Lab in System on Chip Integrated Design

Final Project Report

4108064101 杜冠廷

1. Topic: KT Gray Image Compression Encoder
2. Abstract:

KT Gray Image Compression Architecture combines the specifications of both JPEG and JPEG2000 image compression. This method not only avoids some drawbacks of JPEG, such as mosaic distortion, but also retains several advantages of JPEG2000, such as real-time decoding. Compared to JPEG2000, it boasts lower computational complexity, and is also much more suitable for hardware implementation.

In this project, I utilized Zedboard for the hardware implementation of the encoder and then developed the decoder using MATLAB.

1. Architecture:
   1. System:
      1. Hardware design, which includes DWT, quantization, difference calculation, and Huffman coding.
      2. Communication protocol, which involves AXI4-Stream and AXI4-Lite.
      3. Software design, which includes reading/writing data from an SD card.

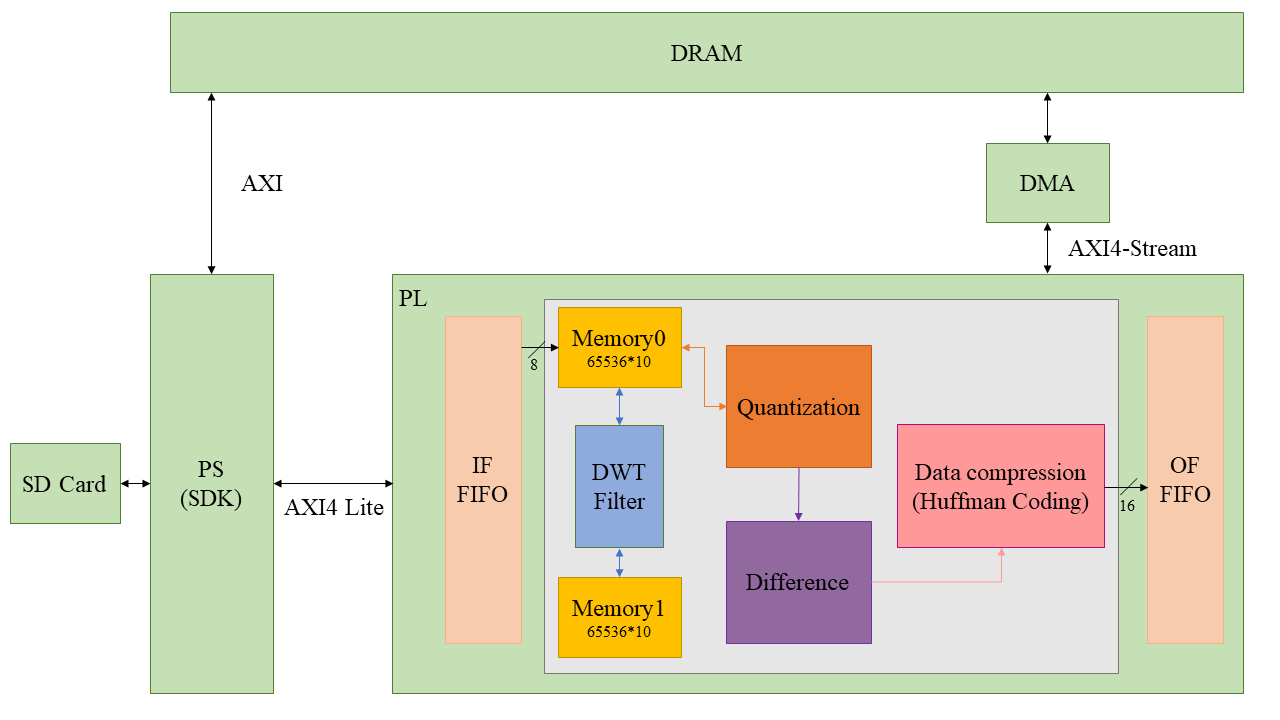


Figure 1 System Architecture

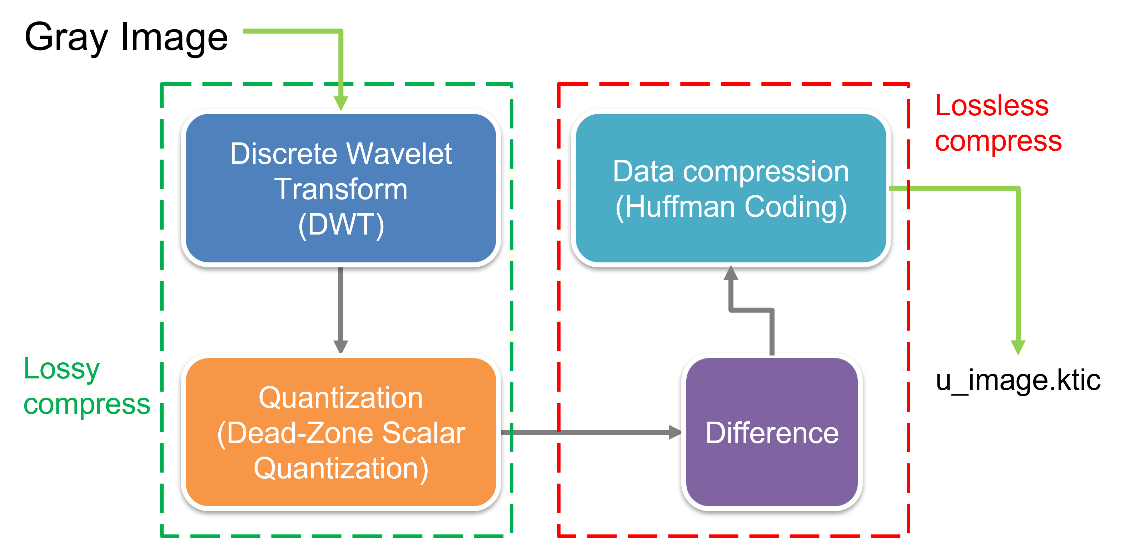


Figure 2 Encoder Architecture

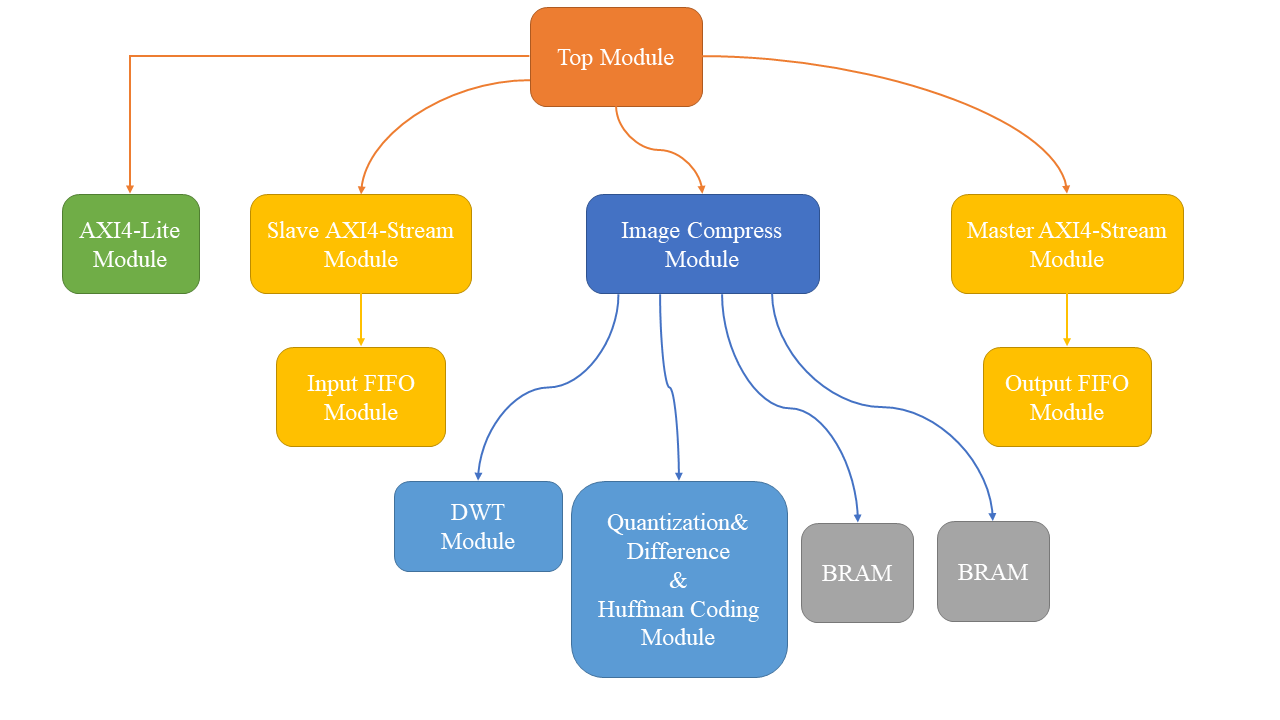


Figure 3 Design Module

* 1. Hardware Design:
     1. Layer 1: DWT layer(Discrete Time Wavelet Transform)
     2. Layer 2: Quantization layer(Dead-Zone Scalar Quantization)

In this layer, we will quantize the data by discarding some bits to achieve data compression.

