



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» (ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

ОТЧЕТ

по лабораторной работе № 5

Название: Разработка ускорителей вычислений средствами САПР
высокоуровневого синтеза Xilinx Vitis HLS

Дисциплина: Архитектура ЭВМ

Студент

ИУ7-52Б

(Группа)

(Подпись,
дата)

Короткая В. М.

(И.О. Фамилия)

Преподаватель

(Подпись,
дата)

Попов А. Ю.

(И.О. Фамилия)

Москва, 2021

Цель работы

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.

Индивидуальный вариант 10

```
extern "C" {  
  
void var010_no_pragmas(int* c, const int* a, const int* b, const int len) {  
  
    int maxA = a[len-1];  
  
    for (int i = len-1; i >=0 ; i--) {  
  
        if (maxA < a[i]) {  
  
            maxA = a[i];  
  
            c[i] = maxA;  
  
        } else {  
  
            c[i] = b[i];  
  
        }  
  
    }  
  
}  
  
}
```

```
extern "C" {  
  
void var010_unrolled(int* c, const int* a, const int* b, const int len) {  
  
    int maxA = a[len-1];  
  
    for (int i = len-1; i >=0 ; i--) {  
  
        #pragma HLS UNROLL factor=2  
  
        if (maxA < a[i]) {  
  
            maxA = a[i];  
  
            c[i] = maxA;  
  
        } else {  
  
            c[i] = b[i];  
  
        }  
  
    }  
  
}  
  
}
```

Частичное разворачивание цикла

```

extern "C" {

void var010_pipelined(int* c, const int* a, const int* b, const int len) {

    int maxA = a[len-1];

    for (int i = len-1; i >=0 ; i--) {

        #pragma HLS PIPELINE

        if (maxA < a[i]) {

            maxA = a[i];

            c[i] = maxA;

        } else {

            c[i] = b[i];

        }

    }

}

}

```

Конвейерная организация цикла

```

extern "C" {

void var010_pipe_unroll(int* c, const int* a, const int* b, const int len) {

    int maxA = a[len-1];

    for (int i = len-1; i >=0 ; i--) {

        #pragma HLS PIPELINE

        #pragma HLS UNROLL factor = 2

        if (maxA < a[i]) {

            maxA = a[i];

            c[i] = maxA;

        } else {

            c[i] = b[i];

        }

    }

}

}

```

Конвейеризация и частичное разворачивание

Сборка и отладка префта осуществляется в трех режимах: программная эмуляция (Emulation-SW), аппаратная эмуляция (Emulation-HW), аппаратное исполнение (Hardware).

Emulation-SW

Проект был запущен в режиме программной эмуляции. Результаты работы:

