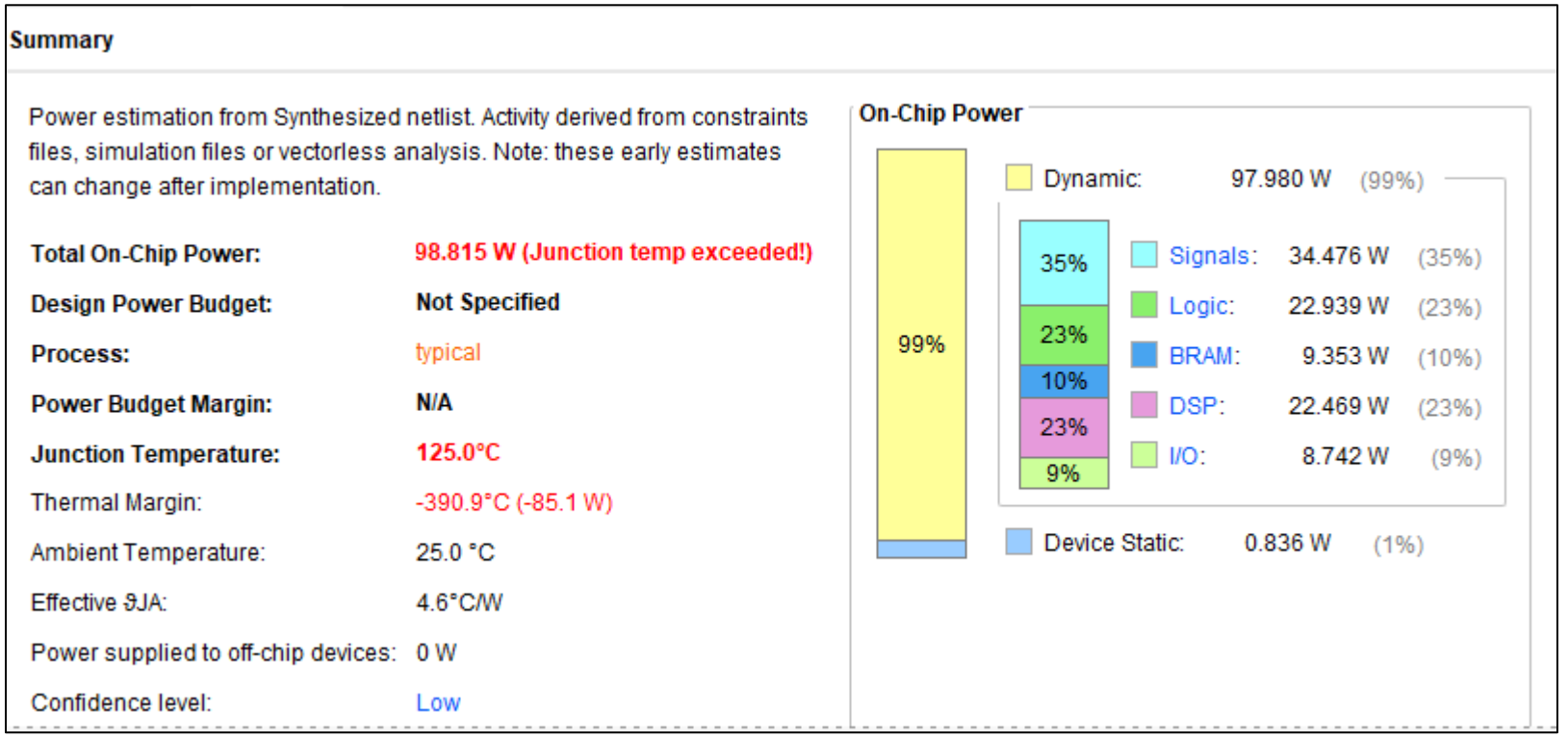


Figure. 2. Power Reports

1-2. Board Implementation - LUT

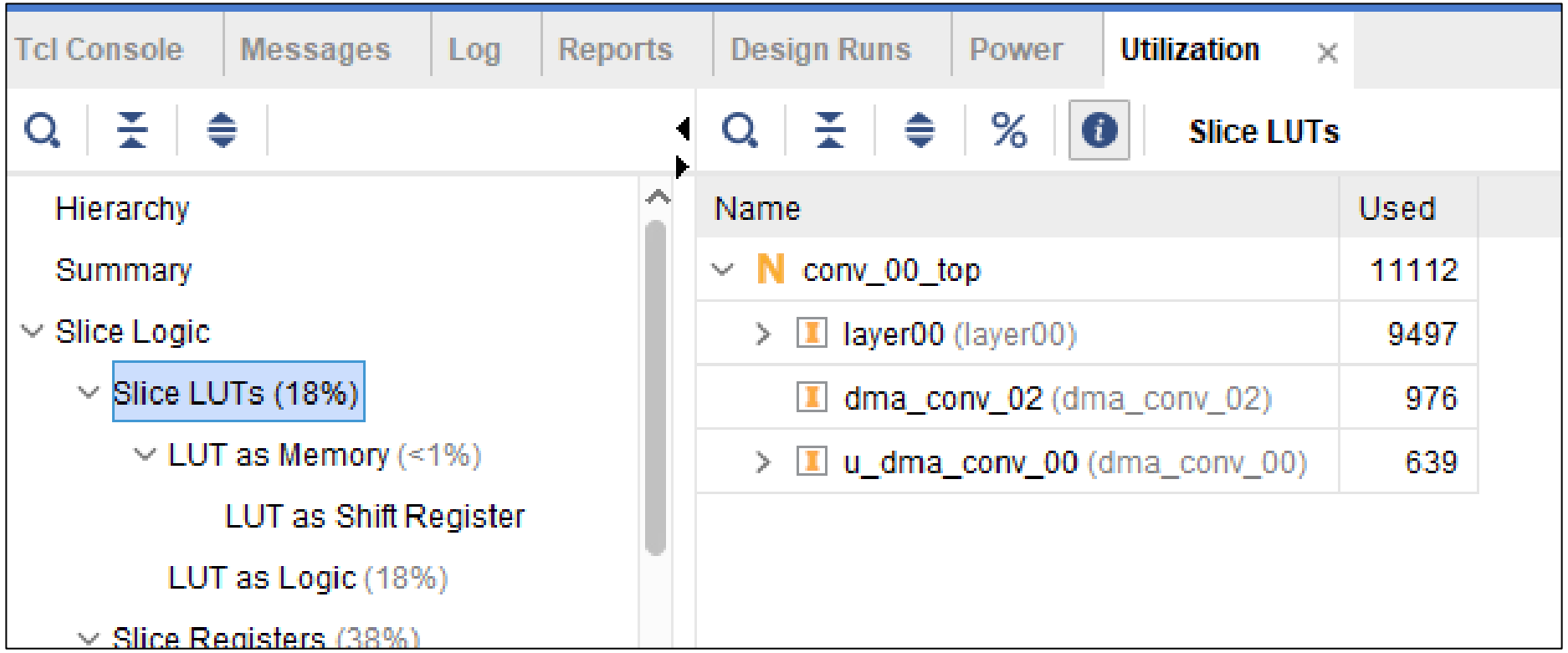


Figure. 3. LUT Reports

1-2. Board Implementation - BRAM

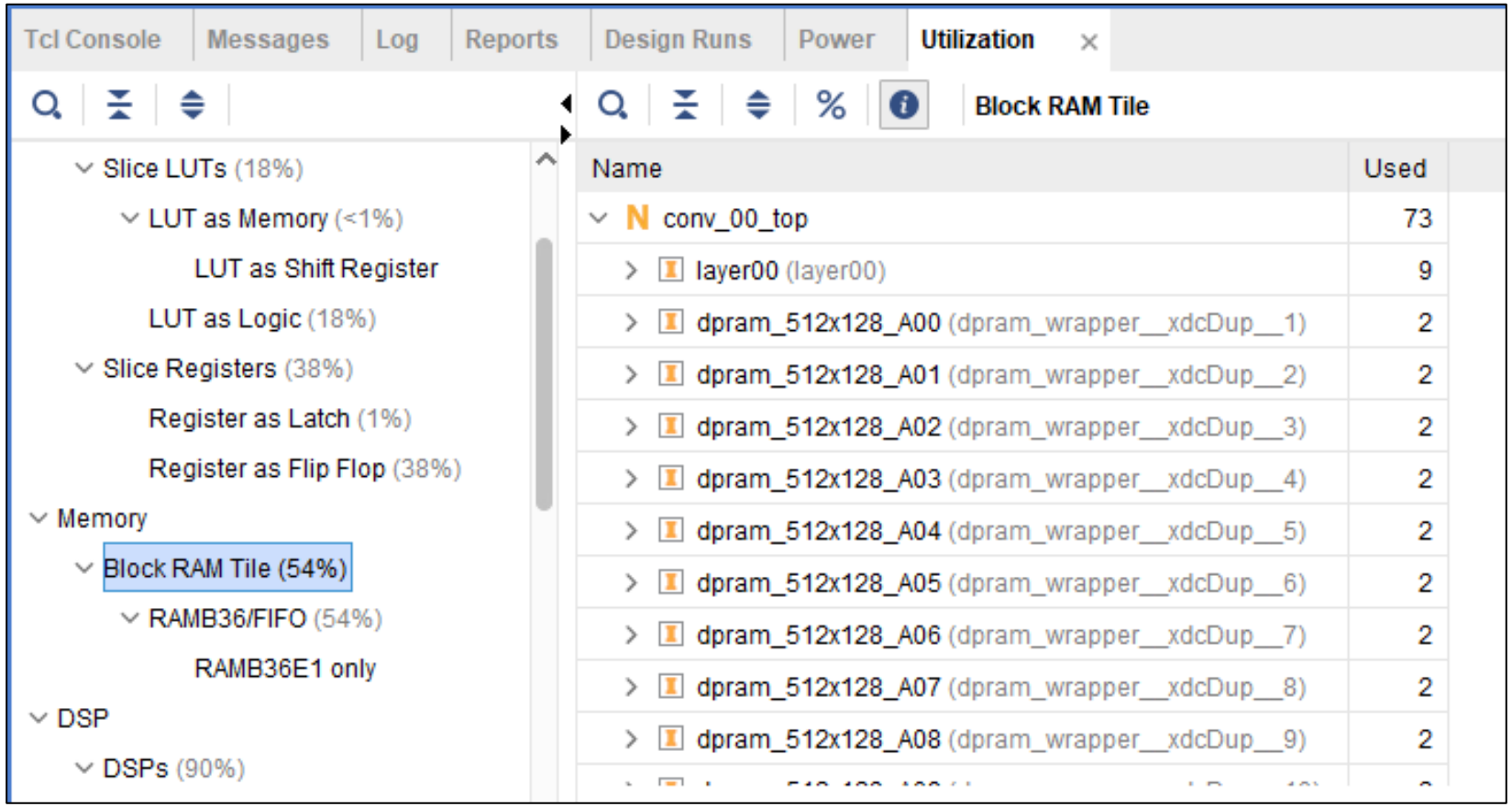


Figure. 4. BRAM Reports

1-2. Board Implementation - DSP

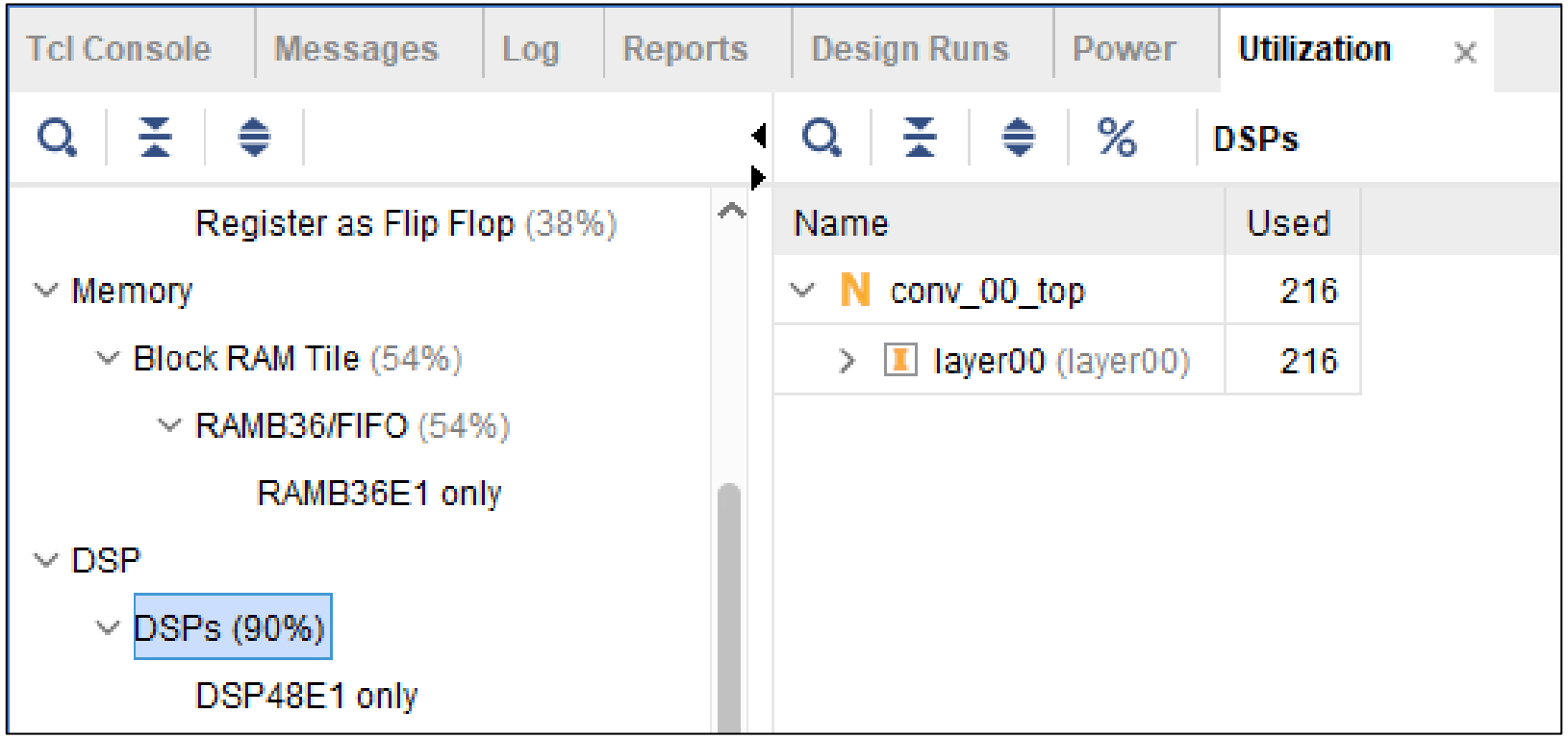


Figure. 5. DSP Reports

2. Idea & Discrimination

- Concept 1:

  Layer 0, 2, 4에서는 IFM, OFM의 크기보다 Weight 크기가 현저히 작으므로 ROM으로 미리 저장하여 사용한다.

  Layer 06부터는 Weight의 크기가 Feature Map보다 크기 때문에 DMA를 통해 weight 값을 가져온다.

2. Idea & Discrimination

- Concept 2:

  하나의 DSP로 2개의 8-bits*8-bits 연산을 수행하여 DSP slice를 절약한다.
