

Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Кафедра компьютерных систем и программных технологий

Отчет по лабораторной работе №11_3
Курс: «Проектирование реконфигурируемых гибридных
вычислительных систем»
Тема: « Задержка (Latency) »

Выполнил студент гр. 3540901/81501

Селиверстов Я.А.

(подпись)

Руководитель

Антонов А.П.

(подпись)

“ ____ ” _____ 2019 г.

Санкт – Петербург
2019

ОГЛАВЛЕНИЕ

1. Задание.....	3
2. Исходный код.....	6
3. Моделирование	9
4. Исследование	10
4.1. Решение 1а	10
4.1.2.Синтез решения 1а	10
4.2. Решение 2а	13
4.2.2.Синтез решения 2а	14
4.3. Решение 3а	16
4.3.2.Синтез решения 3а	16
4.4. Решение 4а	18
4.4.2.Синтез решения 4а	19
9. Вывод.....	21

1. Задание

- Создать проект lab11_3
- Микросхема: ха7a12tcsg325-1q
- Познакомиться с исходным кодом функции loop_imperfect.c
- Познакомиться с исходным кодом теста loop_imperfect_test.c для проверки функции. Осуществить моделирование (с выводом результатов в консоль)
- Исследование:
- Solution_1a
 - задать: clock period 10; clock_uncertainty 0.1
 - установить реализацию ПО УМОЛЧАНИЮ
 - осуществить синтез для:
 - привести в отчете:
 - performance estimates=>summary (timing, latency)
 - utilization estimates=>summary
 - performance Profile
 - Resource profile
 - scheduler viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
 - resource viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
 - Выполнить cosimulation и привести временную диаграмму
- Solution_2a
 - задать: clock period 10; clock_uncertainty 0.1
 - преобразовать функцию из формы Unperfect в форму Perfect
 - установить реализацию ПО УМОЛЧАНИЮ
 - осуществить синтез
 - привести в отчете:
 - performance estimates=>summary (timing, latency)
 - utilization estimates=>summary
 - performance Profile
 - Resource profile
 - scheduler viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval

- resource viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
 - Выполнить cosimulation и привести временную диаграмму
- Сравнить два решения (solution_1a и solution_2a) и сделать выводы; объяснить (посчитать) число циклов Latency, П...
- Solution_3a
 - задать: clock period 10; clock_uncertainty 0.1
 - использовать функцию преобразованную в форму Perfect
 - установить реализацию Flatten для внутреннего цикла
 - осуществить синтез
 - привести в отчете:
 - performance estimates=>summary (timing, latency)
 - utilization estimates=>summary
 - performance Profile
 - Resource profile
 - scheduler viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
 - resource viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
 - Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
- Сравнить два решения (solution_2a и solution_3a) и сделать выводы; объяснить (посчитать) число циклов Latency, П...
- Solution_4a
 - задать: clock period 10; clock_uncertainty 0.1
 - использовать функцию, преобразованную в форму Perfect
 - установить реализацию UNROLLED для внутреннего цикла
 - осуществить синтез
 - привести в отчете:
 - performance estimates=>summary (timing, latency)
 - utilization estimates=>summary

- performance Profile
- Resource profile
- scheduler viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
- resource viewer (выполнить Zoom to Fit)
 - На скриншоте показать Latency
 - На скриншоте показать Initiation Interval
- Выполнить cosimulation и привести временную диаграмму (интерес представляет количество и тип портов)
- Сравнить два решения (solution_3a и solution_4a) и сделать выводы; объяснить (посчитать) число циклов Latency, П...

2. Исходный код

Зададим следующий код устройства:

```
#include "loop_imperfect.h"

void loop_imperfect(din_t A[N], dout_t B[N]) {

    int i, j;
    dint_t acc;

    LOOP_I: for (i=0; i < 20; i++){
        acc = 0;
        LOOP_J: for (j=0; j < 20; j++){
            acc += A[j] * j;
        }
        if (i%2 == 0)
            B[i] = acc / 20;
        else
            B[i] = 0;
    }
}
```

