



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Отчёт

по лабораторной работе №5

Название «Разработка ускорителей вычислений средствами
высокоуровневого синтеза»

Дисциплина «Архитектура ЭВМ»

Вариант 3

Студент ИУ7-55Б

(подпись, дата)

Клименко А.К.

(Фамилия И.О.)

Преподаватель

(подпись, дата)

Попов А.Ю.

(Фамилия И.О.)

Москва, 2022

Содержание

| | |
|---|----|
| Введение | 3 |
| 1 Индивидуальное задание | 4 |
| 2 Сборка и отладка проекта в режиме программной эмуляции (Emulation-SW) | 7 |
| 3 Сборка и отладка проекта в режиме аппаратной эмуляции (Emulation-HW) | 8 |
| 4 Сборка и отладка проекта в режиме аппаратного исполнения (Hardware) . | 9 |
| Заключение | 12 |

Введение

Целью данной работы является изучение методики и технологии синтеза аппаратных устройств ускорения вычислений по описаниям на языках высокого уровня. В ходе лабораторной работы рассматривается маршрут проектирования устройств, представленных в виде синтаксических конструкций ЯВУ C/C++, изучаются принципы работы IDE Xilinx Vitis HLS и методика анализа и отладки устройств.

Для достижения поставленной цели необходимо выполнить следующие **задачи**:

- разработать ускоритель вычислений по индивидуальному заданию;
- разработать код для тестирования ускорителя;
- реализовать ускоритель с помощью средств высоко-уровневого синтеза;
- выполнить отладку реализованного ускорителя.

1 Индивидуальное задание

1) Неоптимизированный цикл:

```
1 extern "C"
2 {
3     void var003_no_pragmas(int* c, const int* a, const int* b, const int len)
4     {
5         int minA = a[0];
6         int minB = b[0];
7         for (int i = 1; i < len; i++)
8         {
9             if (minA > a[i])
10            {
11                minA = a[i];
12                c[i] = minA;
13            }
14            else
15            {
16                c[i] = 0;
17            }
18        }
19        for (int i = 1; i < len; i++)
20        {
21            if (minB > b[i])
22            {
23                minB = b[i];
24                c[i] = minB;
25            }
26        }
27    }
28 }
```

2) Конвейерная организация цикла:

```
1 extern "C"
2 {
3     void var003_pipelined(int* c, const int* a, const int* b, const int len)
4     {
5         int minA = a[0];
6         int minB = b[0];
7         for (int i = 1; i < len; i++)
8         {
9             if (minA > a[i])
10            {
11                minA = a[i];
12                c[i] = minA;
13            }
```

```

14         else
15         {
16             c[i] = 0;
17         }
18     }
19     for (int i = 1; i < len; i++)
20     {
21 #pragma HLS PIPELINE
22         if (minB > b[i])
23         {
24             minB = b[i];
25             c[i] = minB;
26         }
27     }
28 }
29 }

```

3) Частично развернутые циклы:

```

1  extern "C"
2  {
3      void var003_unrolled(int* c, const int* a, const int* b, const int len)
4      {
5          int minA = a[0];
6          int minB = b[0];
7          for (int i = 1; i < len; i++)
8          {
9 #pragma HLS UNROLL factor=5
10             if (minA > a[i])
11             {
12                 minA = a[i];
13                 c[i] = minA;
14             }
15             else
16             {
17                 c[i] = 0;
18             }
19         }
20         for (int i = 1; i < len; i++)
21         {
22 #pragma HLS UNROLL factor=5
23             if (minB > b[i])
24             {
25                 minB = b[i];
26                 c[i] = minB;
27             }
28         }

```

```
29     }
30 }
```

4) Конвеерный и частично развернутый циклы:

```
1  extern "C"
2  {
3      void var003_pipe_unroll(int* c, const int* a, const int* b, const int len)
4      {
5          int minA = a[0];
6          int minB = b[0];
7          for (int i = 1; i < len; i++)
8          {
9              #pragma HLS PIPELINE
10             if (minA > a[i])
11             {
12                 minA = a[i];
13                 c[i] = minA;
14             }
15             else
16             {
17                 c[i] = 0;
18             }
19         }
20         for (int i = 1; i < len; i++)
21         {
22             #pragma HLS UNROLL factor=5
23             if (minB > b[i])
24             {
25                 minB = b[i];
26                 c[i] = minB;
27             }
28         }
29     }
30 }
```

