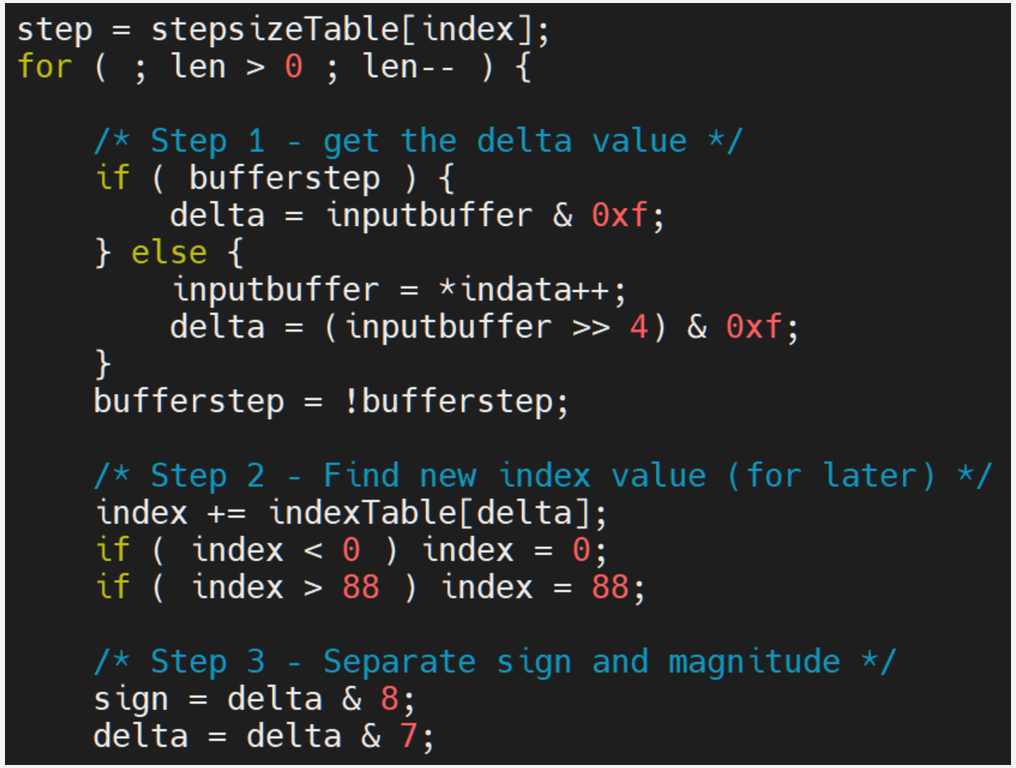
ASIP CPU design

Ruoheng Ma
Supervisor: Sajjad Hussain

10.05.2021

Application Source Code

- Step 7 is output step and not changed, so ignored in this slide

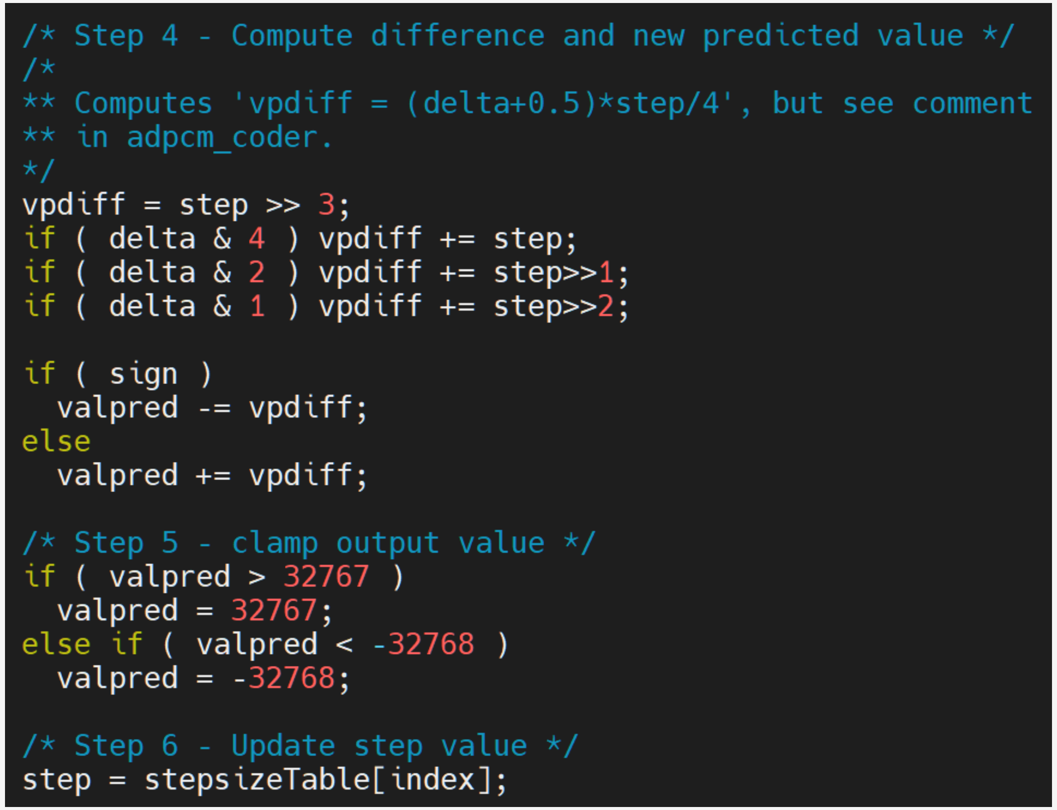


```c
step = stepsizeTable[index];
for ( ; len > 0 ; len-- ) {

    /* Step 1 - get the delta value */
    if ( bufferstep ) {
        delta = inputbuffer & 0xf;
    } else {
        inputbuffer = *indata++;
        delta = (inputbuffer >> 4) & 0xf;
    }
    bufferstep = !bufferstep;

    /* Step 2 - Find new index value (for later) */
    index += indexTable[delta];
    if ( index < 0 ) index = 0;
    if ( index > 88 ) index = 88;

    /* Step 3 - Separate sign and magnitude */
    sign = delta & 8;
    delta = delta & 7;
```