  (a+b)*c => a*b , b*c 로 나눠 값을 사용한다.

3-1. Design Detail - Quantization Method

Layer	Number of Multipliers of Weight
00	16
02	256
04	128
06	256
08	128
10	1024
12	256
13	256
14	256
17	128
20	256

Layer	Number of Multipliers of Input
00	128
02	8
04	16
06	8
08	16
10	2
12	8
13	8
14	8
17	16
20	8

Table. 1. Number of Multipliers on each layers

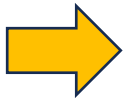
3-1. Design Detail - Quantization Method

- Bias multiplier 값이 2048이 되도록 weight_multiplier와 input_multiplier의 값을 임의로 설정한다. hardware에서 bit shift연산으로 software와의 값 비교시 정확도를 위해서 2^11 (2048)로 설정하였다.

- 각 channel의 min, max의 절대값을 구하여 8bit의 최대값인 127로 나누고 둘 중 큰 값을 multiplier로 사용한다.

3-1. Design Detail - Quantization Method

Multipler	Input	Weight	Bias
CONV0:	48.9323	20.9064	1023
CONV2:	7.86606	260.232	2047
CONV4:	10.6724	191.804	2047
CONV6:	9.81697	208.517	2047
CONV8:	14.7207	139.056	2047
CONV10:	2.82652	724.211	2047
CONV12:	5.94372	344.397	2047
CONV13:	7.3469	278.621	2047
CONV14:	6.74057	303.684	2047
CONV17:	16.8386	121.566	2047
CONV20:	8.36135	244.817	2047



Multipler	Input	Weight	Bias
CONV0:	128	16	2048
CONV2:	8	256	2048
CONV4:	16	128	2048
CONV6:	8	256	2048
CONV8:	16	128	2048
CONV10:	2	1024	2048
CONV12:	8	256	2048
CONV13:	8	256	2048
CONV14:	8	256	2048
CONV17:	16	128	2048
CONV20:	8	256	2048