2 Сборка и отладка проекта в режиме программной эмуляции (Emulation-SW)

Листинг 2.1 — Результат работы программы

```
Found Platform
Platform Name: Xilinx
INFO: Reading /iu_home/iu7123/workspace_5/lab_5_system/Emulation-SW/binary_container_1.xclbin
Loading: '/iu_home/iu7123/workspace_5/lab_5_system/Emulation-SW/binary_container_1.xclbin'
Trying to program device[0]: xilinx_u200_xdma_201830_2
Device[0]: program successful!

|-----+-----|
| Kernel                | Wall-Clock Time (ns) |
|-----+-----|
| var003_no_pragmas      |          7936506 |
|-----+-----|
| var001_unrolled        |          6522734 |
|-----+-----|
| var003_pipelined       |          3496275 |
|-----+-----|
| var003_pipe_unroll     |          6079168 |
|-----+-----|

Note: Wall Clock Time is meaningful for real hardware execution only, not for emulation.
Please refer to profile summary for kernel execution time for hardware emulation.
TEST PASSED.
```

3 Сборка и отладка проекта в режиме аппаратной эмуляции (Emulation-HW)

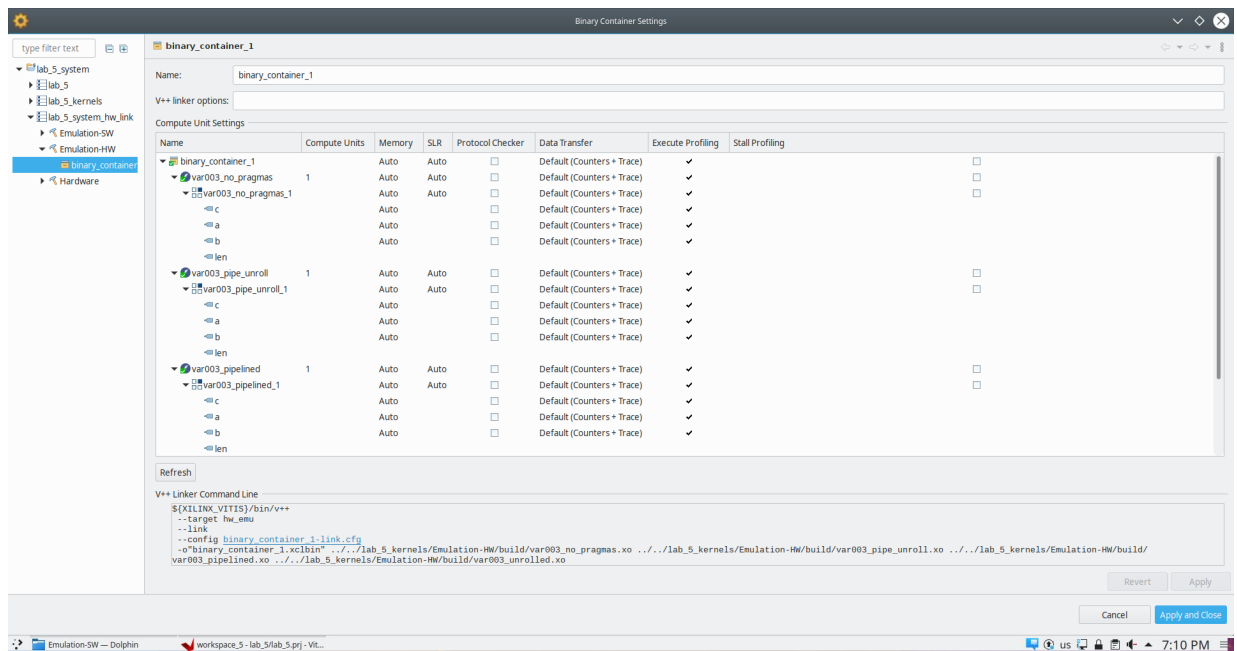


Рисунок 3.1 — Assistant View

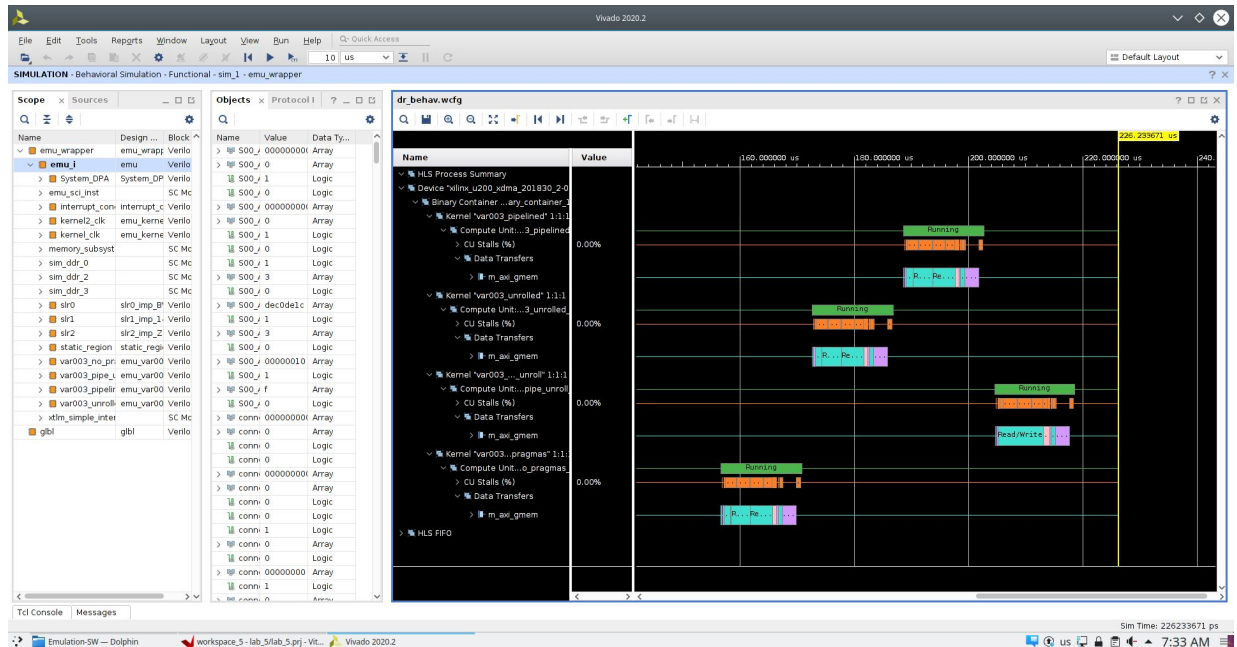


Рисунок 3.2 — Окно внутрисхемного отладчика Vivado

4 Сборка и отладка проекта в режиме аппаратного исполнения (Hardware)