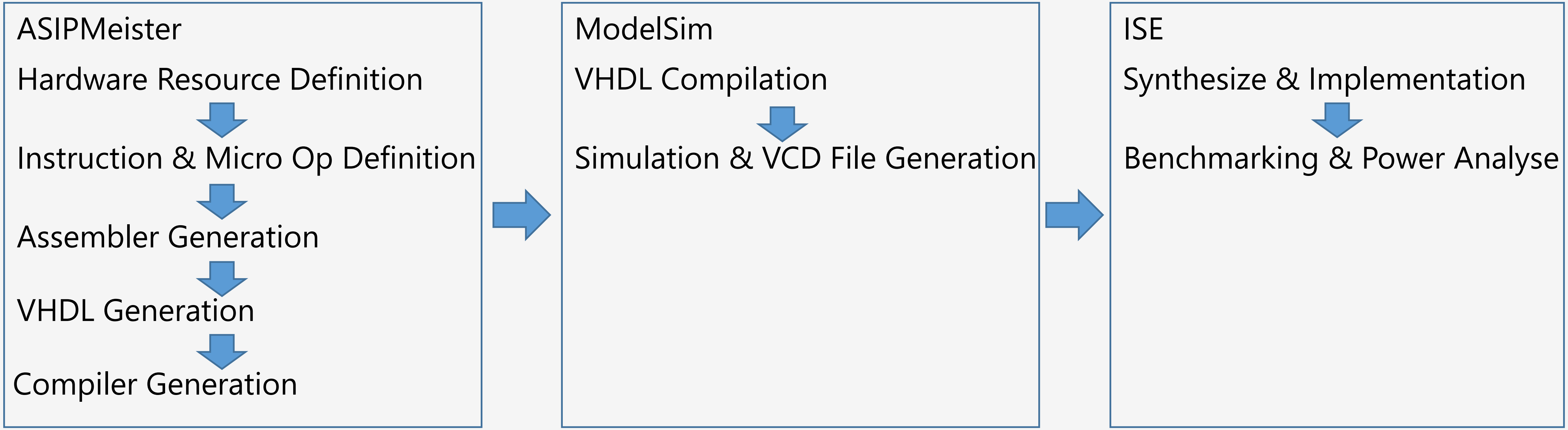
```c
/* Step 4 - Compute difference and new predicted value */
/*
** Computes 'vpdiff = (delta+0.5)*step/4', but see comment
** in adpcm_coder.
*/
vpdiff = step >> 3;
if ( delta & 4 ) vpdiff += step;
if ( delta & 2 ) vpdiff += step>>1;
if ( delta & 1 ) vpdiff += step>>2;

if ( sign )
  valpred -= vpdiff;
else
  valpred += vpdiff;

/* Step 5 - clamp output value */
if ( valpred > 32767 )
  valpred = 32767;
else if ( valpred < -32768 )
  valpred = -32768;

/* Step 6 - Update step value */
step = stepsizeTable[index];
```

Workflow



ASIPMeister

Hardware Resource Definition

Instruction & Micro Op Definition

Assembler Generation

VHDL Generation

Compiler Generation

ModelSim

VHDL Compilation

Simulation & VCD File Generation

ISE

Synthesize & Implementation

Benchmarking & Power Analyse

Problem

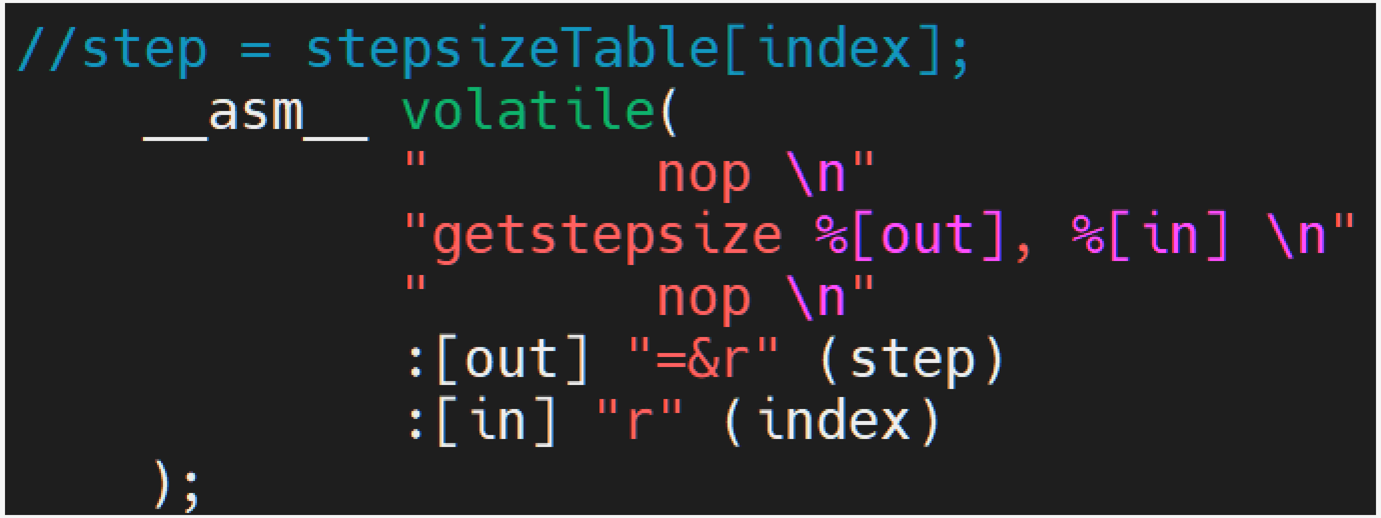
- No access to FPGA board
  - Slowest frequency is ignored