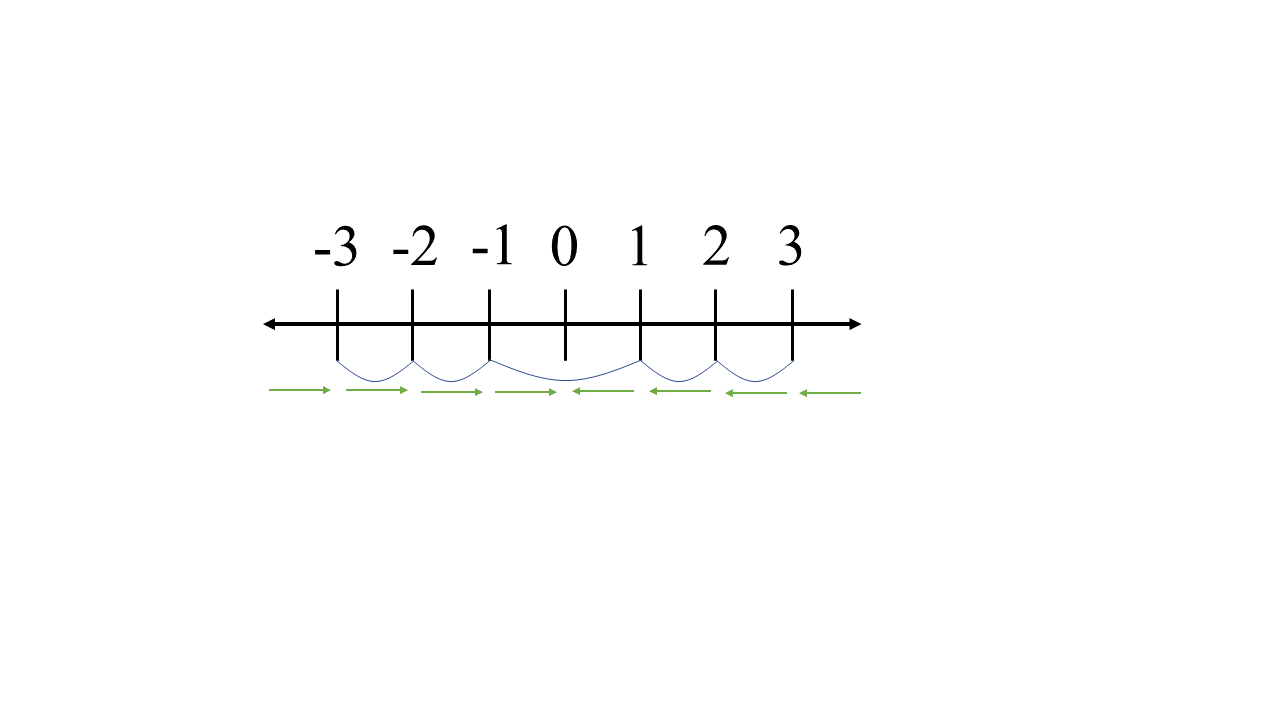


Figure 4 Dead-Zone Scalar Quantization

* + 1. Layer 3: Difference layer

In this layer, we first divide the data processed in the previous steps into four regions: LL, HH, LH, and HL. Then, for each region, we subtract each column of data from its left column and record the difference. This step aims to concentrate the frequency of data occurrence.