Код устройства в форме PERFECT имеет вид

```
#include "loop_imperfect.h"

void loop_imperfect(din_t A[N], dout_t B[N]) {

    int i, j;
    dint_t acc;

    LOOP_I: for (i=0; i < 20; i++){
        LOOP_J: for (j=0; j < 20; j++){
            if (j == 0)
            {
                acc = 0;
            }
            acc += A[j] * j;
            if (j == 19)
            {
                if (i%2 == 0)
                    B[i] = acc / 20;
                else
                    B[i] = 0;
            }
        }
    }
}
```

Заголовочный файл определим как:

```
#ifndef _LOOP_IMPERFECT_H_
#define _LOOP_IMPERFECT_H_

#include "ap_cint.h"
#define N 20

typedef int5 din_t;
typedef int12 dint_t;
typedef int6 dout_t;

void loop_imperfect(din_t A[N], dout_t B[N]);

#endif
```

Исходный код теста имеет вид:

```
#include <stdio.h>
#include "loop_imperfect.h"

int main() {
    din_t A[N];
    dout_t B[N];

    int i, retval = 0;
    FILE *fp;

    for (i = 0; i < N; ++i) {
        A[i] = i;
    }
    // Save the results to a file
    fp = fopen("result.dat", "w");

    // Call the function
    loop_imperfect(A, B);
    for (i = 0; i < N; ++i) {
        fprintf(fp, "%d\n", B[i]);
    }
    fclose(fp);

    // Compare the results file with the golden results
    retval = system("diff --brief -w result.dat result.golden.dat");
    if (retval != 0) {
        printf("Test_failed_!!!\n");
        retval = 1;
    } else {
        printf("Test_passed_!\n");
    }

    // Return 0 if the test passed
    return retval;
}
```

Скрипт для запуска программы с консоли имеет вид:

```
open_project -reset lab11_3_imperfect

add_files loop_imperfect.c
add_files -tb loop_imperfect_test.c
add_files -tb result.golden.dat
set_top loop_imperfect

open_solution solution_1a -reset
set_part {xa7a12tcsg325-1q}
create_clock -period 10ns
set_clock_uncertainty 0.1

csim_design
csynth_design
# cosim_design -trace_level all

open_project -reset lab11_3_perfect

add_files loop_perfect.c
add_files -tb loop_imperfect_test.c
add_files -tb result.golden.dat
set_top loop_imperfect

set solutions [list 2a 3a 4a]

foreach sol $solutions {
    open_solution solution_$sol -reset
    set_part {xa7a12tcsg325-1q}
    create_clock -period 10ns
    set_clock_uncertainty 0.1

    if {$sol == "3a"} {
        set_directive_loop_flatten "loop_imperfect/LOOP_J"
    }
    if {$sol == "4a"} {
        set_directive_unroll "loop_imperfect/LOOP_J"
    }

    csim_design
    csynth_design
    # cosim_design -trace_level all
}

exit
```


3. Моделирование

Результаты моделирования, подтверждающие корректность работы устройства, имеют вид:

```
INFO: [APCC 202-3] Tmp directory is /tmp/apcc_db_sobol/1573141575889476931815
INFO: [APCC 202-1] APCC is done.
    Generating csim.exe
Test passed !
INFO: [SIM 211-1] CSim done with 0 errors.
INFO: [SIM 211-3] ***** CSIM finish *****
```

4. Исследование

4.1. Решение 1a

В соответствие с планом лабораторной работы устанавливаем:

- clock period = 10;
- clock uncertainty = 0.1;
- реализация ПО УМОЛЧАНИЮ

Директивы данного решения имеют вид:

▼ ● loop_imperfect
 ● A
 ● B
 ▼ $\frac{x+y}{=?}$ LOOP_I
 $\frac{x+y}{=?}$ LOOP_J

4.1.2.Синтез решения 1a

Результаты оценки производительности имеют вид:

Performance Estimates

▣ Timing (ns)

▣ Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	9.332	0.10

▣ Latency (clock cycles)

▣ Summary

Latency		Interval		
min	max	min	max	Type
861	861	861	861	none

По данным результатам синтеза можно сделать вывод, что оценка производительности видно устройства соответствует заданным критериям.