CPU - Performance Improvement

- Stepsize getter
- Index getter
- Decode accelerator

CPU - Performance Improvement

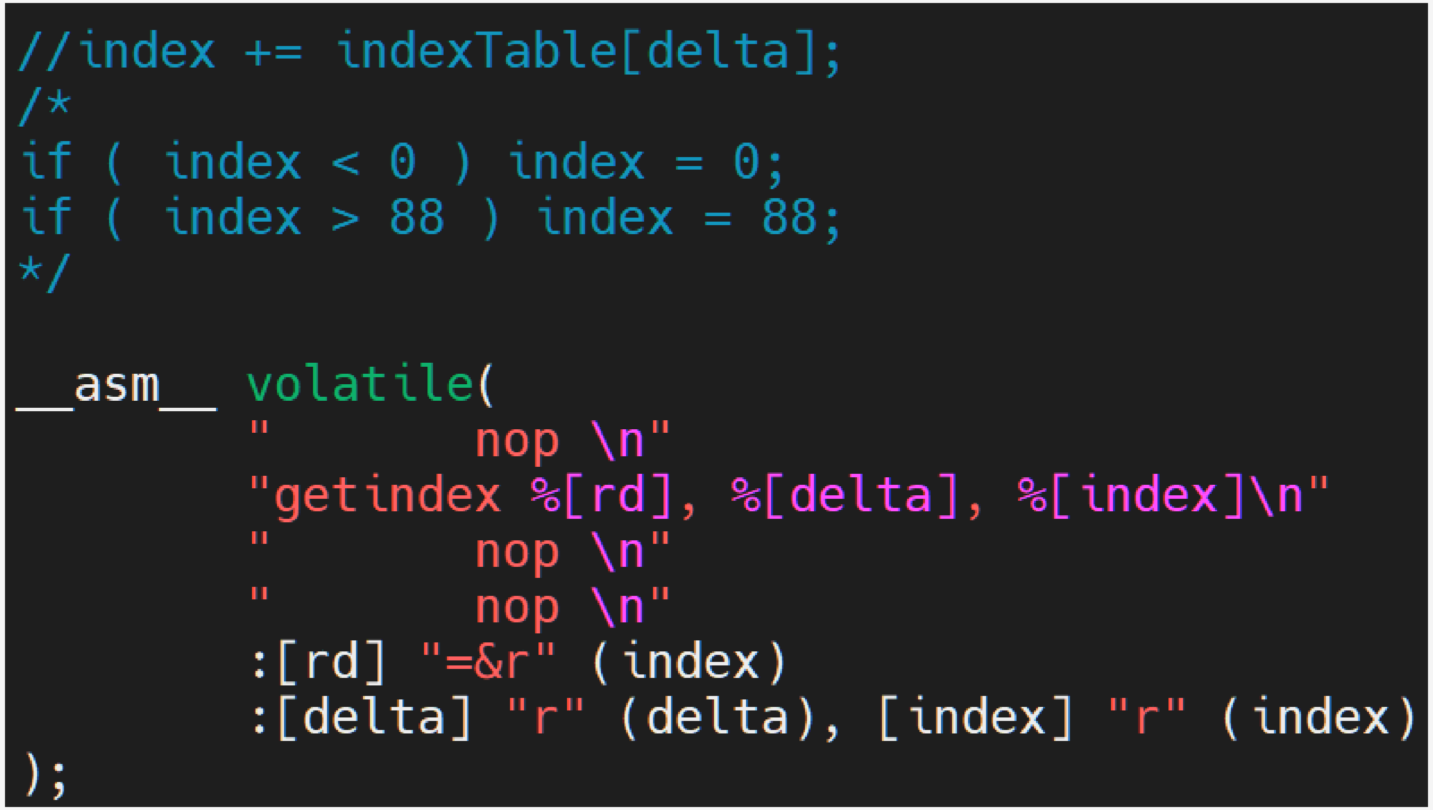
- Stepsize getter



```
//step = stepsizeTable[index];
    __asm__ volatile(
            "       nop \n"
            "getstepsize %[out], %[in] \n"
            "       nop \n"
            :[out] "=&r" (step)
            :[in] "r" (index)
    );
```

CPU - Performance Improvement

- Index getter

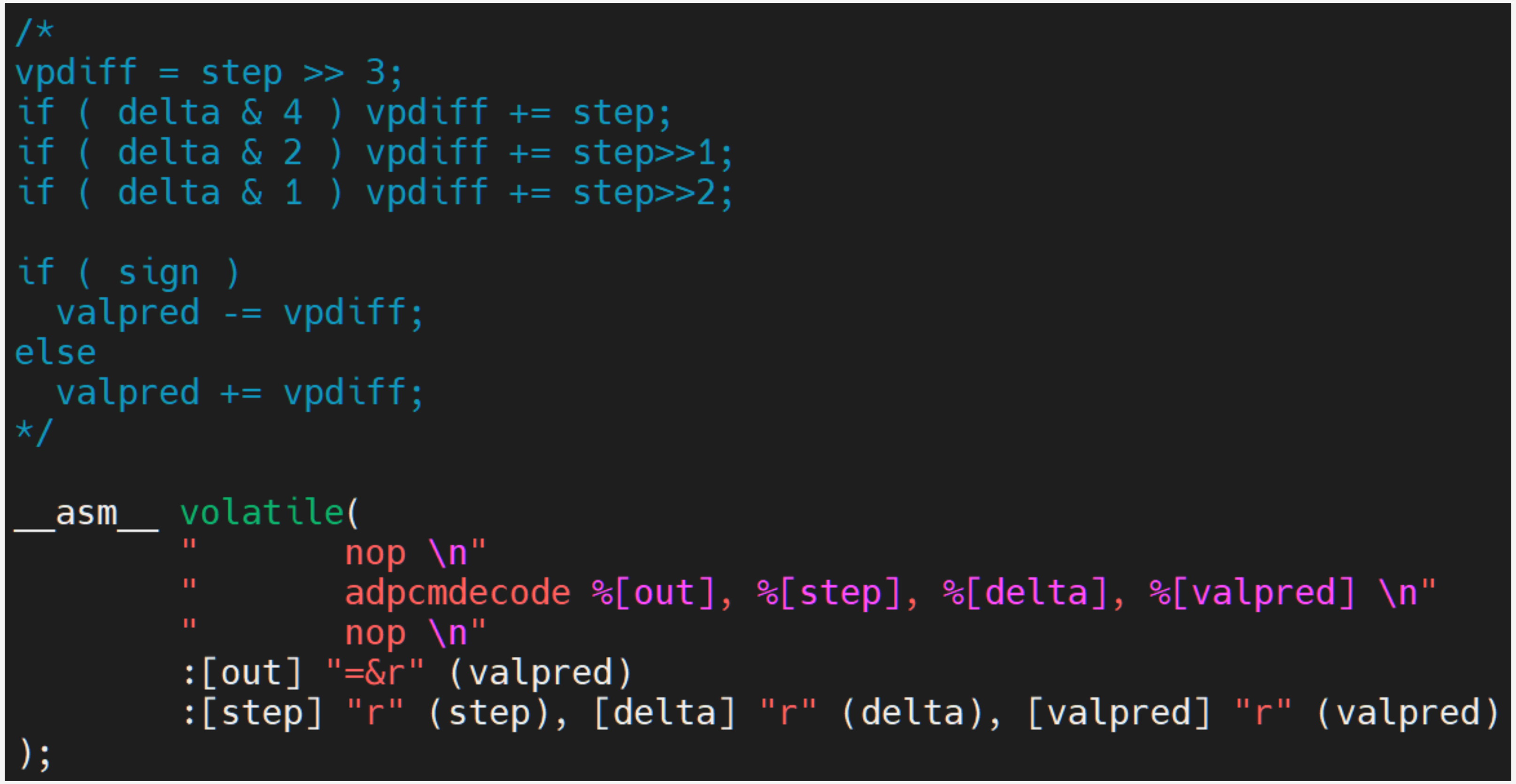


```
//index += indexTable[delta];
/*
if ( index < 0 ) index = 0;
if ( index > 88 ) index = 88;
*/

__asm__ volatile(
        "       nop \n"
        "getindex %[rd], %[delta], %[index]\n"
        "       nop \n"
        "       nop \n"
        :[rd] "=&r" (index)
        :[delta] "r" (delta), [index] "r" (index)
);
```