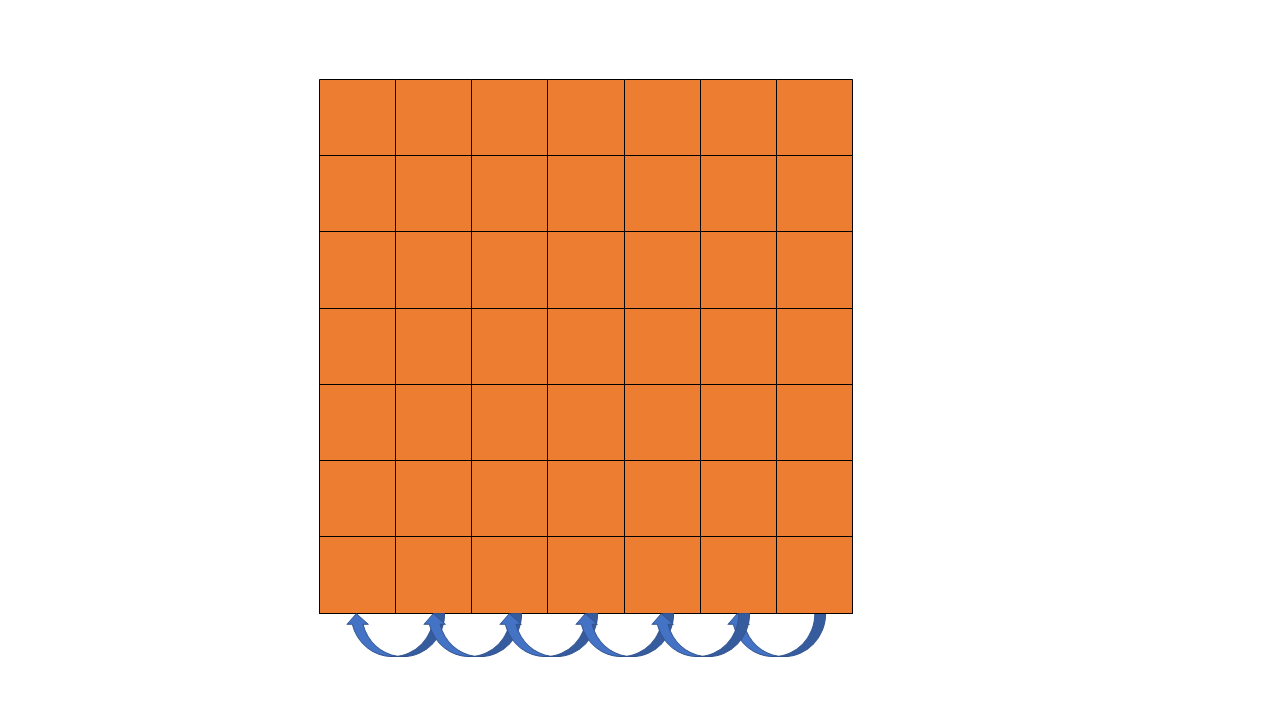


Figure 5 Difference

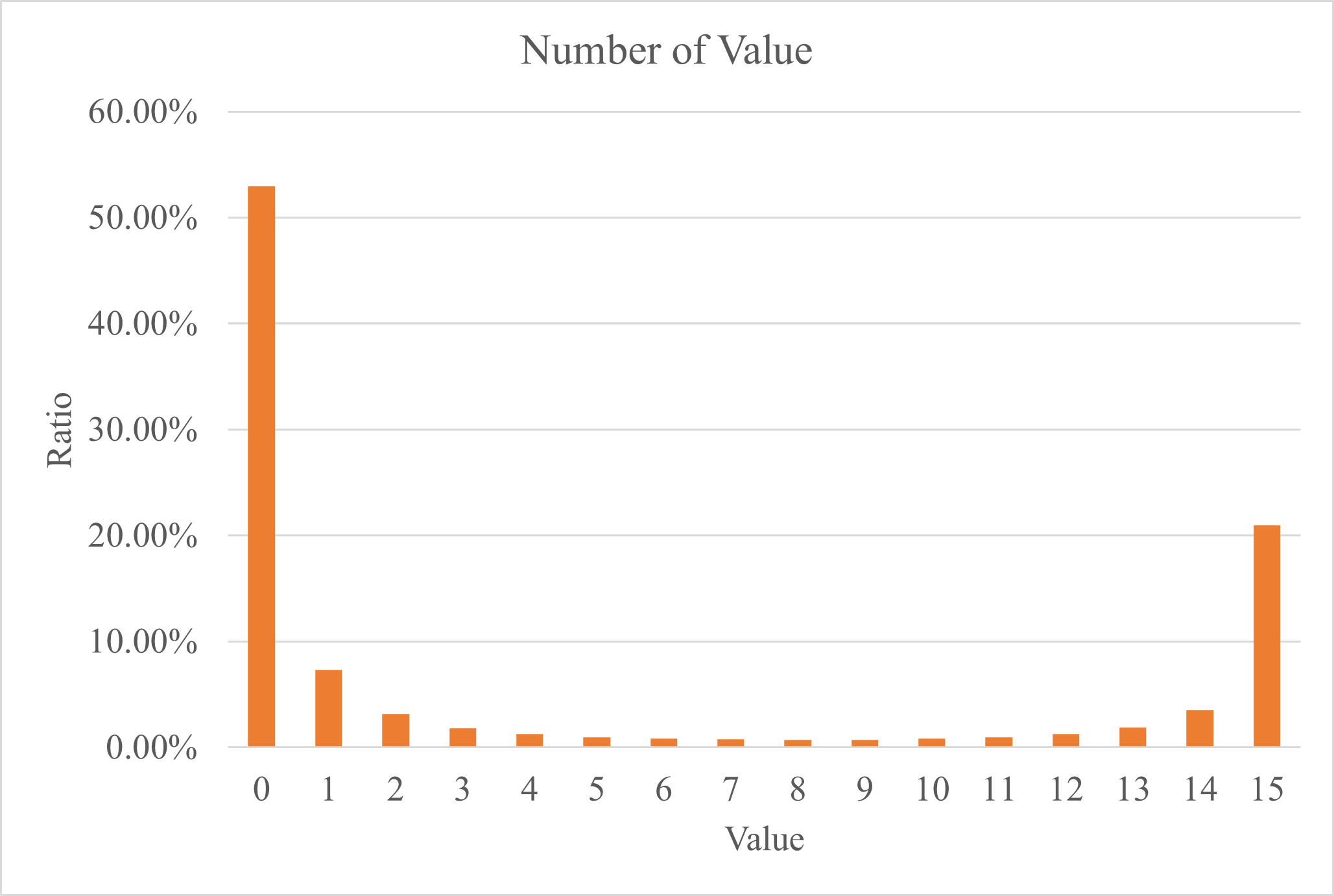


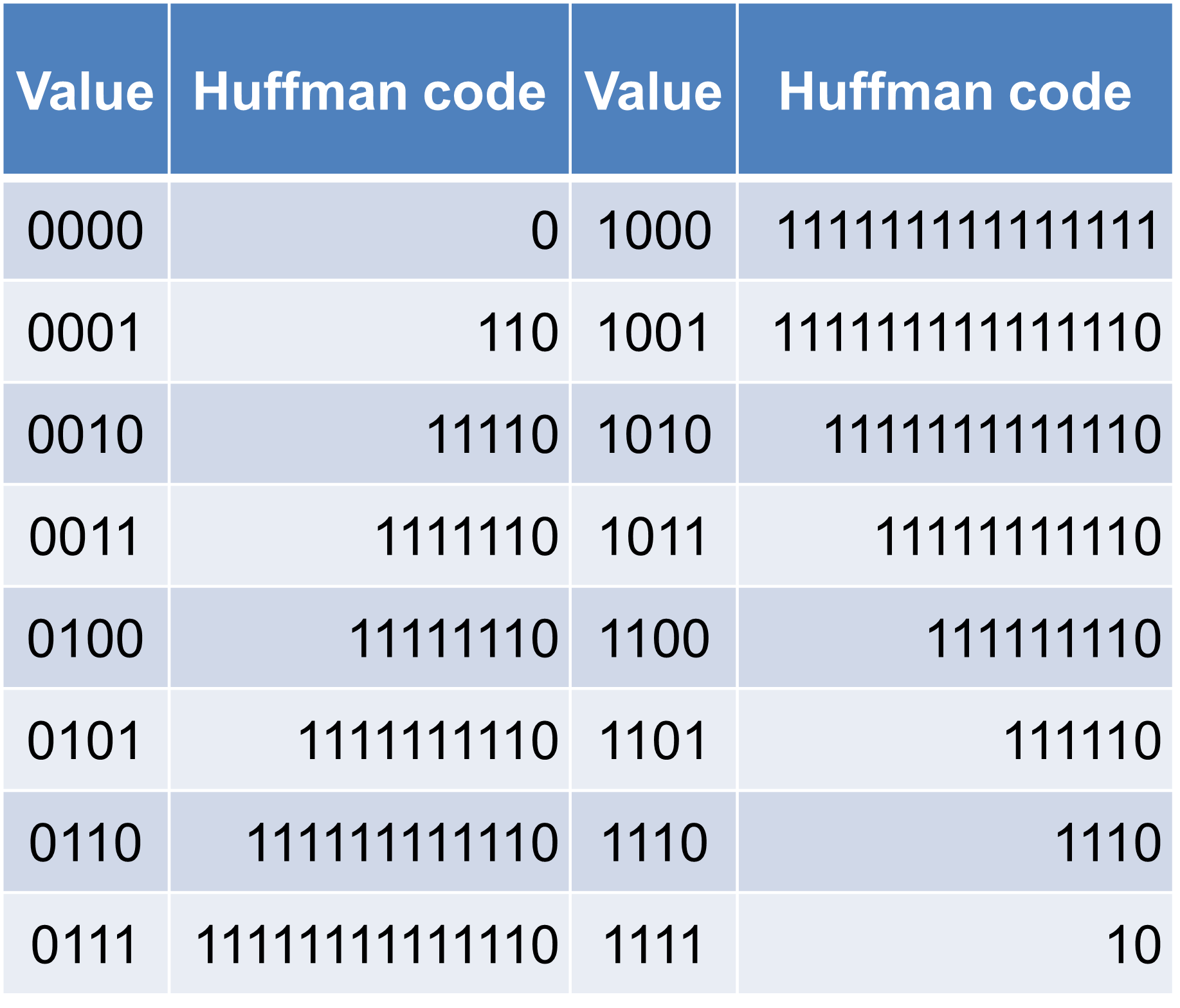
Figure 6 Frequency of Data Occurrence

* + 1. Layer 4: Data compression layer

In this layer, we first divide the data of each pixel into groups of 4 bits and apply Huffman coding to each group individually. As a result, the number of data after this layer will be M times the original. However, due to Huffman coding, the total number of output bits will be fewer than the input.

The pre-analysis reveals that the frequencies of data occurrences are mostly the same, so we predefine the format of Huffman coding to reduce complexity and speed up the computational speed of the architecture.

