

# Team ID: 3

Student1 ID name: R08943003 姚鈞嚴

Student2 ID name: R08943023 陳博仁

<i>RTL category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
RTL Simulation	filename.v	Verilog (or VHDL) synthesizable RTL code
<i>Gate-Level category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
Pre-layout	filename_syn.v	Verilog gate-level netlist generated by Synopsys
Gate-level		Design Compiler
Simulation	filename_syn.sdf	Pre-layout gate-level sdf
<i>Physical category</i>		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
P&R	filename.tar	<b>archive of the design database directory</b>
	filename.gds	GDSII layout <i>Attach the screenshot of streamout gds below below this table</i>
	Error_number	Number of DRC error(s) <i>Attach the screenshot below this table</i>
	Error_number	Number of LVS error(s) <i>Attach the screenshot below this table</i>
Post-layout	filename_pr.v	Verilog gate-level netlist generated by Cadence
Gate-level		Encounter or Synopsys IC Compiler
Simulation		filename_pr.sdf

Post-route Setup time screenshot:

Post-route Hold time screenshot:

Stream out GDS screenshot:

DRC report screenshot:

LVS report screenshot:

Annotation (optional, for TA to reproduce you design easily):

Briefly explain your design and the problem you encounter:

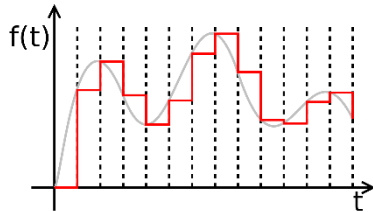


Fig. 1. Time-domain view of zero-order hold [1].

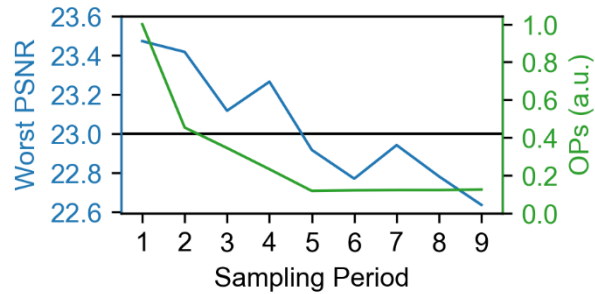


Fig. 2. Plot of PSNR and average number of operations (OPs) against sampling period.

From the view point of software, our design features in down sampling, shrunk search range, and early skipping via cost threshold. They are explained as follows.

- a) *Down Sampling*: Fig. 1 shows an example that how motion vector relates to zero-order hold. Let the original signal  $f(t)$  (gray line) be one-dimensional position of one block. After down sampling and interpolating it by zero-order hold, one can get the step-shaped signal (red line). As an analogy to motion vector, after down sampling, we can use zero motion vector to describe the flat segment, and use valued motion vector to describe the steep segment. Since zero-order hold does not require information from frames, the computation complexity can be reduced. Fig. 2 illustrates the relations of PSNR, number of operations to sampling period. To define an operation (the definition is all the same throughout the report), it can be receiving one pixel outside the design, loading one pixel to PE (including padding), accumulating one cost from a pair of pixels, and outputting one motion vector. The result shows that to keep qualified PSNR, the maximum sampling period is four, so the reduction of operations is 76.8 %.
- b) *Shrunk Search Range*: To find a best matching block, one should pad the block with given search range. If we say  $N$  is block size, and  $p$  is search range, the number of operations required to load one pixel to PE is

$$N_{OP} = N^2 + (N + p)^2, \quad (1)$$

where the first term is for the block of current frame, and the second term is for

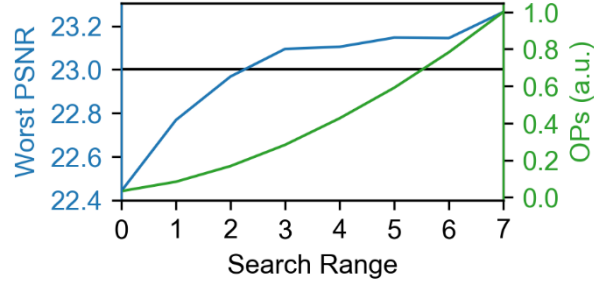


Fig. 3. Plot of PSNR and OPs against size of search range.