CPU - Performance Improvement

- Decode accelerator



```
/*
vpdiff = step >> 3;
if ( delta & 4 ) vpdiff += step;
if ( delta & 2 ) vpdiff += step>>1;
if ( delta & 1 ) vpdiff += step>>2;

if ( sign )
  valpred -= vpdiff;
else
  valpred += vpdiff;
*/

__asm__ volatile(
        "       nop \n"
        "       adpcmdecode %[out], %[step], %[delta], %[valpred] \n"
        "       nop \n"
        :[out] "=&r" (valpred)
        :[step] "r" (step), [delta] "r" (delta), [valpred] "r" (valpred)
);
```

CPU - Performance Improvement

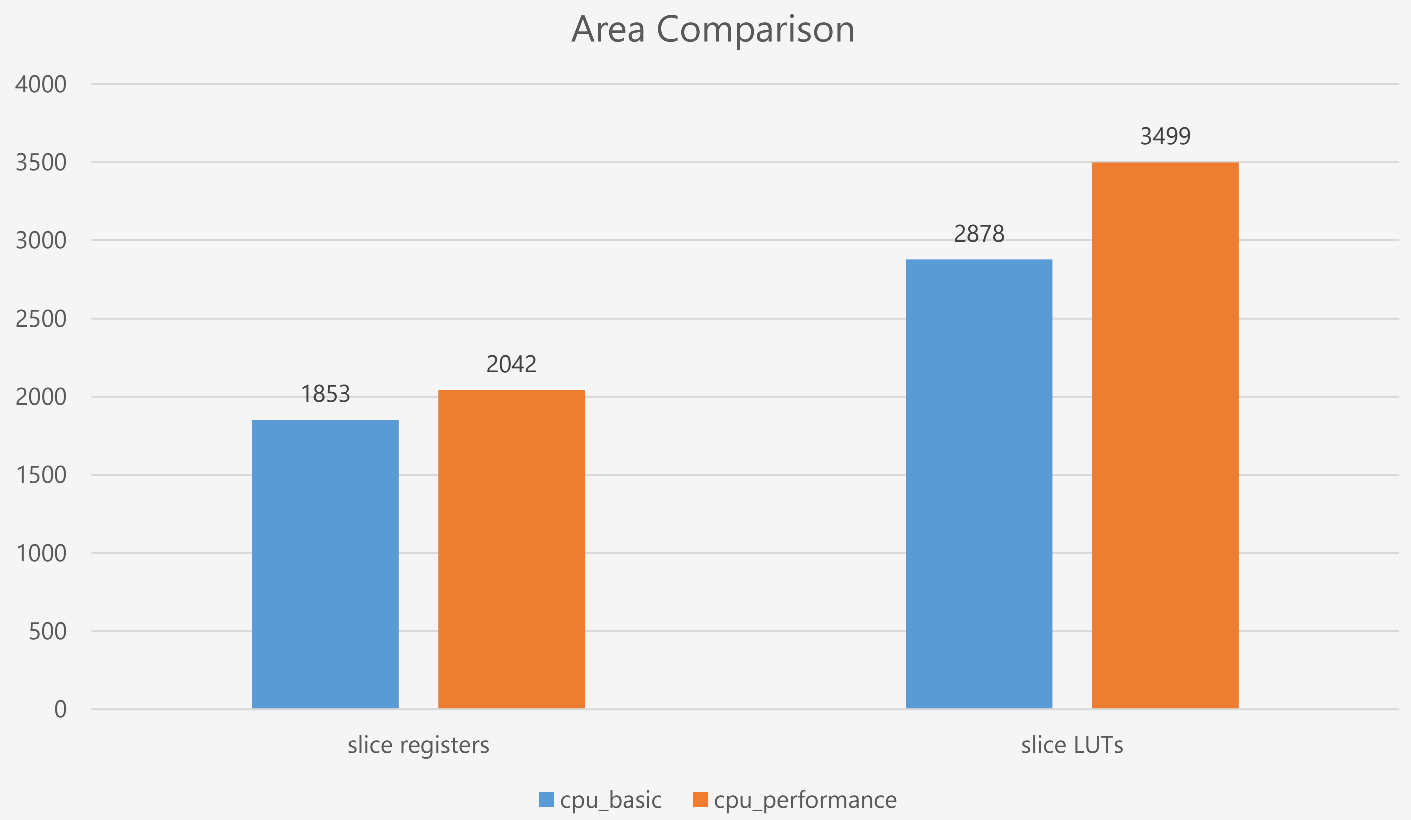
- Problem
  - Forwarding unit does not work properly