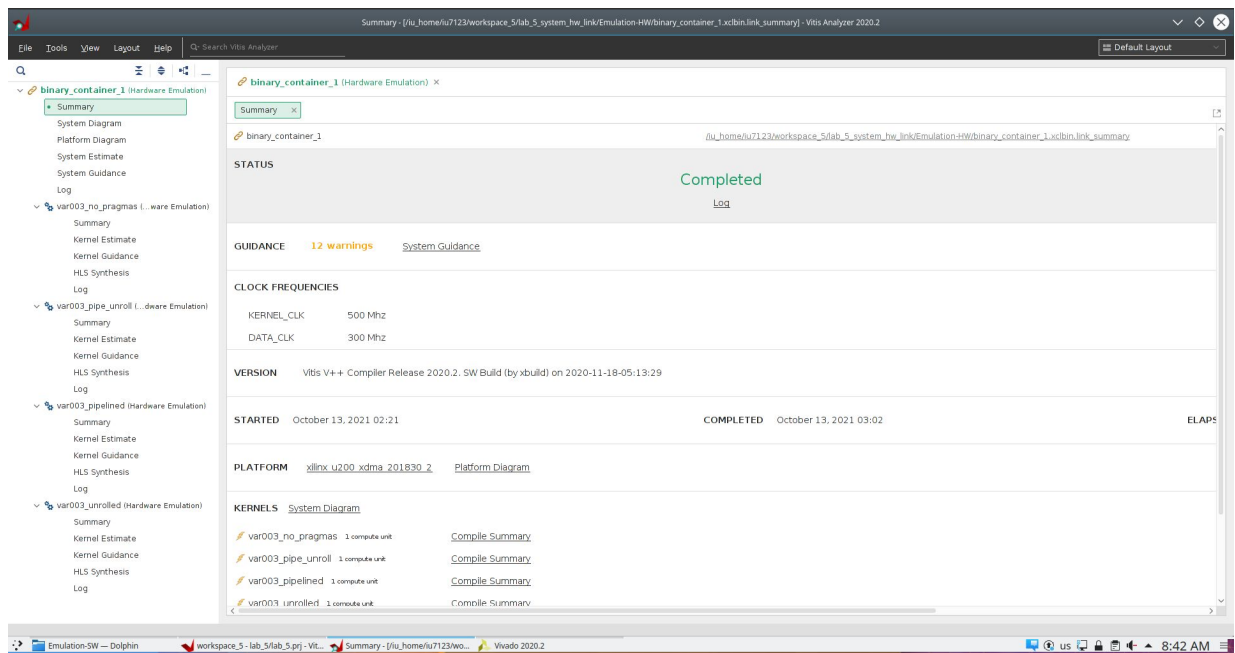


Рисунок 4.1 — Содержимое вкладки Summary

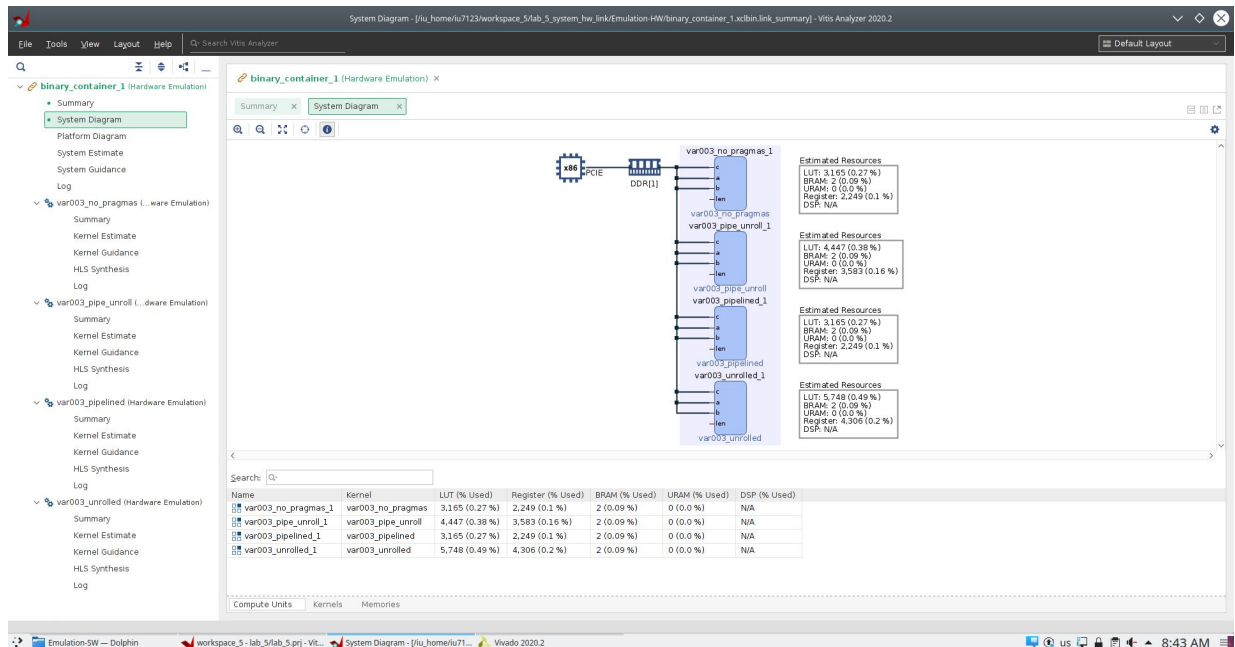


Рисунок 4.2 — Содержимое вкладки System Diagram

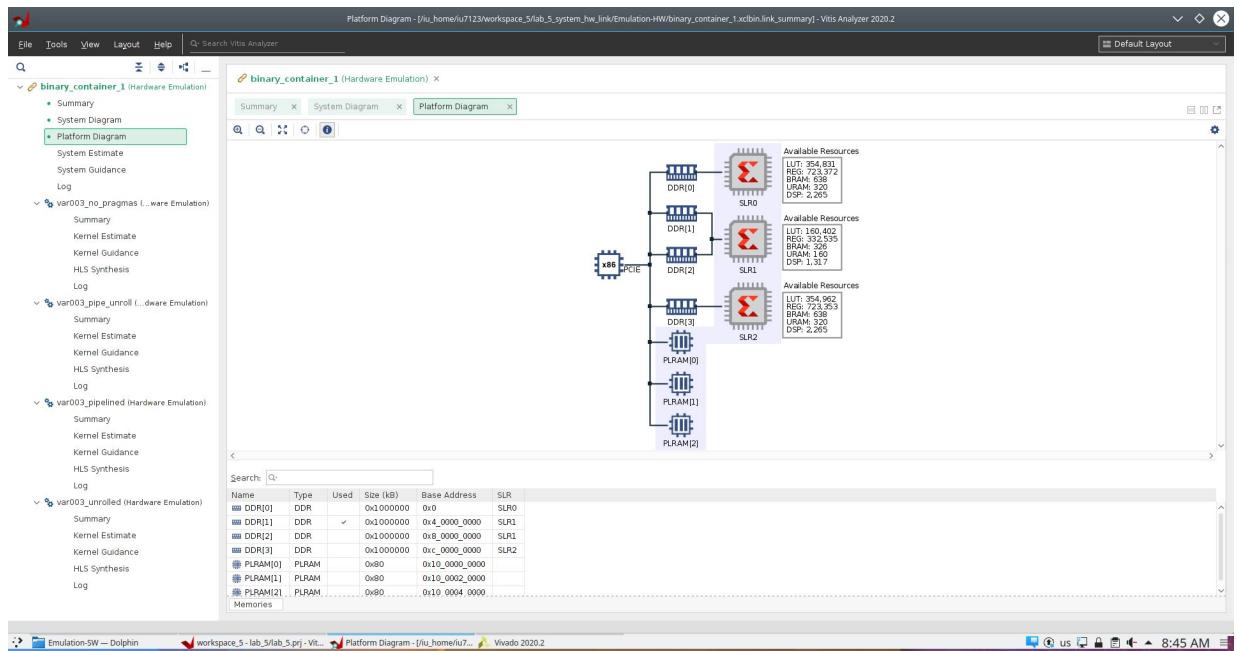


Рисунок 4.3 — Содержимое вкладки Platform Diagram

DATE: Wed Oct 13 02:14:35 2021 VERSION: 2020.2 (Build 3064766 on Wed Nov 18 09:12:47 MST 2020) PROJECT: var003_no_pragmas SOLUTION: solution (Vitis Kernel Flow Target)

| Name | Issue Type | Latency (cycles) | Latency (ns) | Iteration Latency | Interval | Trip Count | Pipelined | BRAM | BRAM (%) | DSP | DSP (%) | FF | FF (%) | LUT | LUT (%) | Slack |
|-------------------|------------|------------------|--------------|-------------------|----------|------------|-----------|------|----------|-----|---------|------|--------|------|---------|-------|
| var003_no_pragmas | | | | | 0 | | no | 2 | ~0 | 0 | 0 | 2249 | ~0 | 3165 | ~0 | 0.00 |
