

Robei 集成电路实战



带符号位小数的加法设计





带符号位小数的加法设计

1

有符号数

2

有符号数和无符号数的说明

3

实验设计：前端

4

实验设计：后端





带符号位小数的加法设计

有符号数

有符号数是针对二进制来讲的。用最高位作为符号位，“0”代表“+”，“1”代表“-”；其余数位用作数值位，代表数值。

有符号数的表示：计算机中的数据用二进制表示，数的符号也只能用0/1表示。一般用最高有效位（**MSB**）来表示数的符号，正数用0表示，负数用1表示。

有符号数的编码方式，常用的是补码，另外还有原码和反码等。用不同二进制编码方式表示有符号数时，所得到的机器数可能不一样，但是真值应该是相同的。





带符号位小数的加法设计

数字电路中的有符号数：

可以通过关键字 `signed` 来定义：

```
input  signed [7:0 ] a, b;  
output signed [15:0] c;  
wire   signed [15:0] x;  
reg     signed [15:0] y;
```

如果没有 `signed` 关键字，所有数据默认都被当做无符号数。



带符号位小数的加法设计

如何确定有符号数的数值：

正数：最高位为0，其数值与相应的无符号数一样。

$\text{signed } 0110 = \text{unsigned } 0110 = (\text{dec}) 6$

负数：最高位为1，其绝对值等于全部数位取反加1。

$1011 \text{----} (\text{全部数位取反}) \text{----} 0100 \text{----} (+1) \text{----} 0101 = 5$

因此1011绝对值为5，本身的值为 -5。



带符号位小数的加法设计

带符号位的数：

最高位为独立的符号位；剩余的位数则作为无符号数表示该数的值。

0100_1100：符号位为0，是正数；
剩余位数_100_1100作为无符号数表示76，所以该数的值为+76；

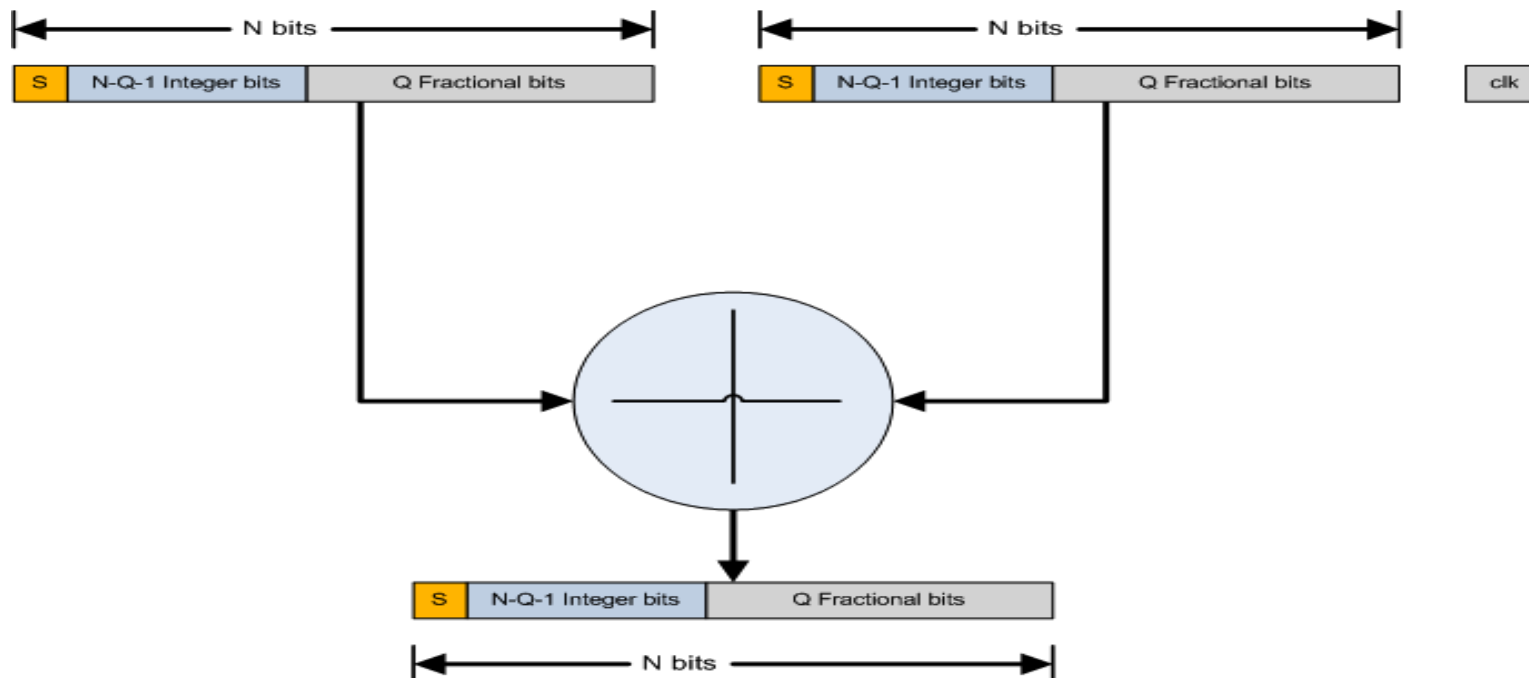
1011_0011：符号位为1，是负数；
剩余位数 _011_0011作为无符号数表示51，所以该数的值为-51。