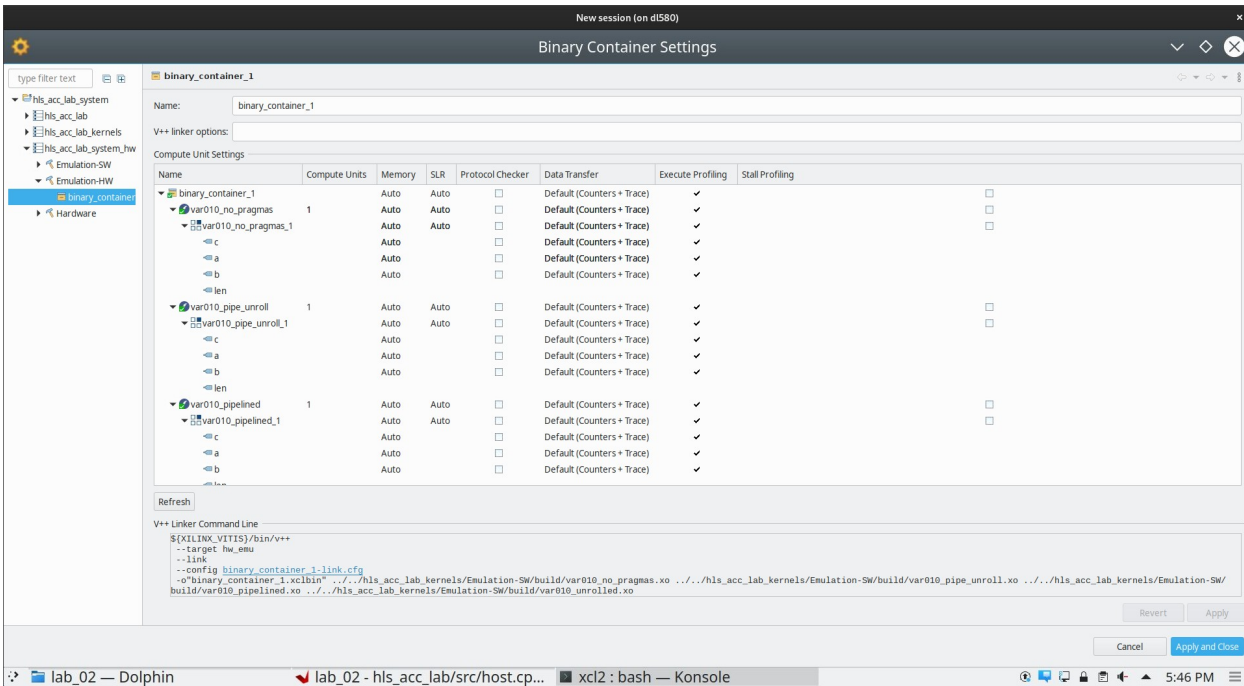
```
[Console output redirected to file:/iu_home/iu7040/lab_02/hls_acc_lab/Emulation-SW/SystemDebugger_hls_acc_lab_system_hls_acc
Found Platform
Platform Name: Xilinx
INFO: Reading /iu_home/iu7040/lab_02/hls_acc_lab_system/Emulation-SW/binary_container_1.xclbin
Loading: '/iu_home/iu7040/lab_02/hls_acc_lab_system/Emulation-SW/binary_container_1.xclbin'
Trying to program device[0]: xilinx_u200_xdma_201830_2
Device[0]: program successful!

+-----+
| Kernel                | Wall-Clock Time (ns) |
+-----+
| var010_no_pragmas     | 2761110               |
+-----+
| var010_unrolled       | 2589096               |
+-----+
| var010_pipelined      | 742786                |
+-----+
| var010_pipe_unroll    | 2235473               |
+-----+

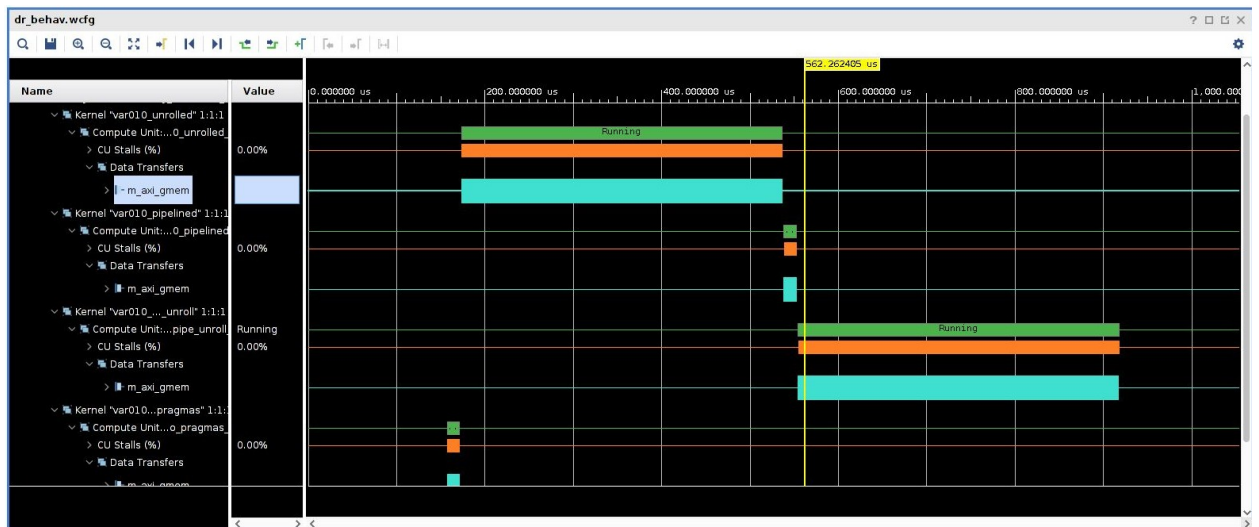
Note: Wall Clock Time is meaningful for real hardware execution only, not for emulation.
Please refer to profile summary for kernel execution time for hardware emulation.
TEST PASSED.
```

Emulation-HW

Проект был собран в режиме аппаратной эмуляции.



Экран Assistant View



Диаграммы работы четырех ядер

Результат работы:

[Console output redirected to
file:/iu_home/iu7040/lab_02/hls_acc_lab/Emulation-HW/SystemDebugger_hls_acc_lab_system_hls_acc_lab.launch.log]

Found Platform

Platform Name: Xilinx

INFO: Reading /iu_home/iu7040/lab_02/hls_acc_lab_system/Emulation-HW/binary_container_1.xclbin

Loading: '/iu_home/iu7040/lab_02/hls_acc_lab_system/Emulation-HW/binary_container_1.xclbin'

Trying to program device[0]: xilinx_u200_xdma_201830_2

INFO: [HW-EMU 01] Hardware emulation runs simulation underneath. Using a large data set will result in long simulation times. It is recommended that a small dataset is used for faster execution. The flow uses approximate models for DDR memory and interconnect and hence the performance data generated is approximate.