Table 1 Huffman Table



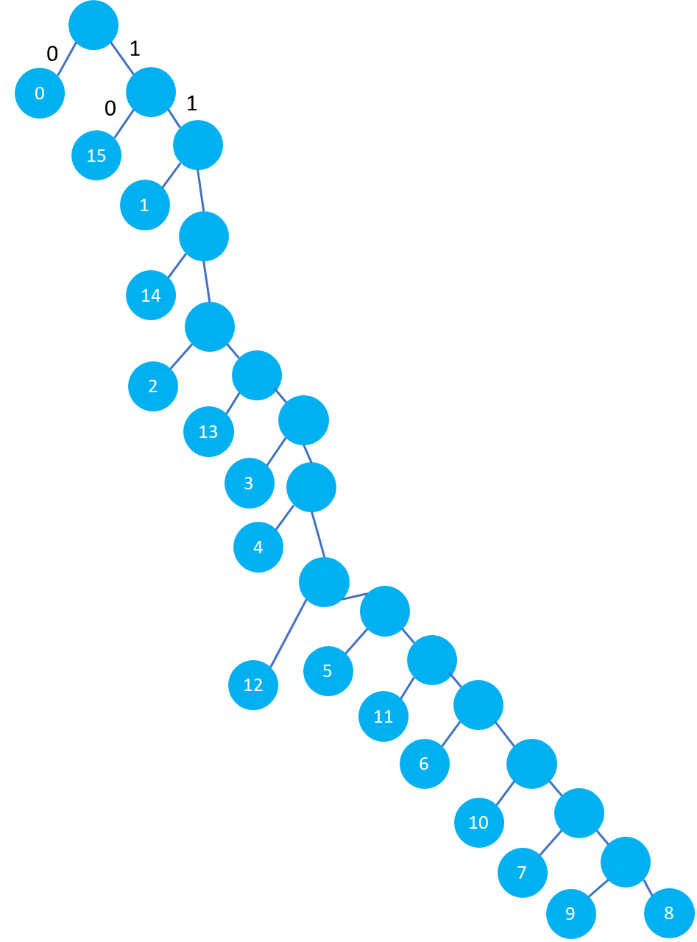


Figure 7 Huffman Tree

1. Specification:
   1. Input image size: 256\*256\*8 (pixel\*pixel\* pixel width)
   2. Output data size: 256\*512\*N (N is unknown)

Hint: Huffman coding is a variable-length data compression technique.

1. Result:
   1. Demo:
      1. Execution time: 13453 .

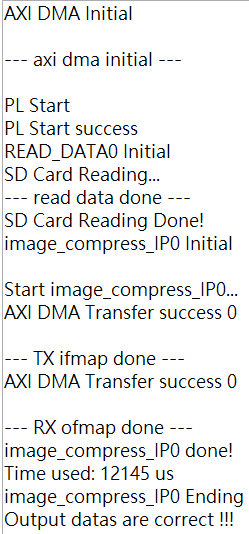


Figure 8 SDK Terminal

* + 1. The average compression ratio is approximately 31%.
    2. Average PSNR (Peak Signal Noise Ratio) above than 42dB.
    3. Compare:

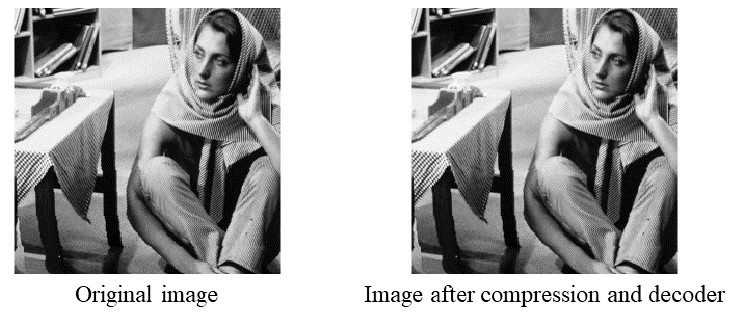


Figure 9 Barara Gray Image



Figure 10 Goldhill Gray Image

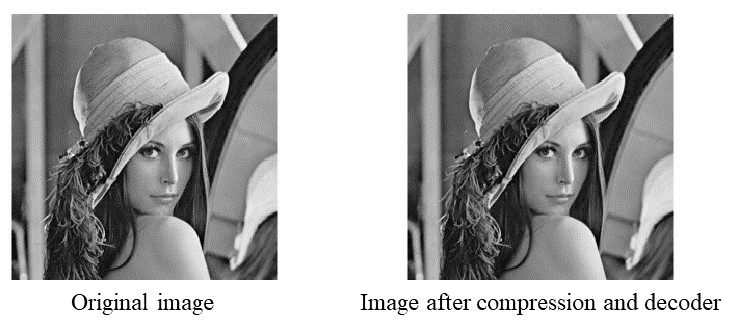


Figure 11 Lena Gray Image

* 1. Implementation Result
     1. Block Design

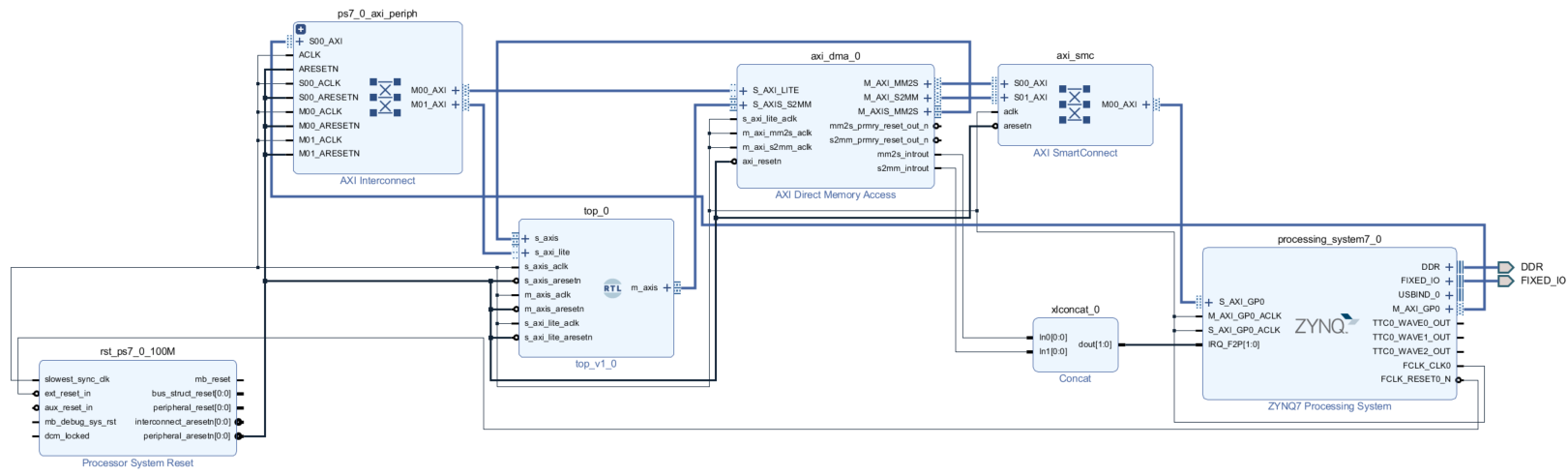


Figure 12 Block Design Diagram

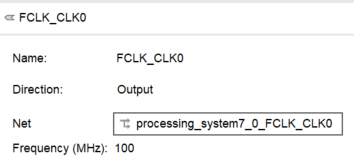


Figure 13 Clock Frequency

* + 1. Synthesis and Implementation Report:

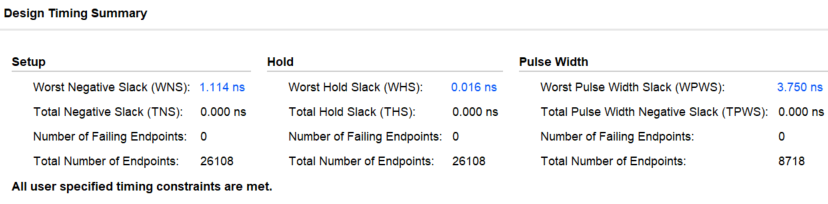


Figure 14 Timing Summary

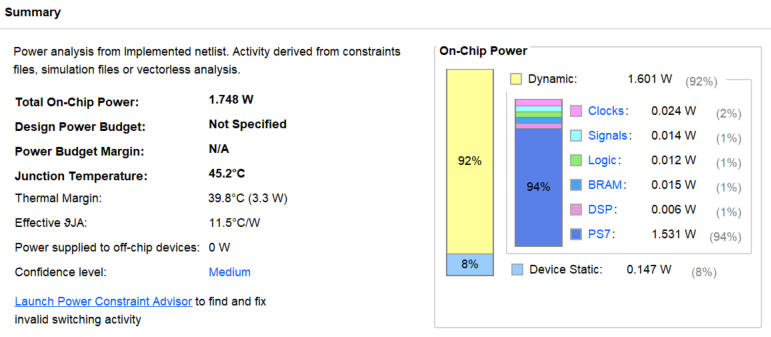


Figure 15 Power Summary

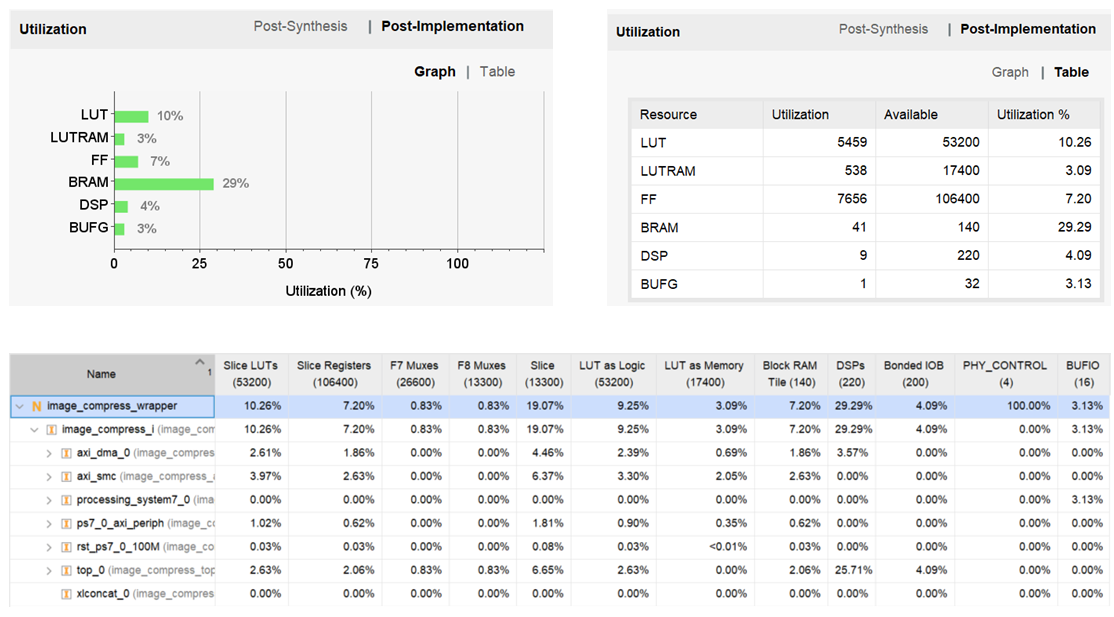


Figure 16 Utilization

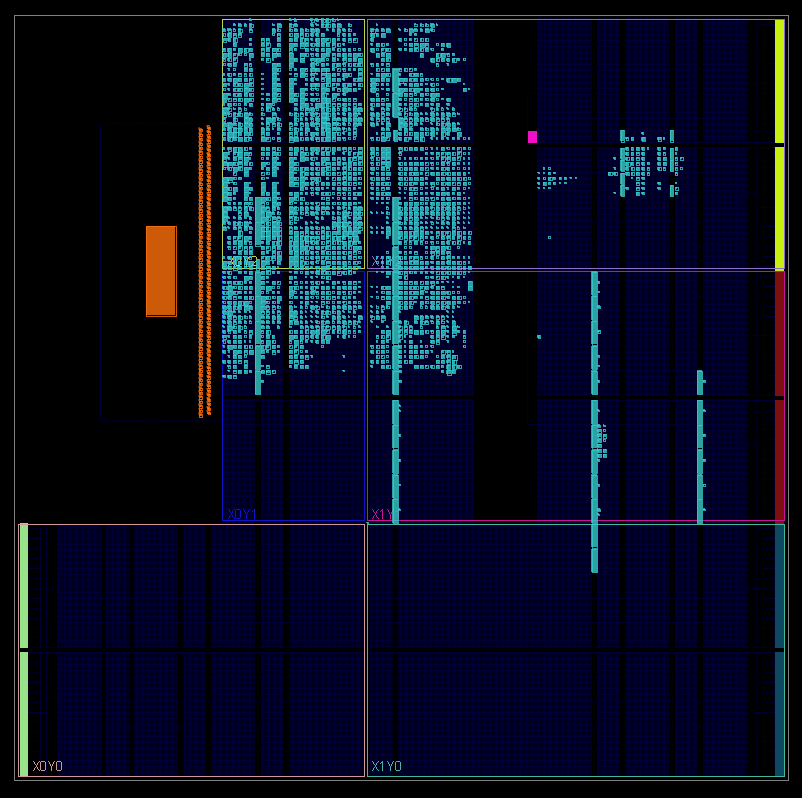


Figure 17 Implemented Visual

1. Acknowledgment:
   1. NCHU EE Undergraduate:
      1. Chun-Wei Su
   2. NCHU EE MSPIC Lab. (611B):
      1. Kun-Yung Chang
   3. NCHU EE ICs & Systems Lab. (612):
      1. Hung-Jui Chang
      2. Mo-Hsuan Hsiung
      3. Shun-Liang Yeh
      4. Chun-Yuan Hsiao
   4. NCHU EE Lab. 716:
      1. Hsing-Yao Wang
      2. Hsuan-Yu Lin
   5. NCHU EE VSIP-IC Lab. (908):
      1. Wen-Chia Yang
2. Reference:
3. Po-Wei Liu, “The Implementation of Image Compression JPEG2000,” M.S. thesis, Dept. Elect. Eng., DYU, Changhua, Taiwan, 2014.
4. M.Puttaraju, and Dr.A.R.Aswatha “FPGA Implementation of 5/3 Integer DWT for Image Compression” International Journal of Advanced Computer Science and Applications, Vol. 3, No. 10, 2012
5. G. K. Khan and A. G. Sawant, "Spartan 6 FPGA implementation of 2D-discrete wavelet transform in Verilog HDL," 2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT), 2016, pp. 139-143, doi: 10.1109/ICAECCT.2016.7942570
6. Hardware Design of the Discrete Wavelet Transform: an Analysis of Complexity, Accuracy and Operating Frequency Dora M. Ballesteros L. 1, Diego Renza 2 and Luis Fernando Pedraza 3 Received: 28-04-2016 | Accepted: 21-10-2016 | Online: 18-11-2016 PACS: 84.40.Ua; 07.50.Qx doi:10.17230/ingciencia.12.24.6
7. https://www.cnblogs.com/chengqi521/p/6732999.html
8. https://www.cnblogs.com/amxiang/p/16543664.html
9. https://zhuanlan.zhihu.com/p/608277782
10. https://blog.csdn.net/weixin\_38071135/article/details/118581250