带符号位小数的加法设计

数据格式:

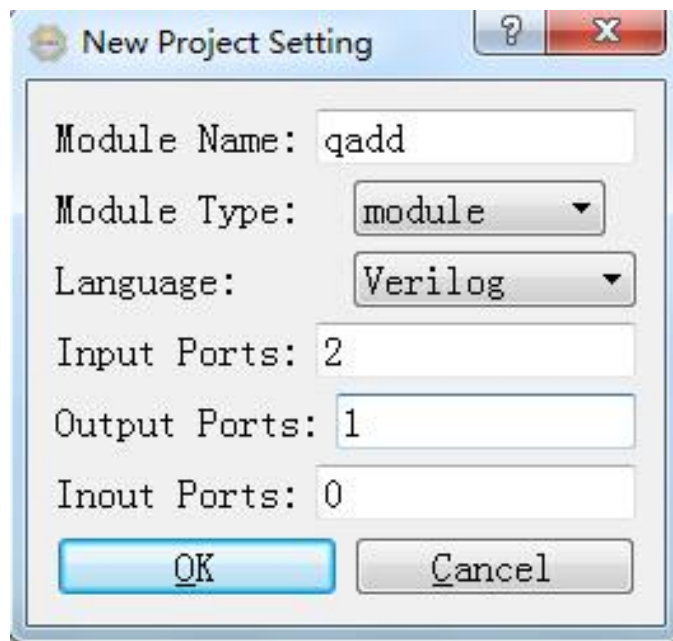
|1|<- N-Q-1 bits ->|<--- Q bits --->|
|S| |FFFFFFFFFFFFFFFFFFFFF|





带符号位小数的加法设计

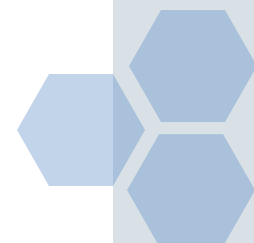
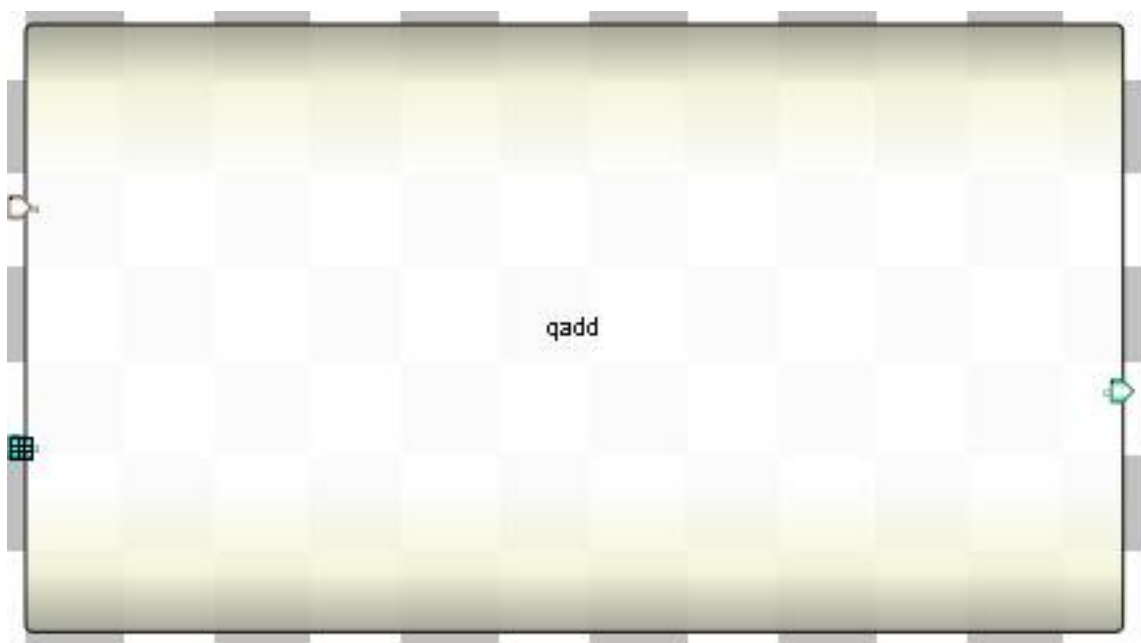
实验设计：前端设计