CPU - Performance Improvement

- Benchmark result – maximal frequency & critical path
- cpu_basic
  - 100.341 MHz
  - 9.966 ns
- cpu_performance
  - 100.422 MHz
  - 9.958 ns

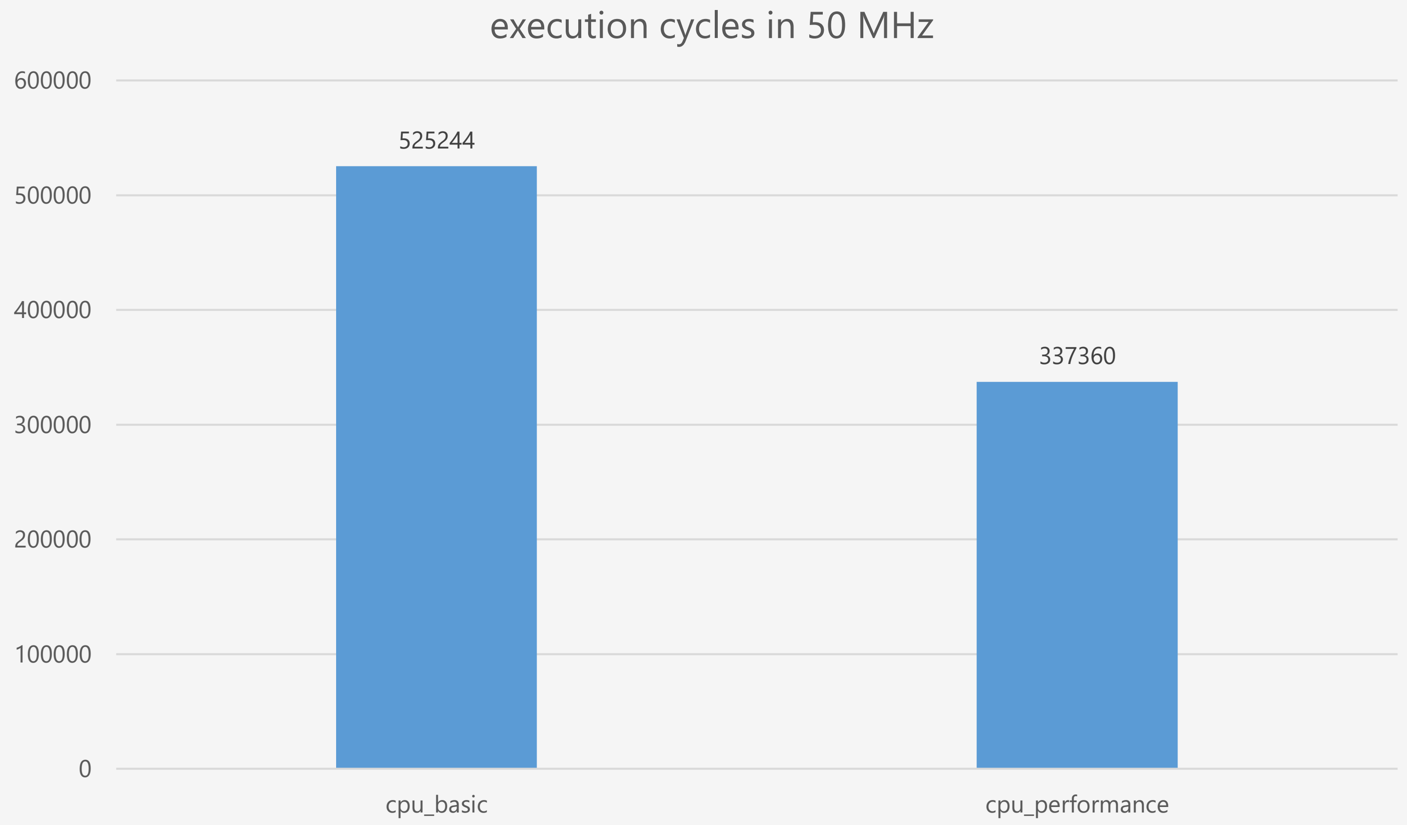
CPU - Performance Improvement

- Benchmark result – area