Figure. 7. Number of multipliers

3-1. Design Detail - Quantization Method

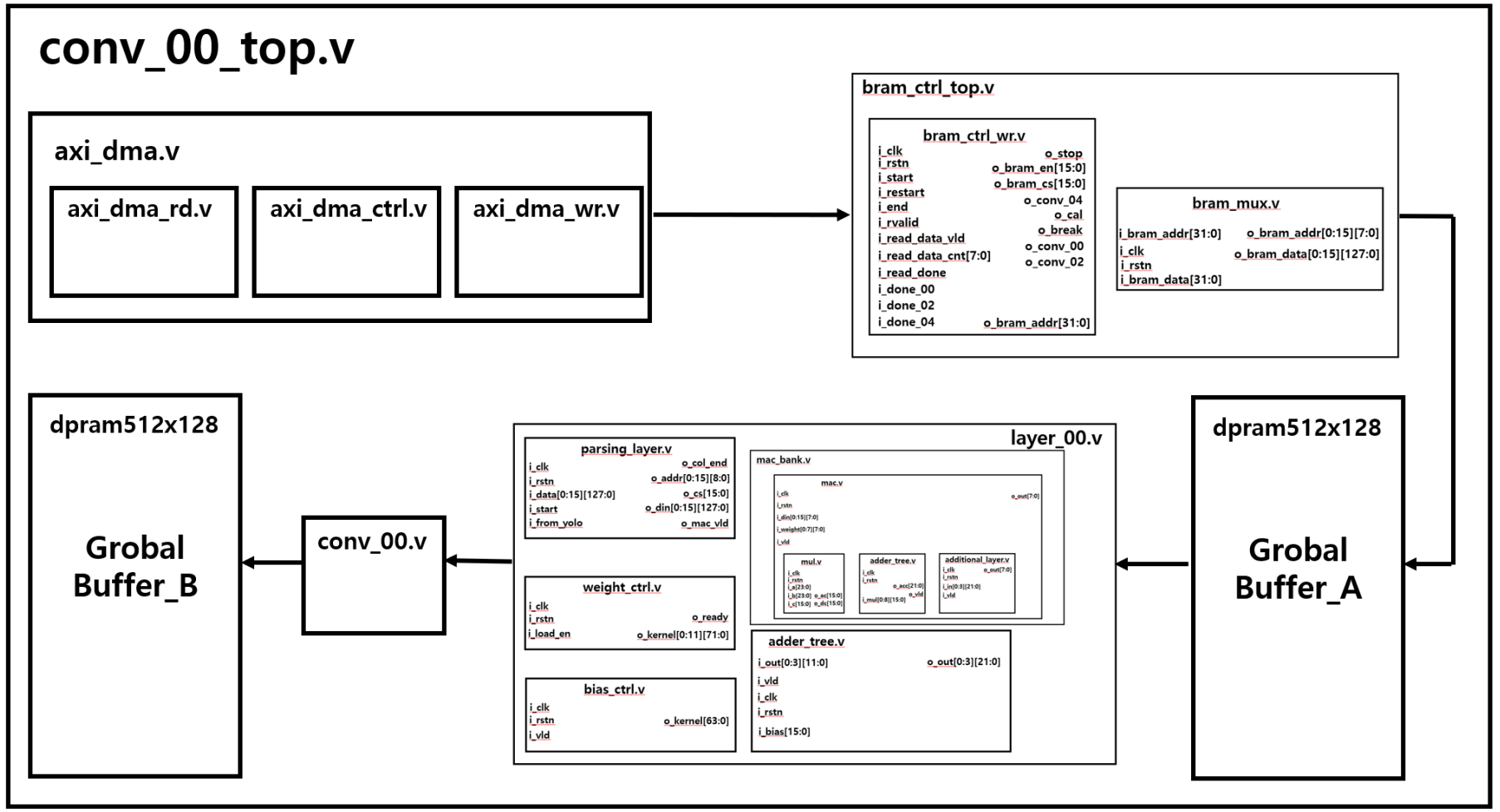


Figure. 8-1. Block Diagram

3-1. Design Detail - Quantization Method