INFO::[Vitis-EM 22] [Time elapsed: 2 minute(s) 52 seconds, Emulation time: 0.0640411 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB

Device[0]: program successful!

Kernel	Wall-Clock Time (ns)
var010_no_pragmas	49036832061

INFO::[Vitis-EM 22] [Time elapsed: 7 minute(s) 42 seconds, Emulation time: 0.181118 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
---------------------------------------	---------------	---------------

var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 0.160 KB	WR = 0.082 KB

INFO::[Vitis-EM 22] [Time elapsed: 12 minute(s) 42 seconds, Emulation time: 0.30822 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 2.957 KB	WR = 1.480 KB

INFO::[Vitis-EM 22] [Time elapsed: 17 minute(s) 45 seconds, Emulation time: 0.426477 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 0.000 KB	WR = 0.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 5.566 KB	WR = 2.793 KB

| var010_unrolled | 880462848761 |

|-----+-----|

| var010_pipelined | 43026561696 |

|-----+-----|

INFO::[Vitis-EM 22] [Time elapsed: 22 minute(s) 48 seconds, Emulation time: 0.555709 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 0.020 KB	WR = 0.004 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB

INFO::[Vitis-EM 22] [Time elapsed: 27 minute(s) 50 seconds, Emulation time: 0.685019 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 2.855 KB	WR = 1.434 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB

INFO::[Vitis-EM 22] [Time elapsed: 32 minute(s) 52 seconds, Emulation time: 0.819856 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 5.840 KB	WR = 2.926 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB

| var010_pipe_unroll | 832430425879 |

|-----+-----|

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.

INFO::[Vitis-EM 22] [Time elapsed: 38 minute(s) 16 seconds, Emulation time: 0.967478 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB

INFO::[Vitis-EM 22] [Time elapsed: 41 minute(s) 15 seconds, Emulation time: 1.05272 ms]

Data transfer between kernel(s) and global memory(s)

var010_no_pragmas_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipe_unroll_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_pipelined_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB
var010_unrolled_1:m_axi_gmem-DDR[1]	RD = 7.980 KB	WR = 4.000 KB

INFO: [HW-EMU 06-0] Waiting for the simulator process to exit

Здесь время работы каждой функции отличается от предыдущей сборки, но рассматривать его как истинный результат нельзя. Такое время работы обусловлено тем, что при выполнении самих функций параллельно также производилось моделирование ядер в Vivado, что замедляло работу программы, так как для каждого такта необходимо производить вычисления и отображать результаты на диаграмме.

Hardware

STATUS

Completed

Log

GUIDANCE

8 warnings

System Guidance

CLOCK FREQUENCIES

KERNEL_CLK

500 Mhz

DATA_CLK

300 Mhz

VERSION

Vitis V++ Compiler Release 2020.2, SW Build (by xbuild) on 2020-11-18-05:13:29

STARTED

October 03, 2021 17:02

COMPLETED

October 03, 2021 17:30

PLATFORM

xilinx_u200_xdma_201830_2

Platform Diagram

KERNELS

System Diagram

var010_no_pragmas

1 compute unit

Compile Summary

var010_pipe_unroll

1 compute unit

Compile Summary

var010_pipelined

1 compute unit

Compile Summary

var010_unrolled

1 compute unit

Compile Summary

Общая информация о сборке (Summary)

Estimated Resources

LUT: 2,116 (0.18 %)
BRAM: 2 (0.09 %)
URAM: 0 (0.0 %)
Register: 1,881 (0.09 %)
DSP: N/A

Estimated Resources

LUT: 3,688 (0.31 %)
BRAM: 2 (0.09 %)
URAM: 0 (0.0 %)
Register: 2,338 (0.11 %)
DSP: N/A

Estimated Resources

LUT: 2,116 (0.18 %)
BRAM: 2 (0.09 %)
URAM: 0 (0.0 %)
Register: 1,881 (0.09 %)
DSP: N/A

Estimated Resources

LUT: 3,688 (0.31 %)
BRAM: 2 (0.09 %)
URAM: 0 (0.0 %)
Register: 2,338 (0.11 %)
DSP: N/A

Search: Q-

Name	Kernel	LUT (% Used)	Register (% Used)	BRAM (% Used)	URAM (% Used)	DSP (% Used)
var010_no_pragmas_1	var010_no_pragmas	2,116 (0.18 %)	1,881 (0.09 %)	2 (0.09 %)	0 (0.0 %)	N/A
var010_pipe_unroll_1	var010_pipe_unroll	3,688 (0.31 %)	2,338 (0.11 %)	2 (0.09 %)	0 (0.0 %)	N/A
var010_pipelined_1	var010_pipelined	2,116 (0.18 %)	1,881 (0.09 %)	2 (0.09 %)	0 (0.0 %)	N/A
var010_unrolled_1	var010_unrolled	3,688 (0.31 %)	2,338 (0.11 %)	2 (0.09 %)	0 (0.0 %)	N/A