| VITIS_LOOP_7_1 | | | | 72 | 1 | | yes | | | | | | | | | |
| VITIS_LOOP_19_2 | | | | 73 | 1 | | yes | | | | | | | | | |

Рисунок 4.4 — HLS (1)

DATE: Wed Oct 13 02:14:41 2021 VERSION: 2020.2 (Build 3064766 on Wed Nov 18 09:12:47 MST 2020) PROJECT: var003_pipe_unroll SOLUTION: solution (Vitis Kernel Flow Target)

| Name | Issue Type | Latency (cycles) | Latency (ns) | Iteration Latency | Interval | Trip Count | Pipelined | BRAM | BRAM (%) | DSP | DSP (%) | FF | FF (%) | LUT | LUT (%) | Slack |
|--------------------|--------------|------------------|--------------|-------------------|----------|------------|-----------|------|----------|-----|---------|------|--------|------|---------|-------|
| var003_pipe_unroll | II Violation | | | | 0 | | no | 2 | ~0 | 0 | 0 | 3583 | ~0 | 4447 | ~0 | 0.00 |
| VITIS_LOOP_7_1 | | | | 72 | 1 | | yes | | | | | | | | | |
| VITIS_LOOP_20_2 | II Violation | | | 77 | 5 | | yes | | | | | | | | | |