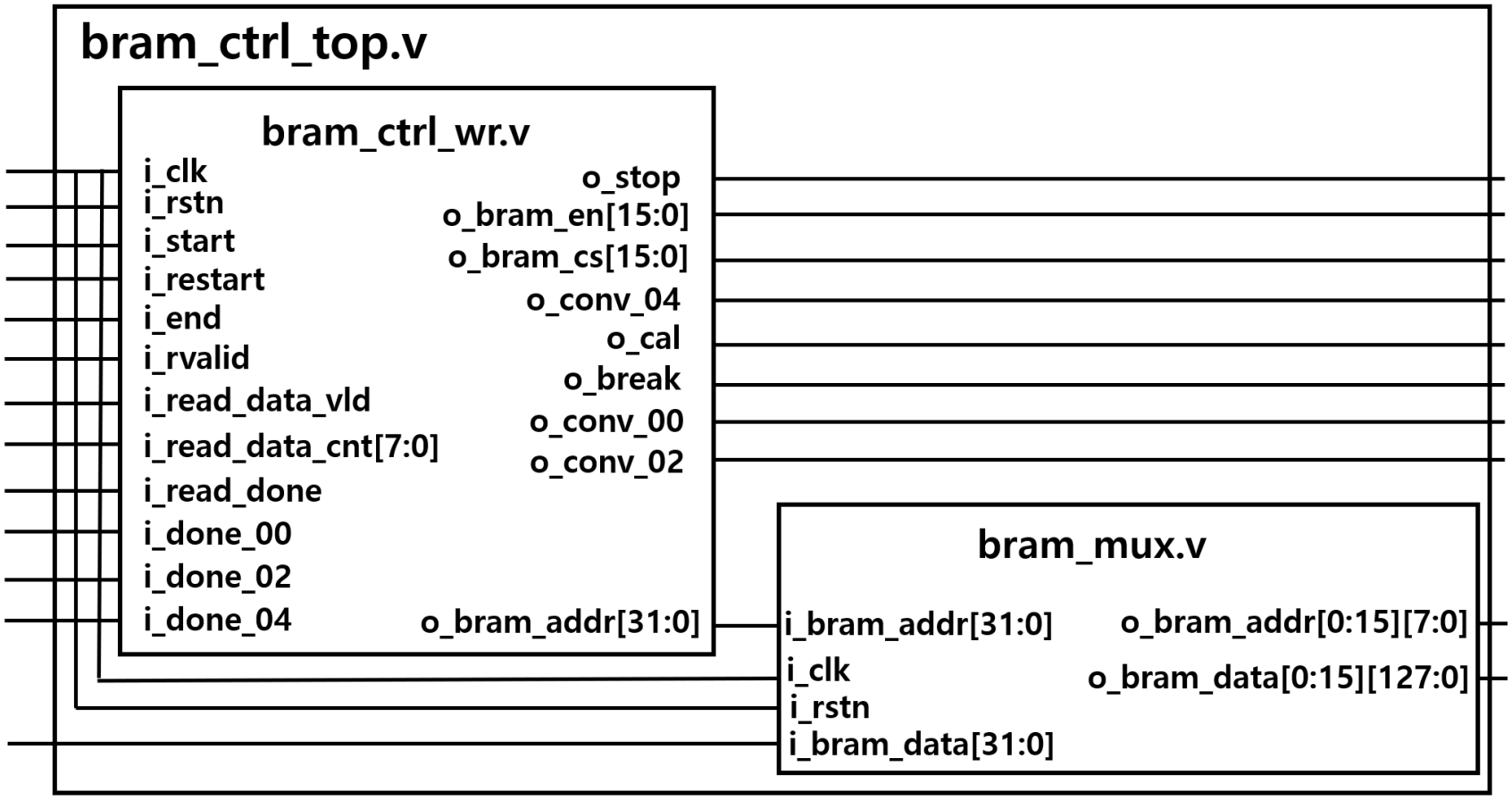


Figure. 8-2. Block Diagram

3-1. Design Detail - Quantization Method

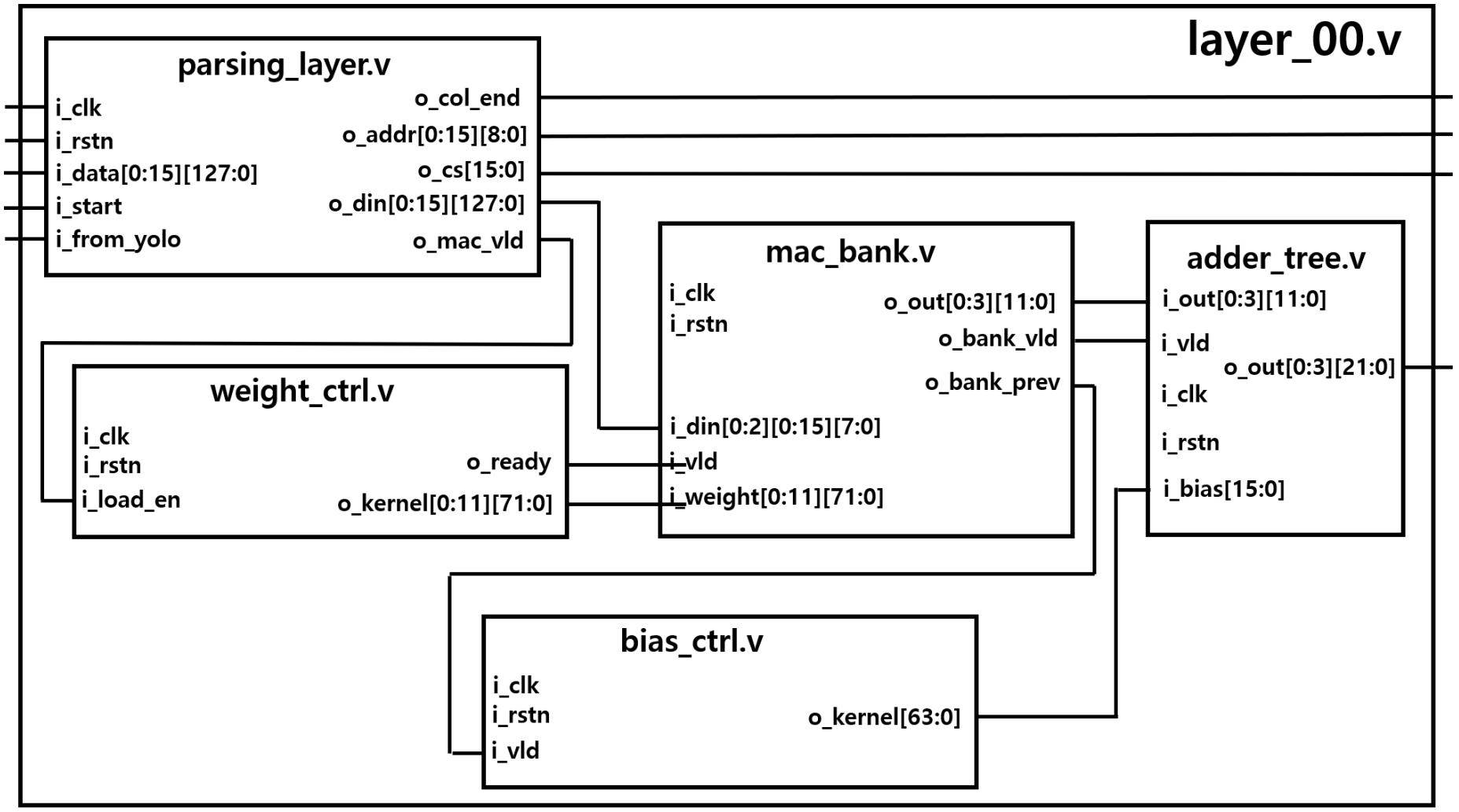


Figure. 8-3. Block Diagram

3-1. Design Detail - Quantization Method