Compute Units

Kernels

Memories

System Diagram

Available Resources

LUT: 354,831
REG: 723,372
BRAM: 638
URAM: 320
DSP: 2,265

Available Resources

LUT: 160,402
REG: 392,595
BRAM: 326
URAM: 160
DSP: 1,317

Available Resources

LUT: 354,962
REG: 723,353
BRAM: 638
URAM: 320
DSP: 2,265

Search: Q-

Name	Type	Used	Size (kB)	Base Address	SLR
DDR[0]	DDR		0x1000000	0x0	SLR0
DDR[1]	DDR	✓	0x1000000	0x4_0000_0000	SLR1
DDR[2]	DDR		0x1000000	0x8_0000_0000	SLR1
DDR[3]	DDR		0x1000000	0xc_0000_0000	SLR2
PLRAM[0]	PLRAM		0x80	0x10_0000_0000	
PLRAM[1]	PLRAM		0x80	0x10_0002_0000	
PLRAM[2]	PLRAM		0x80	0x10_0004_0000	

Platform Diagram

Name	Issue Type	Latency (cycles)	Latency (ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	BRAM (%)	DSP	DSP (%)	FF	FF (%)	LUT	LUT (%)	Slack
var010_no_pragmas	II Violation						no	2	~0	0	0	1881	~0	2116	~0	0.00
VTIS_LOOP_4_1	II Violation			214	2		yes									

Информация о ядре без оптимизации цикла

Name	Issue Type	Latency (cycles)	Latency (ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	BRAM (%)	DSP	DSP (%)	FF	FF (%)	LUT	LUT (%)	Slack
var010_pipe_unroll	II Violation						no	2	~0	0	0	2338	~0	3688	~0	0.00
VTIS_LOOP_4_1	II Violation			427	212		yes									

Информация о ядре с конвейерным частично развернутым циклом

Name	Issue Type	Latency (cycles)	Latency (ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	BRAM (%)	DSP	DSP (%)	FF	FF (%)	LUT	LUT (%)	Slack
var010_pipelined	II Violation						no	2	~0	0	0	2338	~0	3688	~0	0.00
VTIS_LOOP_4_1	II Violation			214	2		yes									

Информация о ядре с конвейерным циклом

Name	Issue Type	Latency (cycles)	Latency (ns)	Iteration Latency	Interval	Trip Count	Pipelined	BRAM	BRAM (%)	DSP	DSP (%)	FF	FF (%)	LUT	LUT (%)	Slack
var010_unrolled	II Violation						no	2	~0	0	0	2338	~0	3688	~0	0.00
VTIS_LOOP_4_1	II Violation			427	212		yes									

Информация о ядре с частично развернутым циклом

Результат работы:

```
INFO: Reading /iu_home/iu7040/lab_02/hls_acc_lab_system/Hardware/binary_container_1.xclbin
Loading: '/iu_home/iu7040/lab_02/hls_acc_lab_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)
var010_no_pragmas	134227035
var010_unrolled	436261311
var010_pipelined	132388299
var010_pipe_unroll	436357874

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.

По результатам работы можно судить, что конвейеризация цикла дает совсем незначительный выигрыш времени по сравнению с неоптимизированным циклом. Это происходит, потому что внутри цикла присутствует зависимость по данным. В частности, в каждой итерации по несколько раз используется переменная *i* — счетчик цикла. Для следующей итерации этот счетчик необходимо инкрементировать, но чтобы это сделать, нужно ждать, пока предыдущая итерация не будет в нем нуждаться. Таким образом, пока не будет выполняться последний оператор в теле цикла, другой поток не сможет начать обработку следующей итерации. Разворачивание цикла же вообще замедляет выполнение задачи примерно в 4 раза. Причиной этому также является зависимость по данным — развернутые итерации цикла должны вычисляться параллельно, но из-за того, что в теле цикла присутствует переменная *ptr*, требуется, чтобы сначала в первой итерации развернутого цикла произошло присваивание и инкремент этой переменной, а только

затем во второй. При этом возникают дополнительные расходы времени на проверку выхода за границы массива и диспетчеризацию параллельных потоков.

Вывод

В результате выполнения лабораторной работы были смоделированы четыре ядра с разными уровнями оптимизаций, изучен синтез ускорителей на ЯВУ на примере C/C++, разработан и протестирован ускоритель вычислений. В результате тестирования и замеров времени работы было установлено, что конвейеризация дает минимальный выигрыш во времени, а разворачивание цикла не позволяет оптимизировать задачу, соответствующую индивидуальному варианту, вследствие зависимостей по данным.