带符号位小数的加法设计

Name	Inout	DataType	Datasize	Function
a	input	wire	32	input a
b	input	wire	32	input b
c	output	reg	32	output c





带符号位小数的加法设计

实验代码:

```
parameter N = 4;           //可扩展
reg [N-1:0] res;
assign c = res;

always @(a or b)
begin
    if(a[N-1] == b[N-1])    //a和b同符号
    begin
        res[N-2:0] = a[N-2:0] + b[N-2:0];
        res[N-1] = a[N-1];
    end
    else if(a[N-1] == 0 && b[N-1] == 1)
    begin                    //a为正 b为负
        if( a[N-2:0] > b[N-2:0] )
        begin                //|a|>|b|, 结果为正
            res[N-2:0] = a[N-2:0] - b[N-2:0];
            res[N-1] = 0;
        end
    end
end
```

.....

Copyright © Robei





带符号位小数的加法设计

实验代码:

```
.....
else                                     //|a|≤|b|结果为0或者负
begin
    res[N-2:0] = b[N-2:0] - a[N-2:0];
    if (res[N-2:0] == 0)                //结果为0
        res[N-1] = 0;
    else
        res[N-1] = 1;                  //结果为负
end
end
else                                     //a为负 b为正
begin
    if( a[N-2:0] >= b[N-2:0] )
    begin                                //|a|≥|b|结果为0或者负
        res[N-2:0] = a[N-2:0] - b[N-2:0];
        if (res[N-2:0] == 0)
            res[N-1] = 0; //结果为0
        else
            res[N-1] = 1; //结果为负
    end
end
.....
```





带符号位小数的加法设计

实验代码：

```
.....  
else                                //|a|<|b|结果为正  
begin  
    res[N-2:0] = b[N-2:0] - a[N-2:0];  
    res[N-1] = 0;  
end  
end  
end
```

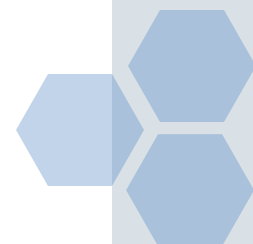
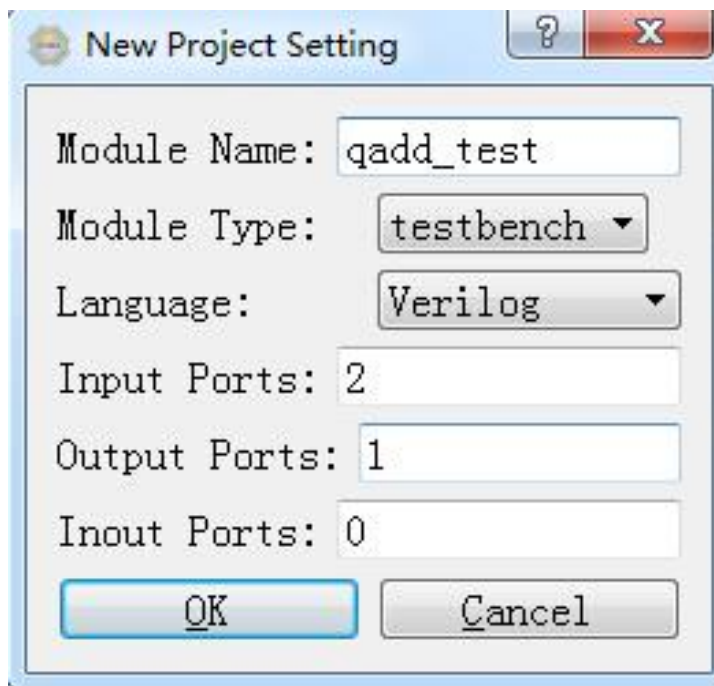
设计完成后，保存并执行。
检查是否有错误，并生成.v文件。