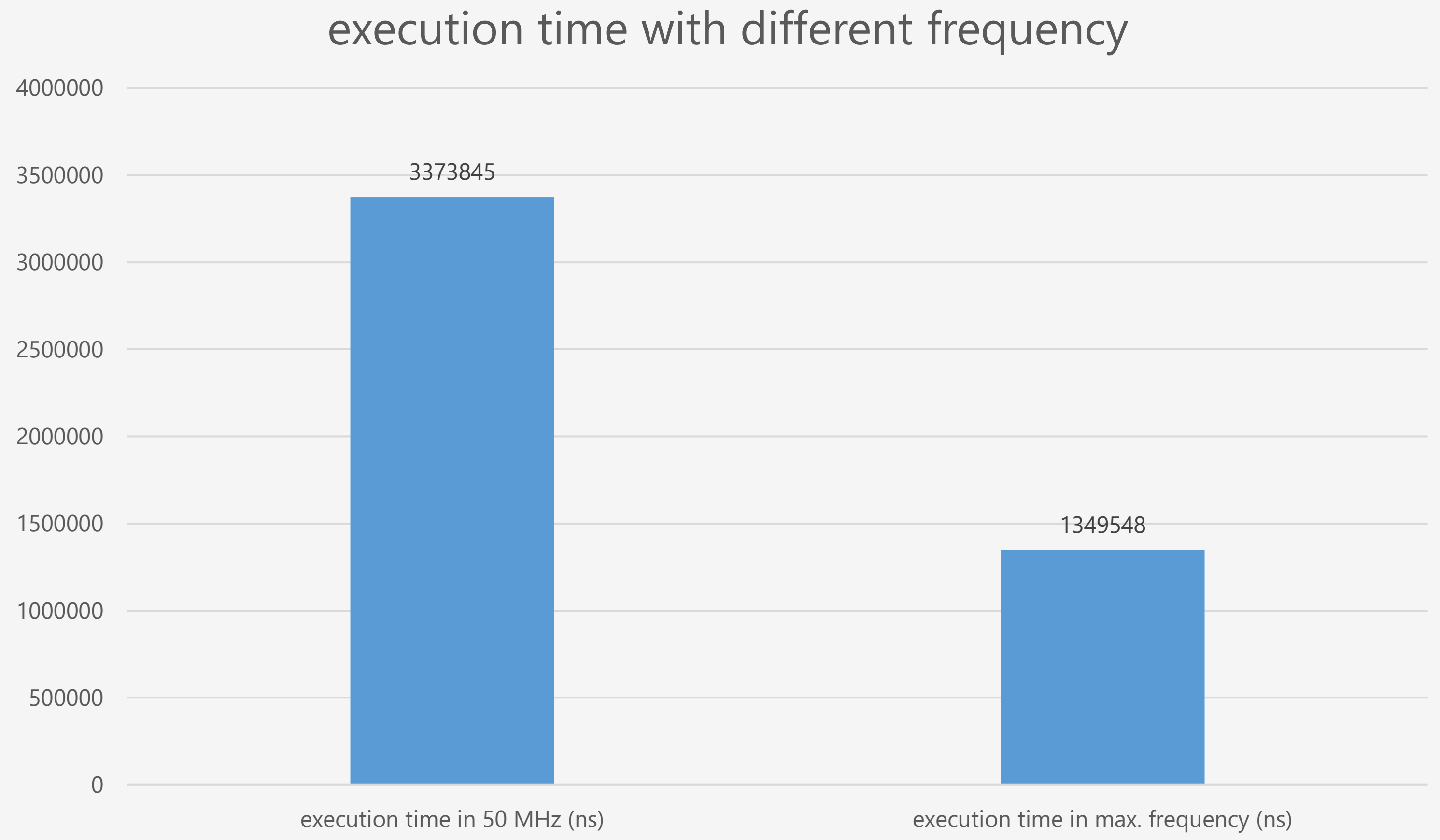
CPU - Performance Improvement

- Benchmark result – execution cycles comparison with basic CPU



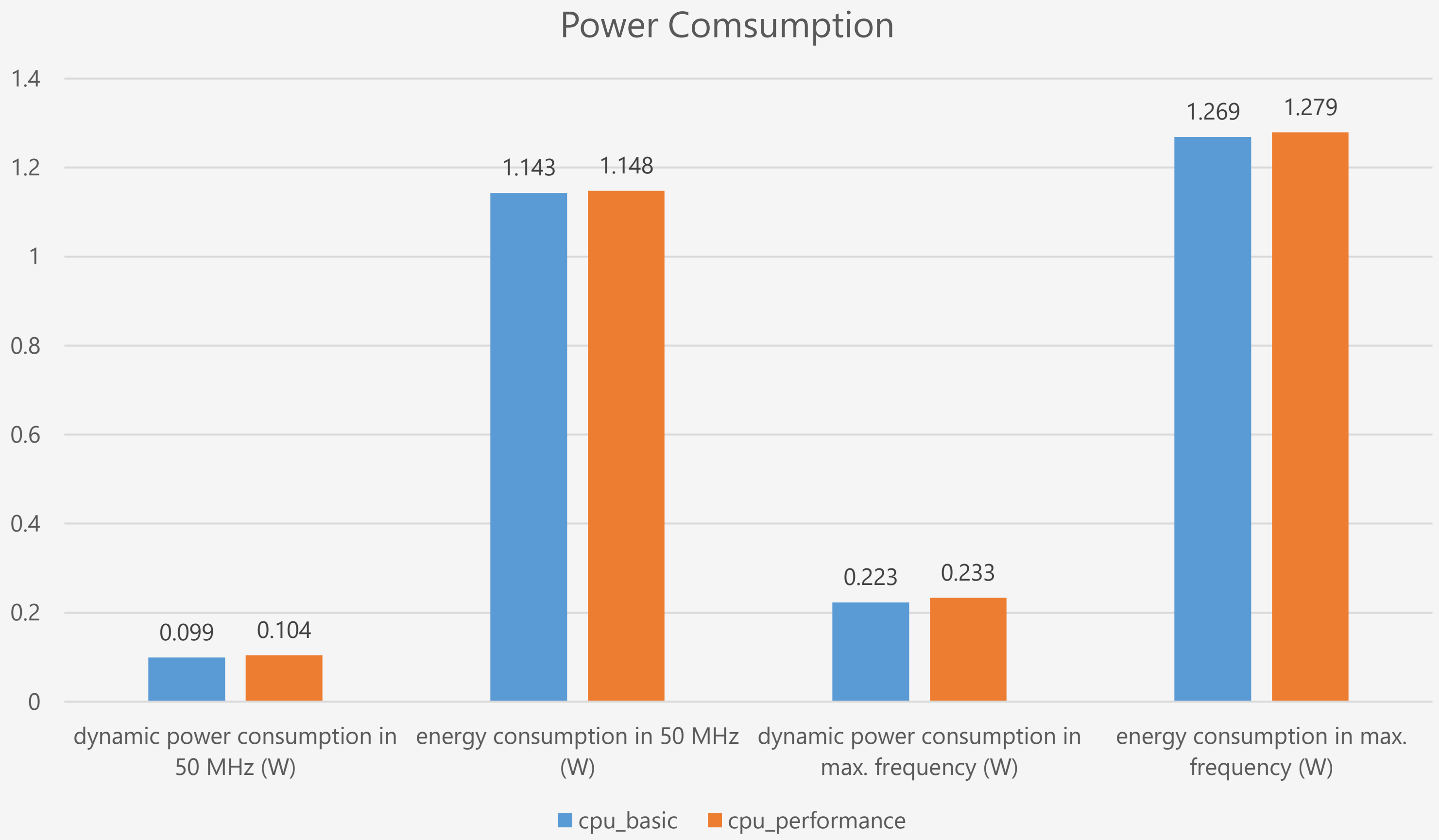
CPU - Performance Improvement

- Benchmark result – execution time with different frequency (vertical axis: ns)