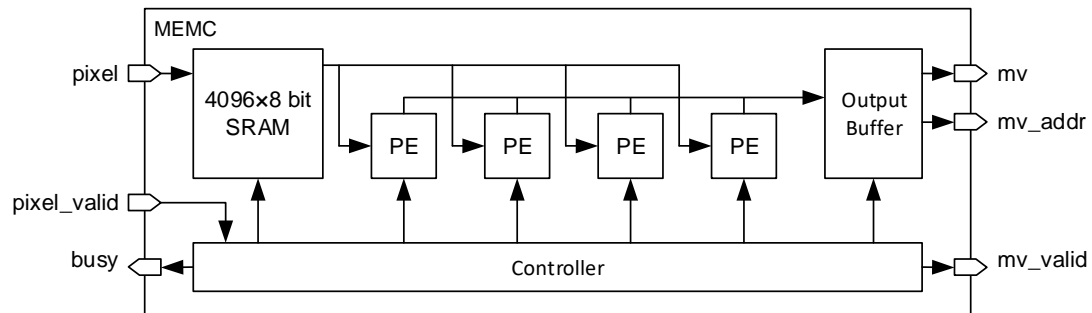
the block of the previous frame. Therefore, if  $p$  decreases,  $N_{OP}$  decreases by second order. Fig. 3 illustrates the result of decreasing search range. The task is passed when  $p \geq 3$ , and the reduction of operations here is 71.7 %.

- c) *Early Skipping via Cost Threshold*: To define an optimal motion vector in each block, one should find out the best matching block of which the cost is the lowest among all candidate blocks. However, during the cost accumulation process, if we are very sure that the cost of one matching block is larger than the current recorded minimum cost, we can skip this block and check the next matching block. Moreover, we also introduced two approaches to skip more cost-accumulating operations: 1) When calculating the cost of the first matching block, there is no reference minimum, so an initial value should be given. If we decrease it, number of operations can be reduced although the cost of the best matching block might be higher than this value. 2) If we find that the cost of one matching block is lower than a given threshold, we can also regard it as the best matching block and skip the finding process. After adding these early skipping checks, we can reduce the operations by 9.4 %.

In terms of hardware, we regard a block whose function is to find out the optimal motion vector as a process element (PE). Parallelism of PEs can decrease loading repetitive padding pixels due to search range, execute more operations in one cycle, and as a result, increase the speed. However, parallelism implies more area. At last, we set the parallel factor to 4. In addition, computing motion vectors requires two frames, so we have to store  $48 \times 64 \times 2 = 6144$  bytes. Since only SRAMs with size of 256, 512, 4096, and 16384 are given, the most efficient way is to implement two SRAMs. However, based on our simulation result, the pixel value can be truncated by 4 bits. This means that we can combine two pixels and store them in the same address in SRAM. After truncating, we can stuff all relating data into one 4096-byte SRAM.

Fig. 4 shows the overall architecture of our MEMC design. The controller determines how input pixels store into the 4096-byte SRAM, how data in the SRAM issue to the four PEs, how the PEs output the motion vectors to the output buffer, and

how the output buffer outputs the result. Because of the high complexity to determine the valid signal to each PE, we padded the frame by zero to the SRAM beforehand. Based on the simulation results, only full-search block matching algorithm can reach the designated PSNR. In the PE, we therefore compute every absolute block difference between previous frame and new frame one by one from top left and find the most matching block in the end. In order to accelerate the process of finding the most matching block, we implemented two accumulators in each PE, reducing almost half of latency while remaining competitive area cost.



Acknowledgement:

We show our great appreciation to Da-Wei Lin, a research member in Energy-Efficient Circuit and System Lab, Graduate Institute of Electronics Engineering, National Taiwan University. He shared the method of shrunk search range and thus helped us make our design better.

Reference:

- [1] "Zero-order hold", *En.wikipedia.org*, 2020. [Online]. Available: [https://en.wikipedia.org/wiki/Zero-order\\_hold](https://en.wikipedia.org/wiki/Zero-order_hold).