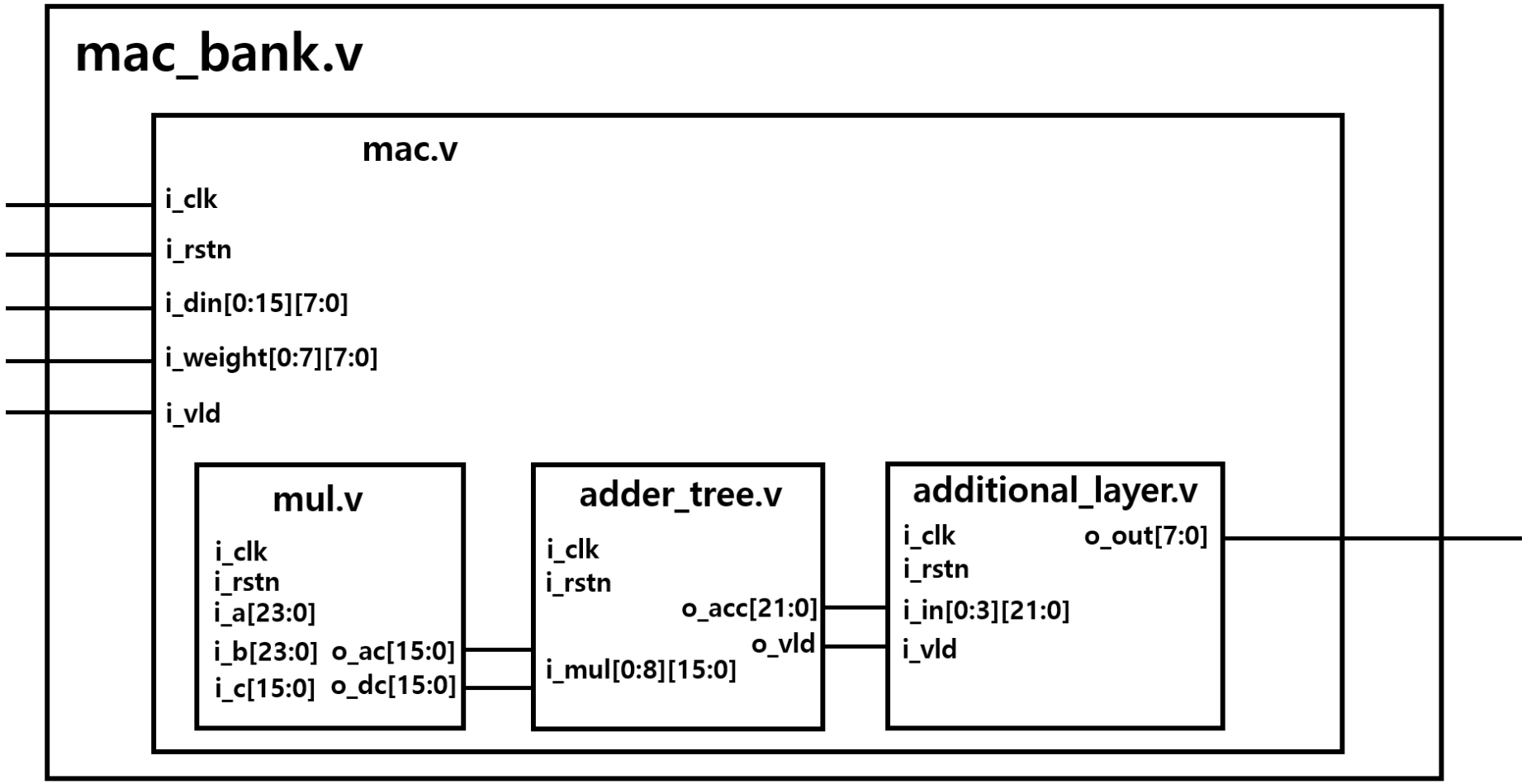


Figure. 8-4. Block Diagram

3-1. Design Detail - Quantization Method

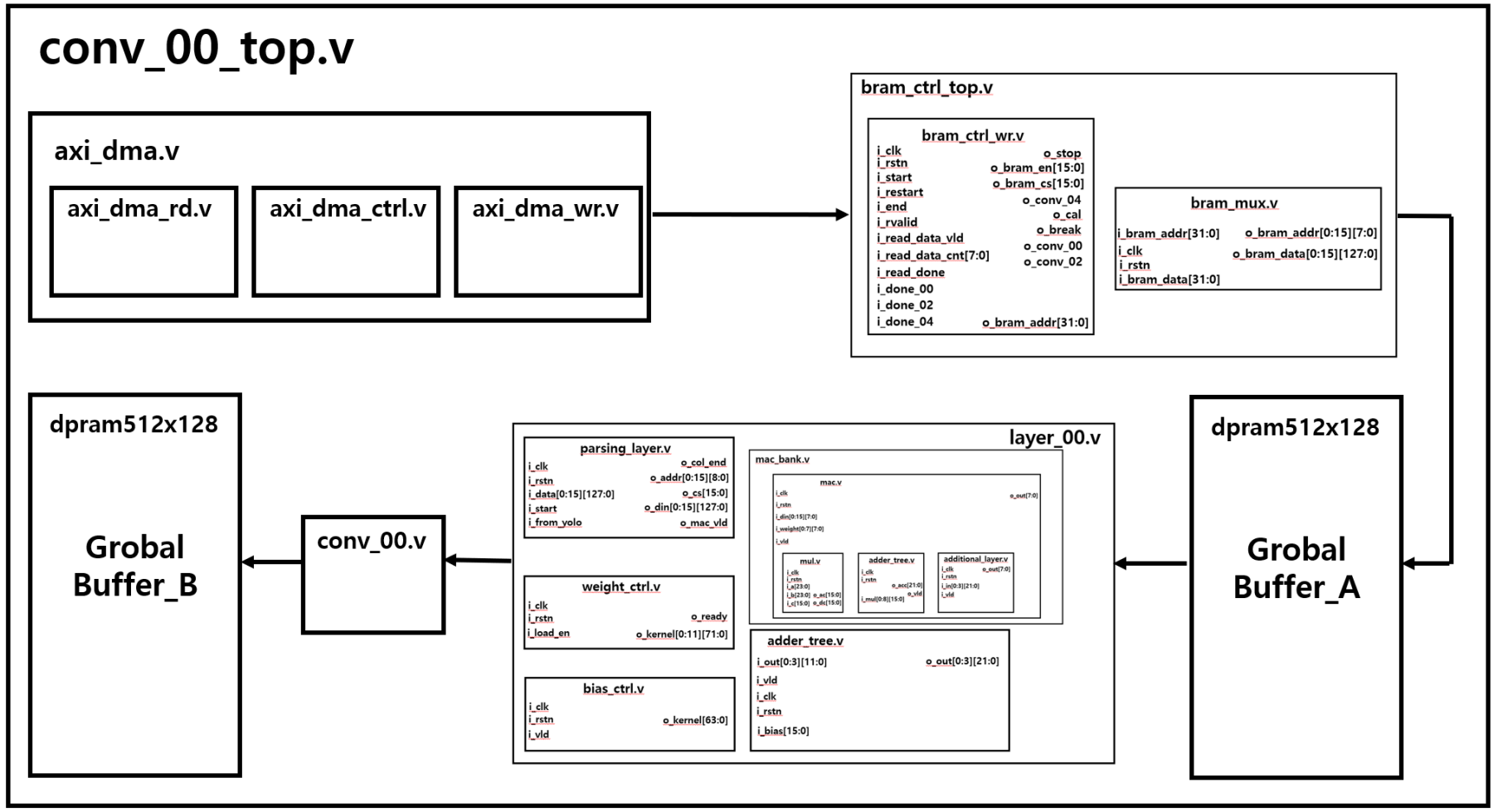
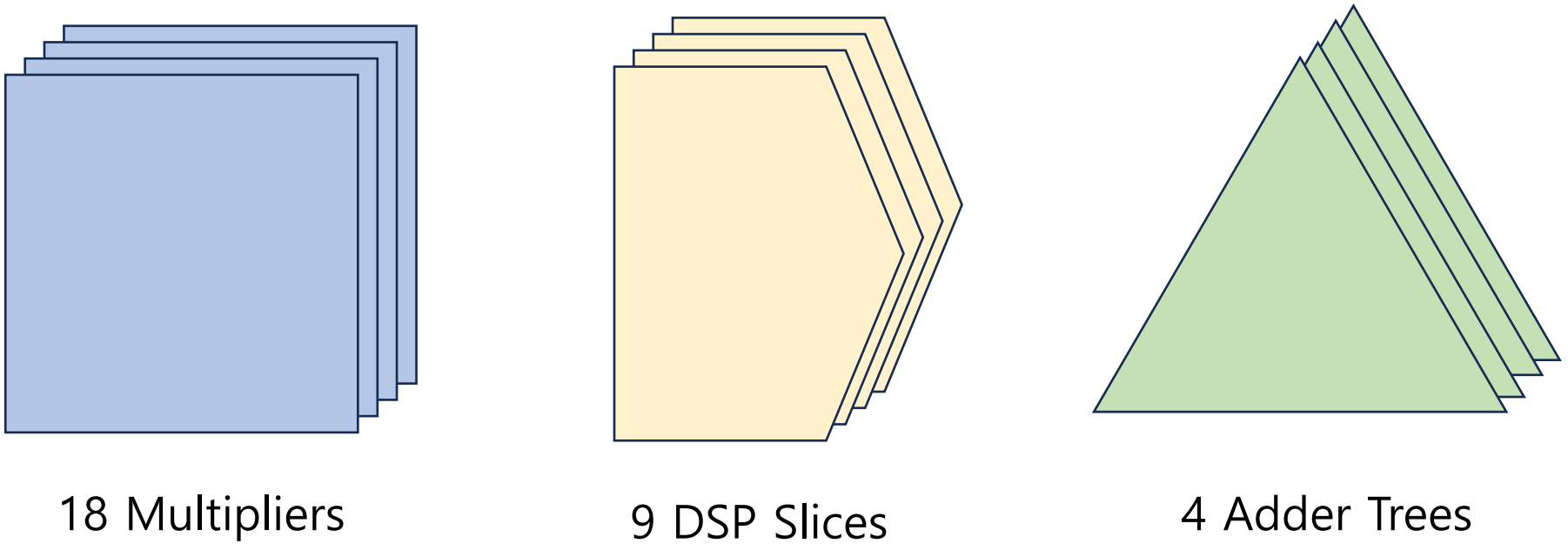