Оценка использования имеет вид:

Utilization Estimates				
Summary				
Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	-	-	-
Expression	-	3	0	47
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	42
Register	-	-	84	-
Total	0	3	84	89
Available	40	4016000	8000	
Utilization (%)	0	7	~0	1

Профиль производительности имеет вид:

Performance Profile		Resource Profile			
	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip count
loop_imperfect	-	861	-	862	-
LOOP_I	no	860	43	-	20
LOOP_J	no	40	2	-	20

Данные планировщика просмотра имеет вид:



Данные обзора ресурсов имеет вид:

	Resource\Control Step	C0	C1	C2	C3	C4
1	I/O Ports					
2	B(p0)			write		write
3	A(p0)			read		
4	Memory Ports					
5	B(p0)			write		write
6	A(p0)			read		
7	Expressions					
8	i_l_fu_135		+			
9	i_phi_fu_93		phi_mux			
10	exitcond1_fu_129		icmp			
11	j_l_fu_147			+		
12	acc_phi_fu_117			phi_mux		
13	j_phi_fu_105			phi_mux		
14	mul_fu_253			*		
15	exitcond_fu_141			icmp		
16	grp_fu_261				+	
17	neg_mul_fu_187					-
18	neg_ti_fu_229					-
19	p_v_fu_217					select
20	tmp_s_fu_239					select

Задержка (внутреннего цикла Loop j) = $20 \times 2 = 40$ тактов.

Для внешнего цикла значение задержка составляет: $\text{LatencyI} = \text{LatencyJ} + 1$ (общий подготовительный такт) + 1(подготовительный такт для цикла J) + 1 (завершающий такт) = $40 + 1 + 1 + 1 = 43$ такта.

С учетом того, что число итераций 20, то

$\text{Latency} = 20 \times \text{LatencyI} + 1$ (подготовительный такт) = 861

4.2. Решение 2а

В соответствие с планом лабораторной работы устанавливаем:

- clock period 10;
- clock_uncertainty 0.1
- преобразовать функцию из формы Unperfect в форму Perfect
- установить реализацию ПО УМОЛЧАНИЮ

Директивы данного решения имеют вид:

▼ ● loop_imperfect

● A

● B

▼ $\frac{x+y}{=?}$ LOOP_I

$\frac{x+y}{=?}$ LOOP_J

4.2.2.Синтез решения 2а

Результаты оценки производительности имеют вид:

Performance Estimates

[-] Timing (ns)

[-] Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	9.332	0.10

[-] Latency (clock cycles)

[-] Summary

Latency		Interval		
min	max	min	max	Type
1241	12041	1241	2041	none

Оценка использования имеет вид:

Utilization Estimates

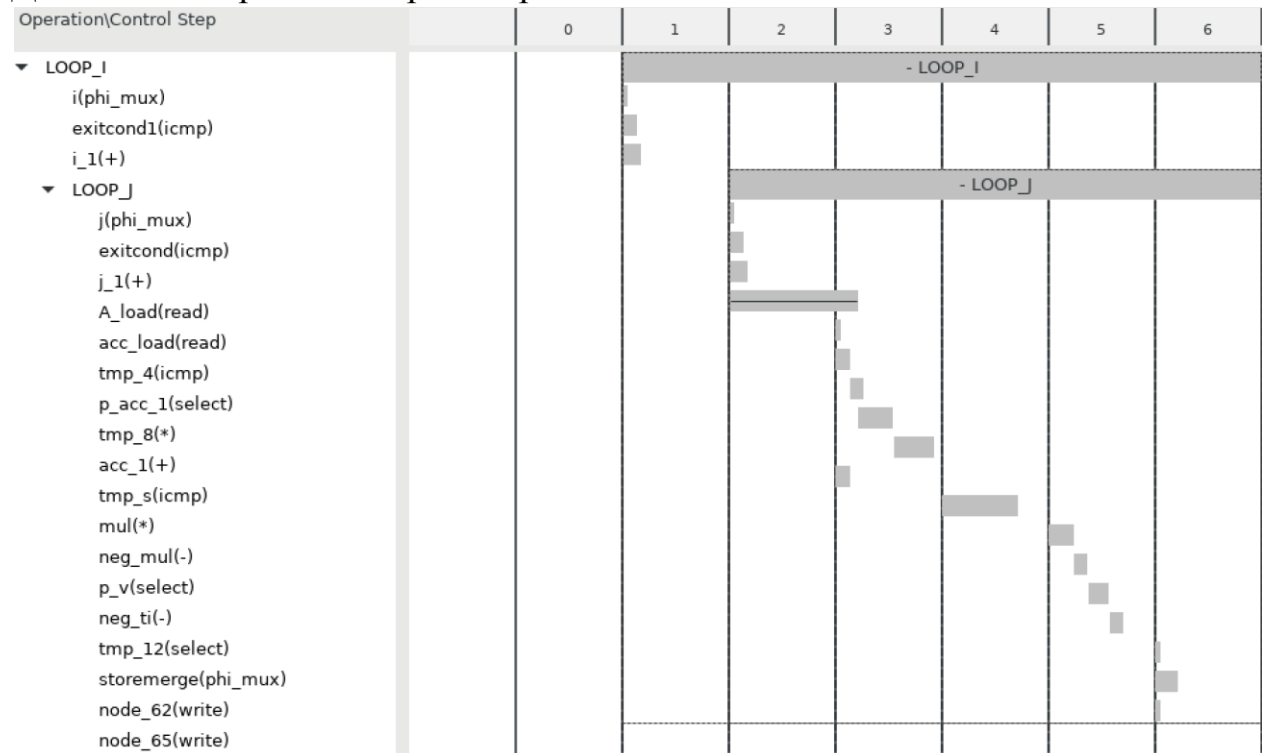
[-] Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	2	-	-
Expression	-	-	0	152
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	68
Register	-	-	98	-
Total	0	2	98	220
Available	40	40	16000	8000
Utilization (%)	0	5	~0	2