CPU - Performance Improvement

- Benchmark result – power consumption (vertical axis: W)

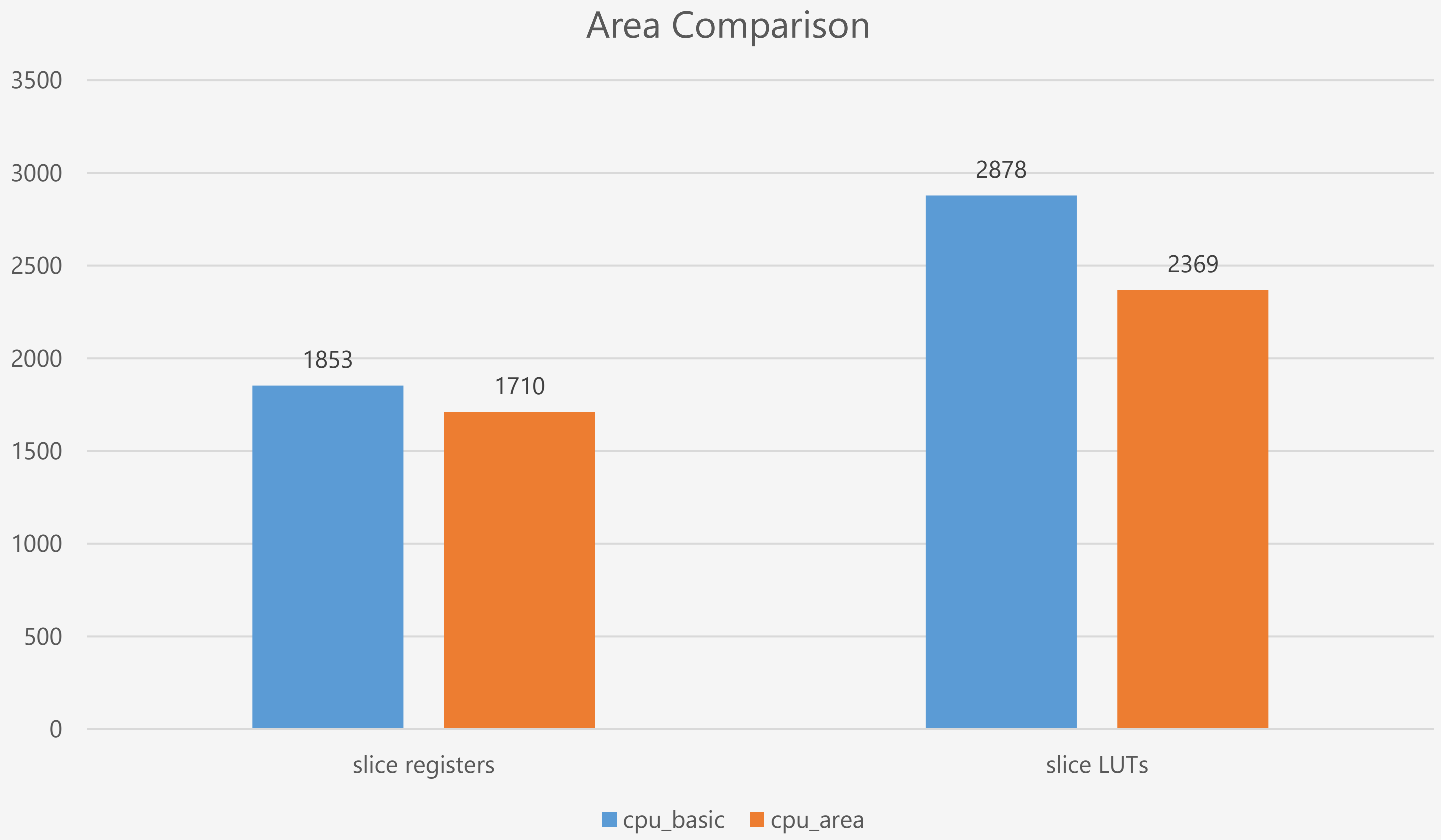


CPU - Area Improvement

- Removed hardware resource
  - MUL
- Removed instructions
  - mul
  - div
  - divu
  - modu