Рисунок 4.5 — HLS (2)

DATE: Wed Oct 13 02:14:37 2021 VERSION: 2020.2 (Build 3064766 on Wed Nov 18 09:12:47 MST 2020) PROJECT: var003_pipelined SOLUTION: solution (Vitis Kernel Flow Target) PRODU

| Name | Issue Type | Latency (cycles) | Latency (ns) | Iteration Latency | Interval | Trip Count | Pipelined | BRAM | BRAM (%) | DSP | DSP (%) | FF | FF (%) | LUT | LUT (%) | Slack |
|------------------|------------|------------------|--------------|-------------------|----------|------------|-----------|------|----------|-----|---------|------|--------|------|---------|-------|
| var003_pipelined | | | | | 0 | | no | 2 | ~0 | 0 | 0 | 2249 | ~0 | 3165 | ~0 | 0.00 |
| VITIS_LOOP_7_1 | | | | 72 | 1 | | yes | | | | | | | | | |
| VITIS_LOOP_19_2 | | | | 73 | 1 | | yes | | | | | | | | | |

Рисунок 4.6 — HLS (3)

DATE: Wed Oct 13 02:14:50 2021 VERSION: 2020.2 (Build 3064766 on Wed Nov 18 09:12:47 MST 2020) PROJECT: var003_unrolled SOLUTION: solution (Vitis Kernel Flow Target) PRODU

| Name | Issue Type | Latency (cycles) | Latency (ns) | Iteration Latency | Interval | Trip Count | Pipelined | BRAM | BRAM (%) | DSP | DSP (%) | FF | FF (%) | LUT | LUT (%) | Slack |
|-----------------|--------------|------------------|--------------|-------------------|----------|------------|-----------|------|----------|-----|---------|------|--------|------|---------|-------|
| var003_unrolled | II Violation | | | | 0 | | no | 2 | ~0 | 0 | 0 | 4306 | ~0 | 5748 | ~0 | 0.00 |
| VITIS_LOOP_7_1 | II Violation | | | 76 | 5 | | yes | | | | | | | | | |
| VITIS_LOOP_20_2 | II Violation | | | 77 | 5 | | yes | | | | | | | | | |

Рисунок 4.7 — HLS (4)

Листинг 4.1 — Результаты работы

```
Found Platform
Platform Name: Xilinx
INFO: Reading /iu_home/iu7123/workspace_5/lab_5_system/Hardware/binary_container_1.xclbin
Loading: '/iu_home/iu7123/workspace_5/lab_5_system/Hardware/binary_container_1.xclbin'
Trying to program device[0]: xilinx_u200_xdma_201830_2
Device[0]: program successful!

|-----+-----|
| Kernel                | Wall-Clock Time (ns) |
|-----+-----|
| var003_no_pragmas     |          3077065 |
|-----+-----|
| var001_unrolled       |          1029812 |
|-----+-----|
| var003_pipelined      |          604712 |
|-----+-----|
| var003_pipe_unroll    |          644299 |
|-----+-----|

Note: Wall Clock Time is meaningful for real hardware execution only, not for emulation.
Please refer to profile summary for kernel execution time for hardware emulation.
TEST PASSED.
```

Заключение

В ходе лабораторной работы были изучены архитектура гетерогенных вычислительных систем и технологии разработки ускорителей вычислений на базе ПЛИС фирмы Xilinx. Была выполнена генерация ядра ускорителя с последующим синтезом, сборкой и тестированием бинарного модуля ускорителя.

В результате сборки проекта было выяснено, что использование оптимизаций приводит к реальному повышению быстродействия работы программы. Следует отметить, что в режиме программной эмуляции выигрыш получился небольшим (примерно в 2 раза) по сравнению с ускорением, полученным в режимах аппаратной эмуляции и аппаратного исполнения ускорение достигло порядка 5 раз.