Профиль производительности имеет вид:

Performance Profile		Resource Profile			
	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip c
▼ ● loop_imperfect	-	1241~2041	-	1242 ~ 2042	-
▼ ● LOOP_I	no	1240 ~ 2040	62 ~ 102	-	20
● LOOP_J	no	60 ~ 100	3 ~ 5	-	20

Данные планировщика просмотра имеет вид:



Данные обзора ресурсов имеет вид:

	Resource\Control Step	C0	C1	C2	C3	C4	C5	C6
1	I/O Ports							
2	A(p0)			read				
3	B(p0)							write
4	Memory Ports							
5	A(p0)			read				
6	B(p0)							write
7	Expressions							
8	i_phi_fu_91		phi_mux					
9	i_1_fu_132		+					
10	exitcond1_fu_126		icmp					
11	j_phi_fu_102			phi_mux				
12	j_1_fu_149			+				
13	exitcond_fu_143			icmp				
14	p_acc_1_fu_169				select			
15	grp_fu_269				+			
16	tmp_4_fu_163				icmp			
17	tmp_s_fu_185				icmp			
18	mul_fu_277					*		
19	neg_mul_fu_206						-	
20	neg_ti_fu_247						-	
21	tmp_12_fu_257						select	
22	p_v_fu_235						select	
23	storemerge_phi_fu_114							phi_mux

После преобразования в форму PERFECT, производительность ухудшилась по сравнению с Решением 1а.

4.3. Решение 3а

В соответствие с планом лабораторной работы устанавливаем:

- clock period 10;
- clock uncertainty 0.1
- использовать функцию, преобразованную в форму Perfect
- установить реализацию Flattend для внутреннего цикла

Директивы данного решения имеют вид:

```
▼ ● loop_imperfect
    ● A
    ● B
    ▼  $\frac{x+y}{=?}$  LOOP_I
        ▼  $\frac{x+y}{=?}$  LOOP_J
            % HLS LOOP_FLATTEN
```

4.3.2. Синтез решения 3а

Результаты оценки производительности имеют вид:

Performance Estimates

▣ Timing (ns)

▣ Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	9.332	0.10

▣ Latency (clock cycles)

▣ Summary

Latency		Interval		
min	max	min	max	Type
1201	2001	1201	2001	none

Оценка использования имеет вид:

Utilization Estimates

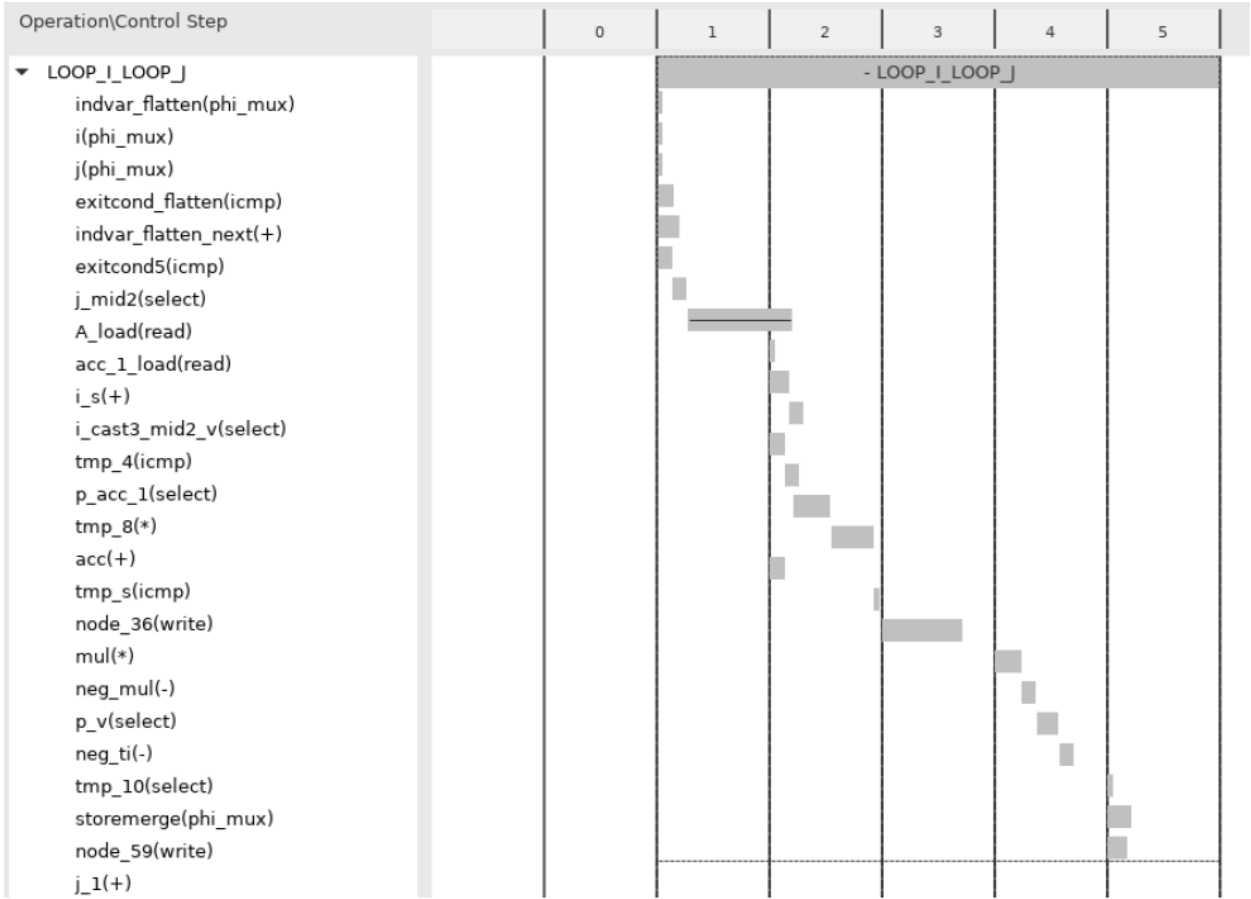
▣ Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	2	-	-
Expression	-	-	0	180
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	74
Register	-	-	116	-
Total	0	2	116	254
Available	40	40	16000	8000
Utilization (%)	0	5	~0	3

Профиль производительности имеет вид:

Performance Profile		Resource Profile			
	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip
▼ ● loop_imperfect	-	1201~2001	-	1202 ~ 2002	-
● LOOP_I_LOOP_J	no	1200 ~ 2000	3 ~ 5	-	400

Данные планировщика просмотра имеет вид:



Данные обзора ресурсов имеет вид:

	Resource\Control Step	C0	C1	C2	C3	C4	C5
1	I/O Ports						
2	A(p0)		read				
3	B(p0)						write
4	Memory Ports						
5	A(p0)		read				
6	B(p0)						write
7	Expressions						
8	indvar_flatten_next_fu_138		+				
9	indvar_flatten_phi_fu_90		phi_mux				
10	j_phi_fu_113		phi_mux				
11	i_phi_fu_101		phi_mux				
12	j_mid2_fu_150		select				
13	exitcond5_fu_144		icmp				
14	exitcond_flatten_fu_132		icmp				
15	i_s_fu_166			+			
16	i_cast3_mid2_v_fu_172			select			
17	p_acc_1_fu_192			select			
18	grp_fu_293			+			
19	tmp_s_fu_207			icmp			
20	tmp_4_fu_187			icmp			
21	mul_fu_302				*		
22	neg_ti_fu_271					-	
23	neg_mul_fu_238					-	
24	tmp_10_fu_281					select	
25	p_v_fu_260					select	
26	j_1_fu_288						+
27	storemerge_phi_fu_124						phi_mux

