

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

Лабораторная №13

Предмет: Проектирование реконфигурируемых гибридных вычислительных систем

Тема: Сравнение типов данных

Задание 2

**Студенты:**

Соболь В.

Темнова А.С.

Группа: 13541/3

**Преподаватель:**

Антонов А.П.

Санкт-Петербург  
2019

# Содержание

<b>1. Задание</b>	<b>3</b>
<b>2. Исходный код</b>	<b>4</b>
<b>3. Скрипт</b>	<b>6</b>
<b>4. Решение 1а</b>	<b>6</b>
4.1. Моделирование . . . . .	6
<b>5. Решение 2а</b>	<b>7</b>
5.1. Моделирование . . . . .	7
5.2. Синтез . . . . .	7
<b>6. Вывод</b>	<b>9</b>

# 1. Задание

1. Создать проект lab13\_2
2. Микросхема: xa7a12tcsg325-1q
3. В папке source текст функции apint\_promotion  
*Познакомьтесь с ним.*
4. Познакомьтесь с тестом.
5. Исследование:
6. Solution\_1a

- Создать версию apint\_ , в которой будет убран Кастинг

```
#include "apint_promotion.h"

dout_t apint_promotion(din_t a,din_t b) {
    dout_t tmp;

    tmp = (dout_t)a * (dout_t)b;
    return tmp;
}
```

7. Осуществить моделирование (при необходимости изменить тест)
8. задать: clock period 10; clock\_uncertainty 0.1
9. установить реализацию ПО УМОЛЧАНИЮ
10. 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
11. Выполнить cosimulation и привести временную диаграмму
12. Solution\_2a
  - Использовать исходную функцию apint\_promotion

- Осуществить моделирование
- задать: clock period 10; clock\_uncertainty 0.1
- 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

13. Сравнить два решения (solution\_1a и solution\_2a) и сделать выводы

## 2. Исходный код

Ниже приведен исходный код устройства и теста.

```

1 #include "apint_promotion.h"
2
3 dout_t apint_promotion(din_t a, din_t b) {
4     dout_t tmp;
5     #ifdef ENABLE_TYPE_CAST
6     tmp = (dout_t)a * (dout_t)b;
7     #else
8     tmp = a * b;
9     #endif
10    return tmp;
11 }

```

Рис. 2.1. Исходный код устройства

```

1 #ifndef _APINT_PROMOTION_H_
2 #define _APINT_PROMOTION_H_
3
4 #include <stdio.h>
5 #include "ap_cint.h"
6
7 typedef int18 din_t;
8 typedef int36 dout_t;
9
10 dout_t apint_promotion(din_t a, din_t b);
11
12 #endif

```

Рис. 2.2. Заголовочный файл

```

1 #include "apint_promotion.h"
2
3 int main () {
4     din_t A, B;
5     dout_t RES;
6     int i, retval=0;
7     FILE *fp;
8
9     // Save the results to a file
10    fp=fopen("result.dat", "w");
11
12    // Call the function
13    A=65536;
14    B=65536;
15    for(i=0; i<(20);++i) {
16        RES=apint_promotion(A,B);
17    #ifndef __MINGW32__
18        fprintf(fp, "%lld_\n", RES);
19    #else
20        fprintf(fp, "%I64d_\n", RES);
21    #endif
22        A=A+1024;
23        B=B-2047;
24    }
25    fclose(fp);
26
27    // Compare the results file with the golden results
28    retval = system("diff --brief -w result.dat result.golden.dat");
29    if (retval != 0) {
30        printf("Test_failed_!!!\n");
31        retval=1;
32    } else {
33        printf("Test_passed_!\n");
34    }
35
36    // Return 0 if the test passes
37    return retval;
38 }

```

Рис. 2.3. Исходный код теста

### 3. Скрипт

Ниже приводится скрипт, для автоматизации выполнения лабораторной работы.

```
1 open_project -reset lab13_2
2
3
4 add_files apint_promotion.c
5 add_files -tb apint_promotion_test.c
6 add_files -tb result.golden.dat
7
8 set_top apint_promotion
9
10 # open_solution solution_1a -reset
11 # set_part {xa7a12tcsg325-1q}
12 # create_clock -period 10ns
13 # set_clock_uncertainty 0.1
14
15
16 # csim_design
17 # csynth_design
18 # cosim_design -trace_level all
19
20
21 add_files apint_promotion.c -cflags "-w_-DENABLE_TYPE_CAST"
22
23 open_solution solution_2a -reset
24 set_part {xa7a12tcsg325-1q}
25 create_clock -period 10ns
26 set_clock_uncertainty 0.1
27
28
29 csim_design
30 csynth_design
31 # cosim_design -trace_level all
```

Рис. 3.1. Скрипт

### 4. Решение 1a

#### 4.1. Моделирование

Ниже приведены результаты моделирования.

```
INFO: [APCC 202-1] APCC is done.
    Generating csim.exe
Files result.dat and result.golden.dat differ
Test failed !!!
@E Simulation failed: Function 'main' returns nonzero value '1'.
ERROR: [SIM 211-100] 'csim_design' failed: nonzero return value.
INFO: [SIM 211-3] ***** CSIM finish *****
```

Рис. 4.1. Результаты моделирования

По результатам моделирования видно, что устройство работает не корректно.

## 5. Решение 2а

### 5.1. Моделирование

Ниже приведены результаты моделирования.

```
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 *****
```

Рис. 5.1. Результаты моделирования

По результатам моделирования видно, что устройство работает корректно.

### 5.2. Синтез

По оценке производительности видно, что устройство соответствует заданным критериям.

Performance Estimates

[-] Timing (ns)

[-] Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00	7.180	0.10

[-] Latency (clock cycles)

[-] Summary

Latency		Interval		
min	max	min	max	Type
0	0	0	0	none

Рис. 5.2. Performance estimates

## Utilization Estimates

### Summary

Name	BRAM_18K	DSP48E	FF	LUT
DSP	-	1	-	-
Expression	-	-	-	-
FIFO	-	-	-	-
Instance	-	-	-	-
Memory	-	-	-	-
Multiplexer	-	-	-	-
Register	-	-	-	-
Total	0	1	0	0
Available	40	40	16000	8000
Utilization (%)	0	2	0	0

Рис. 5.3. Utilization estimates

Performance Profile		Resource Profile			
	Pipelined	Latency	Iteration Latency	Initiation Interval	Trip count
apint_promotion	-	0	-	1	-

Рис. 5.4. Performance profile

Operation\Control Step	0
b_read(read)	
a_read(read)	
tmp_2(*)	

Рис. 5.5. Scheduler viewer



	Resource\Control S...	C0
1	<input type="checkbox"/> I/O Ports	
2	b	read
3	ap_return	ret
4	a	read
5	<input type="checkbox"/> Expressions	
6	tmp_2_fu_34	*

Рис. 5.6. Resource viewer

## 6. Вывод

В данной работе продемонстрирована важность порядка преобразования типов. В первом решении, операция умножения выполняется для двух операндов с типом `d_in`, который значительно меньше, чем `d_out`. Во время умножения происходит переполнение, и результат оказывается некорректным.

Во втором решении, умножение выполняется над операндами с типом `d_out`, так как произведено явное преобразование типов. Это тип более вместительный, и переполнения не происходит.