Figure. 8-5. Block Diagram

3-3. Design Detail - MAC & DSP

- CONV00 layer:

18 Multipliers → 9 DSP Slices
4 Adder trees



18 Multipliers　　9 DSP Slices　　4 Adder Trees

Figure. 8. CONV00 Configurations

3-3. MAC & DSP

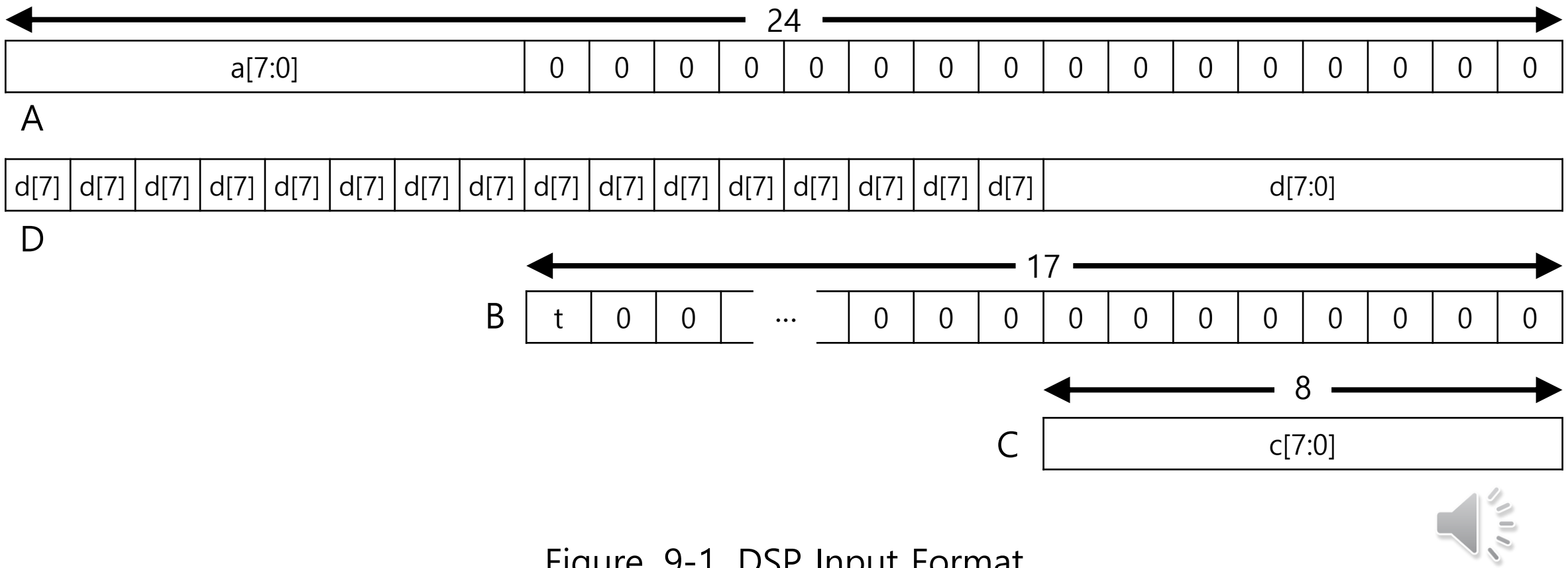


Figure. 9-1. DSP Input Format

3-3. MAC & DSP

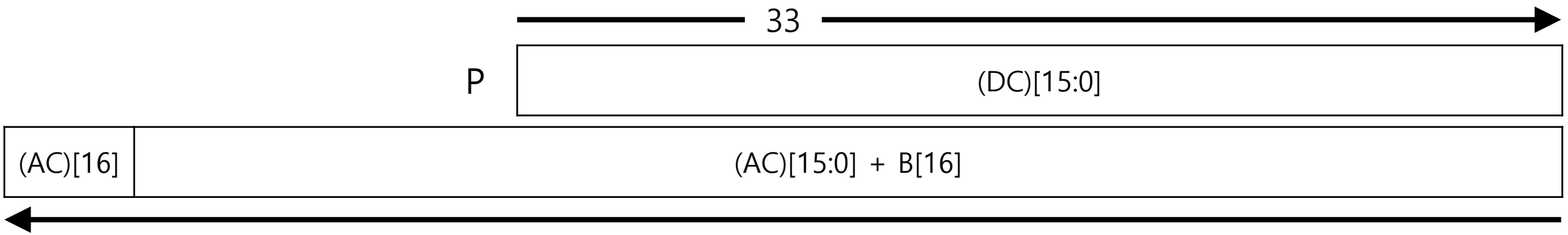


Figure. 9-2. DSP Output Format

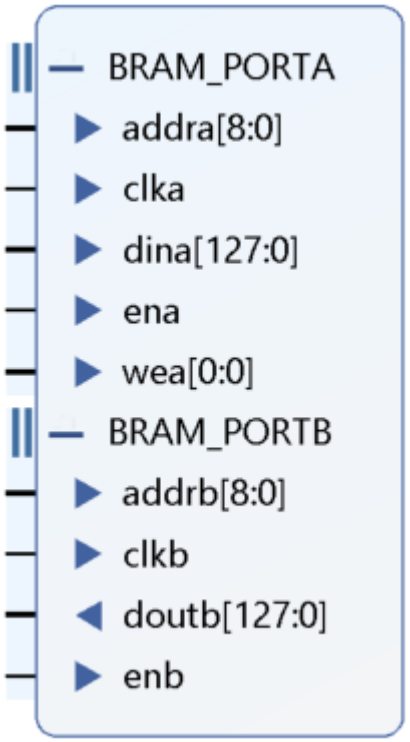
3-3. MAC & DSP

Courtesy of Xilinx

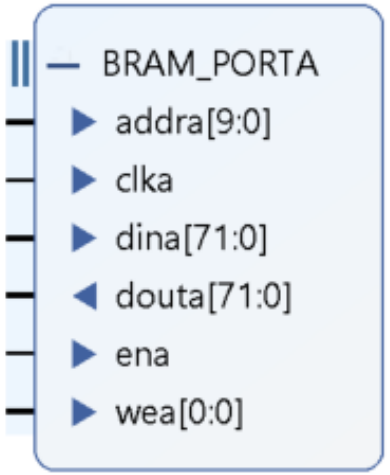
i	a_i	d_i	b_i	$a_i b_i$	$d_i b_i$	$P_i = \sum_{j=0}^{i} (a_j 2^G + d_j) b_j$	$P_i[35:18]$	$\sum_{j=0}^{i} a_j b_j$	$P_i[17:0]$	$\sum_{j=0}^{i} d_j b_j$
0	1	-4	-2	-2	8	-524280	-2	-2	8	8
1	2	8	-3	-6	-24	-2097168	-9	-8	-16	-16
2	3	17	2	6	34	-524270	-2	-2	18	18