По результатам синтеза можно сделать вывод, что два вложенных цикла преобразовались в один, число итераций 400, вместе с тем длительность одной итерации составляет от 3 до 5 тактов, таким образом, данное решение менее производительнее, чем Решение 1a

4.4. Решение 4a

В соответствие с планом лабораторной работы устанавливаем:

- clock period 10;
- clock_uncertainty 0.1
- использовать функцию, преобразованную в форму Perfect
- установить реализацию UNROLLED для внутреннего цикла

Директивы данного решения имеют вид:

```

▼ ● loop_imperfect
    ● A
    ● B
    ▼  $x+y=?$  LOOP_I
        ▼  $x+y=?$  LOOP_J
            % HLS UNROLL

```

4.4.2. Синтез решения 4а

Результаты оценки производительности имеют вид:

Performance Estimates

▣ Timing (ns)

▣ Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	11.000	0.10

▣ Latency (clock cycles)

▣ Summary

Latency		Interval		
min	max	min	max	Type
261	301	261	301	none

По оценке производительности видно, что устройство соответствует НЕ заданным критериям.

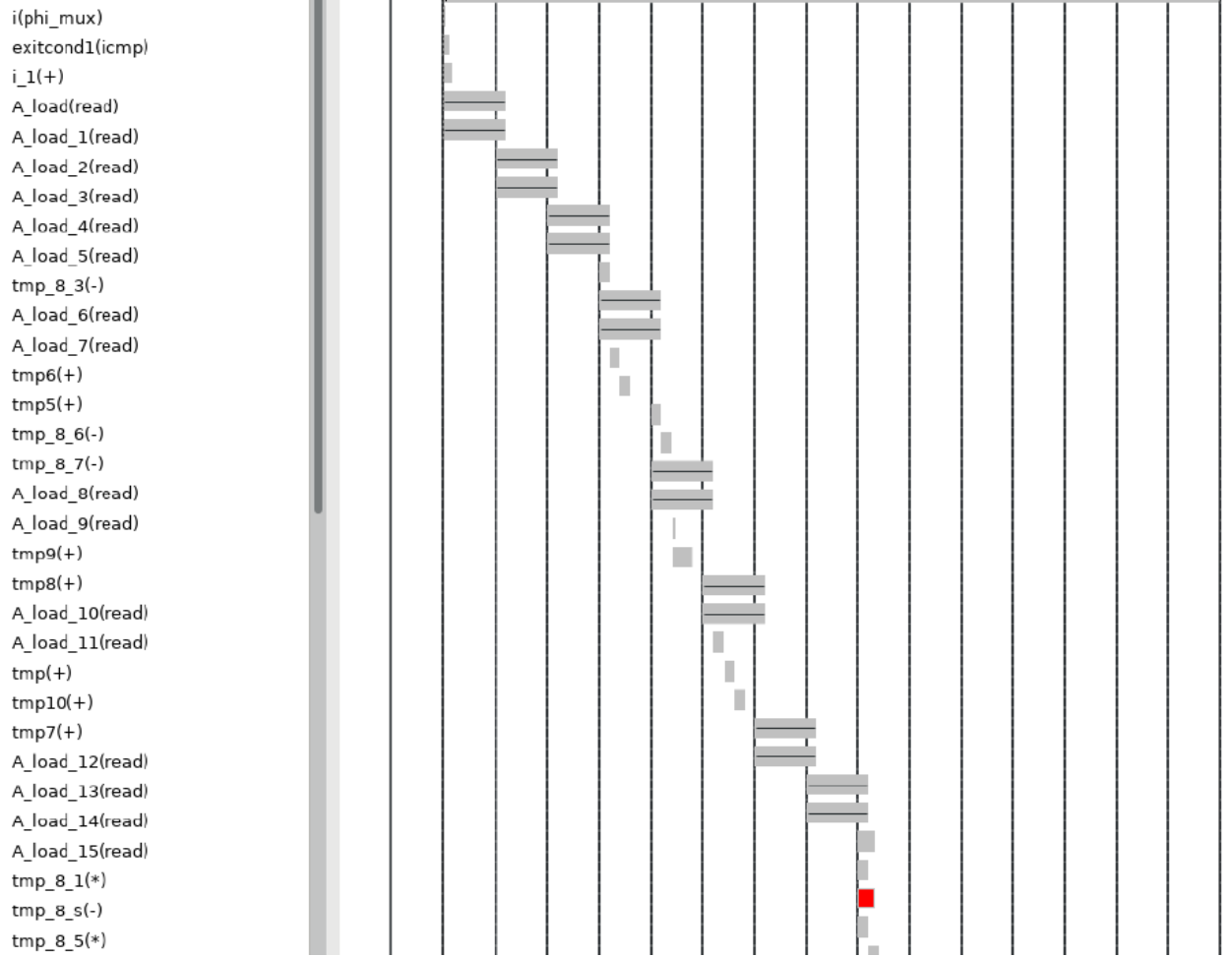
Оценка использования имеет вид:

Utilization Estimates					
▣ Summary					
Name	BRAM_18K	DSP48E	FF	LUT	
DSP	-	4	-	-	
Expression	-	-	0	478	
FIFO	-	-	-	-	
Instance	-	-	-	-	
Memory	-	-	-	-	
Multiplexer	-	-	-	218	
Register	-	-	229	-	
Total	0	4	229	696	
Available	40	40	16000	8000	
Utilization (%)	0	10	1	8	

Профиль производительности имеет вид:

Performance Profile		Resource Profile				
		Pipelined	Latency	Iteration Latency	Initiation Interval	Trip count
loop_imperfect	-	-	261~301	-	262 ~ 302	-
LOOP_I	no	no	260 ~ 300	13 ~ 15	-	20

Данные планировщика просмотра имеет вид:



Данные обзора ресурсов имеет вид:

Resource\Control Step	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
1 I/O Ports																
2 A(p1)		read	read	read	read	read	read	read	read	read	read					
3 A(p0)		read	read	read	read	read	read	read	read	read	read					
4 B(p0)																write
5 Memory Ports																
6 A(p1)		read	read	read	read	read	read	read	read	read	read					
7 A(p0)		read	read	read	read	read	read	read	read	read	read					
8 B(p0)																write
9 Expressions																
10 i_1_fu_327		+				*										
11 i_phi_mux		phi_mux														
12 exitcond1_fu_321		icmp														
13 tmp6_fu_383					+											
14 tmp5_fu_393					+											
15 tmp8_3_fu_349					-											
16 tmp8_fu_477						+										
17 tmp9_fu_471						+										
18 tmp8_7_fu_449						-										
19 tmp8_6_fu_423						-										
20 tmp10_fu_532							+									
21 tmp_fu_522							+									
22 tmp7_fu_542							+									
23 tmp16_fu_662										+						
24 tmp15_fu_656										+						
25 tmp8_9_fu_608										-						
26 tmp8_s_fu_573										-						
27 tmp8_10_fu_634										-						
28 grp_fu_909										+						
29 grp_fu_916										+						
30 tmp13_fu_747											+					
31 tmp21_fu_763											+					
32 tmp20_fu_753											+					
33 tmp2_fu_735											+					
34 tmp4_fu_716											+					
35 tmp3_fu_722											+					
36 tmp1_fu_782												+				
37 grp_fu_925												+				
38 acc_2_s_fu_832													+			
39 tmp19_fu_808														+		

Представленное решение имеет лучшую оценочную производительность, за счёт разворачивания внутреннего цикла, несмотря на то, что не укладывается в отведённый временной интервал.

9. Вывод

Директива UNROLL позволяет «развернуть» цикл для получения конвейера, однако, чем больше «глубина» такого конвейера, тем больше количество затраченных ресурсов. Для управления глубиной конвейера используется параметр factor, что позволяет получить «золотую середину» между пропускной способностью и требуемыми ресурсами. Для объединения циклов и оптимизации проекта при написании кода стоит учитывать, что циклы лучше писать в виде PERFECT для максимальной оптимизации результата.