CPU - Area Improvement

- Benchmark result – area

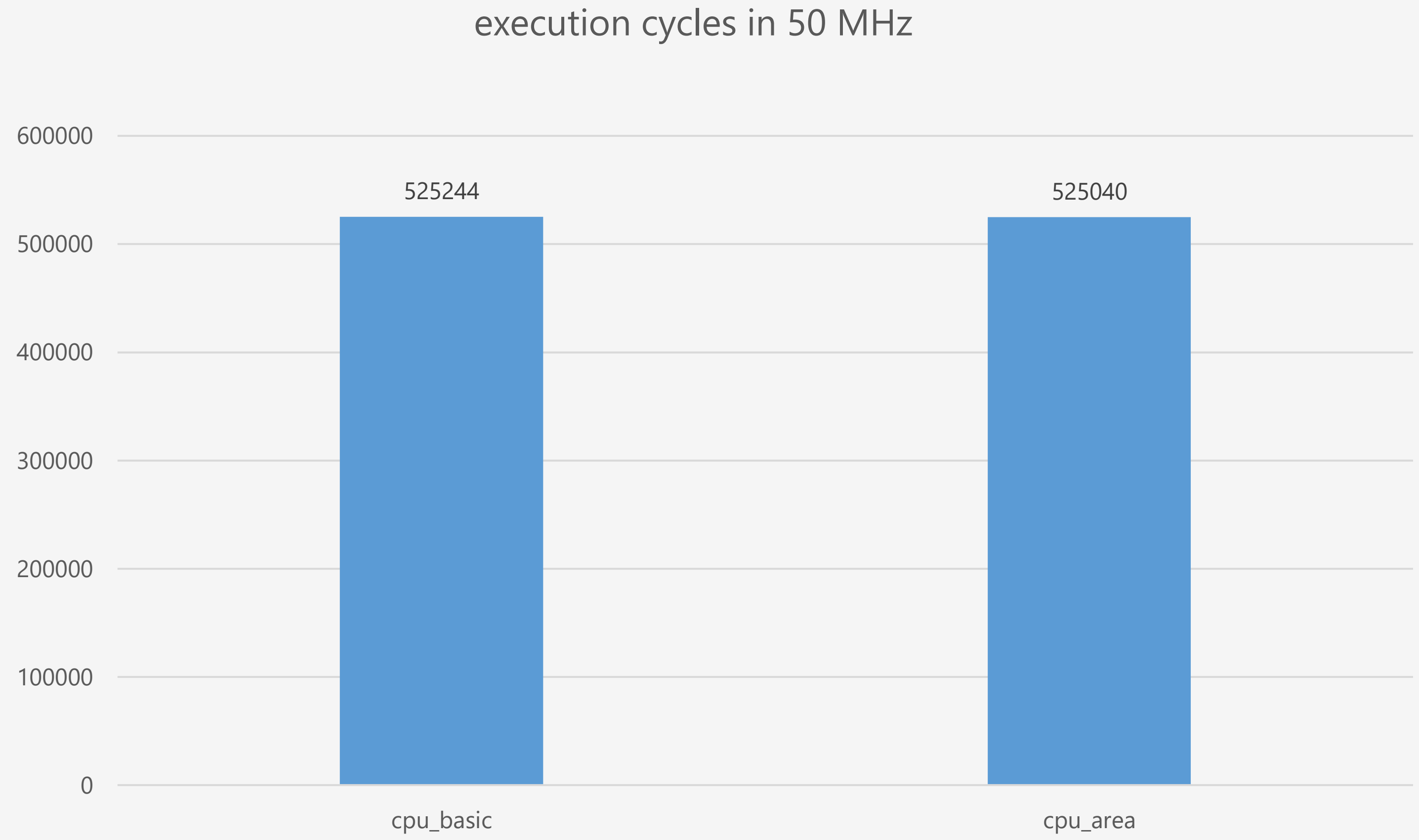


CPU - Area Improvement

- Benchmark result – maximal frequency & critical path
- cpu_basic
  - 100.341 MHz
  - 9.966 ns
- cpu_area
  - 100.675 MHz
  - 9.933 ns

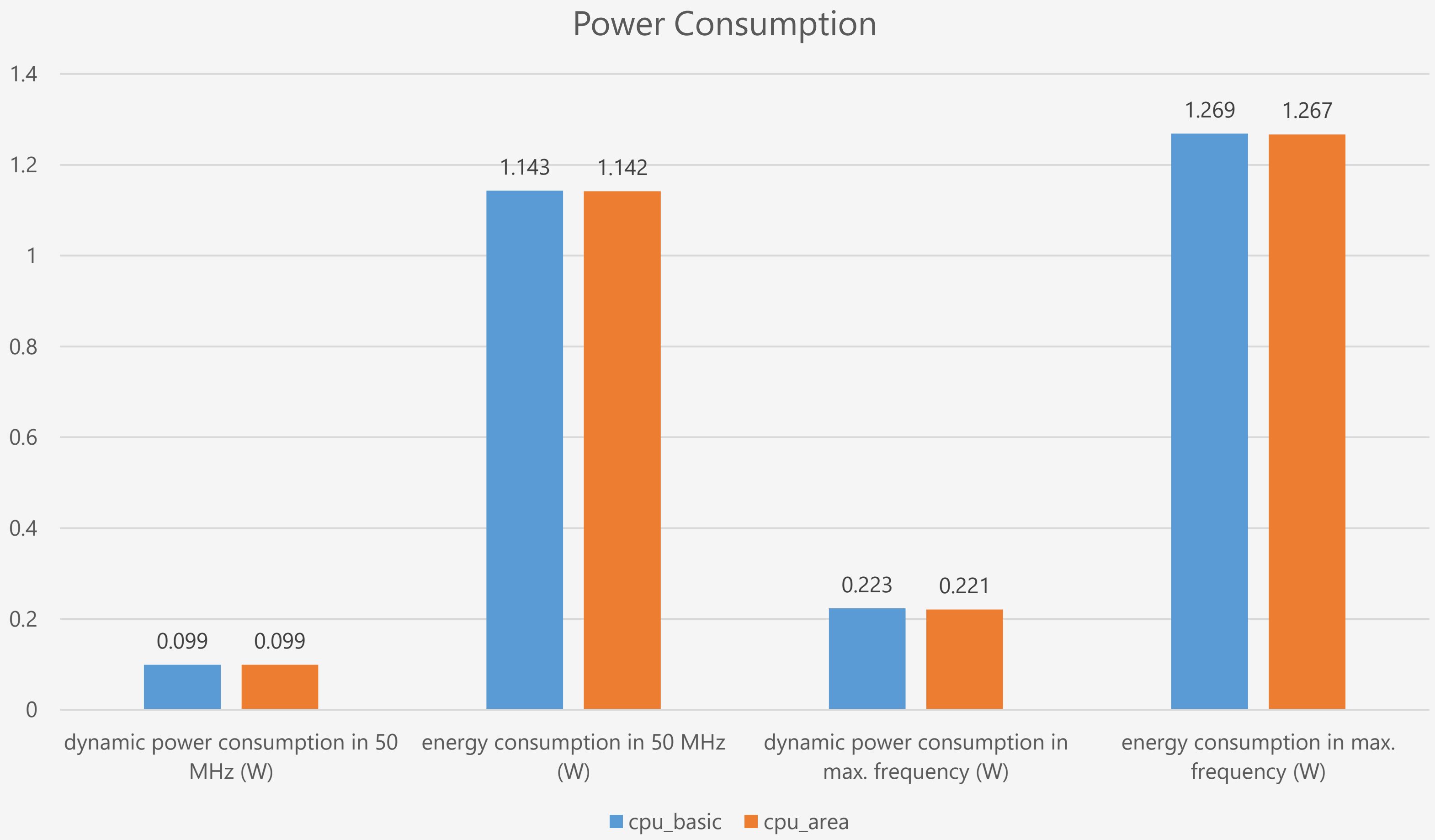
CPU - Area Improvement

- Benchmark result – execution cycles comparison with basic CPU



CPU - Area Improvement

- Benchmark result – power consumption (vertical axis: W)



Thank you for your attention!