3	4	-19	1	4	-19	524287	1	2	-1	-1
4	5	-1	2	10	-2	3145725	11	12	-3	-3
5	6	4	1	6	4	4718593	18	18	1	1
6	7	-2	1	7	-2	65553599	24	25	-1	-1

Table. 1. Example Dot Products

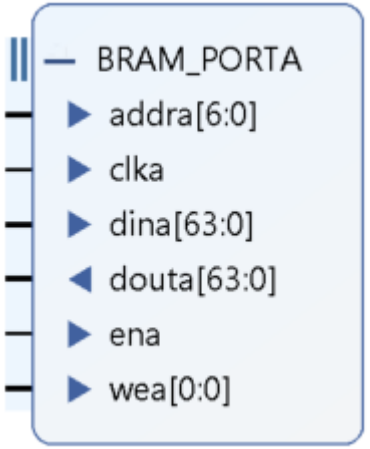
3-4. BRAM Configurations



512x128
36K BRAMs: 2



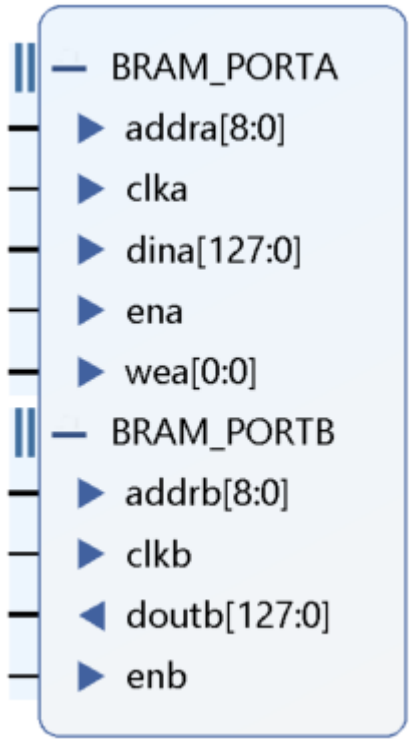
1024x72
36K BRAMs: 2



128x64
36K BRAMs: 1

Figure. 10. BRAM Configurations

3-4. BRAM Configurations

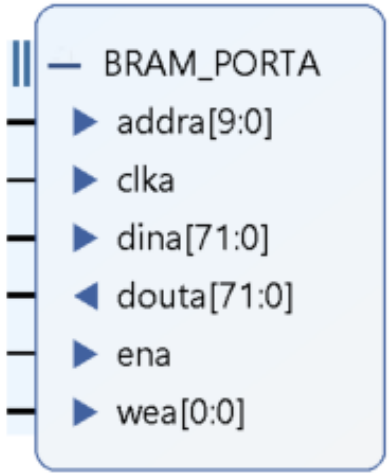


512x128
36K BRAMs: 2
x2

- DPRAM

Global buffer A and B

3-4. BRAM Configurations

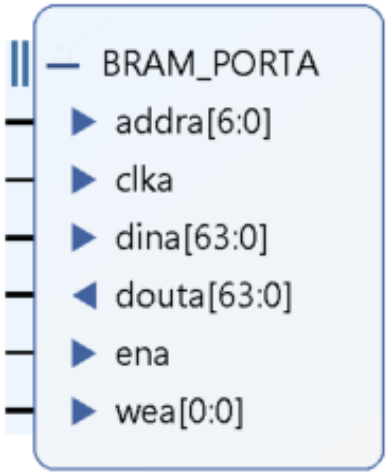


1024x72
36K BRAMs: 2
x4

- SPROM

For saving weight values of CONV00, 02, 04 layer

3-4. BRAM Configurations



128x64
36K BRAMs: 1
x1

- SPROM

For saving all of the bias values

4. Others