带符号位小数的加法设计

测试模块设计：

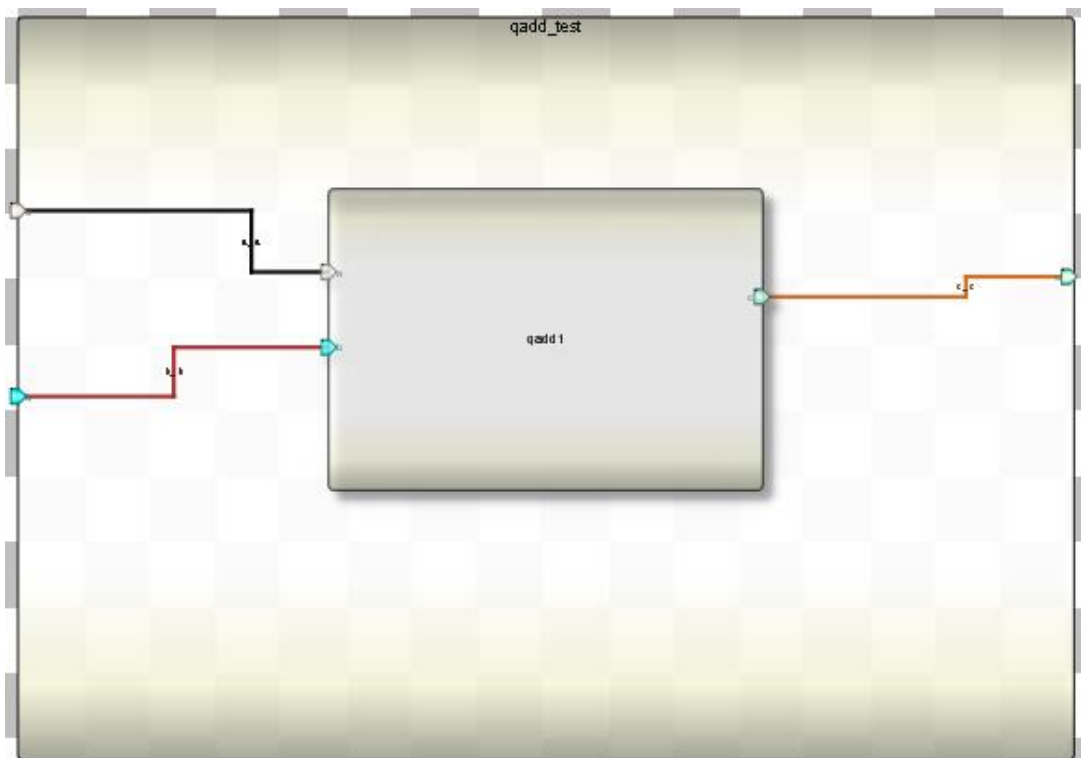




带符号位小数的加法设计

Ports:

Name	Inout	DataType	Datasize	Function
a	input	reg	32	
b	input	reg	32	
c	output	wire	32	



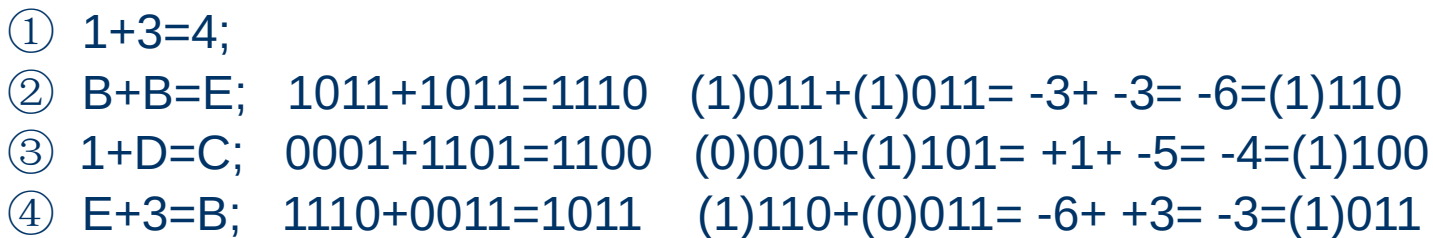


带符号位小数的加法设计

测试代码:

```
initial begin
    a = 4'b0001;
    b = 4'b0011;
#10
    a = 4'b1011;
    b = 4'b1011;
#10
    a = 4'b0001;
    b = 4'b1101;
#10
    a = 4'b1110;
    b = 4'b0011;
#10
    $finish;
end
```

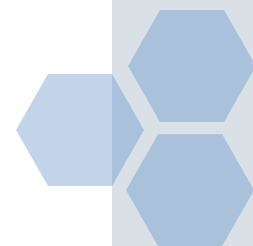
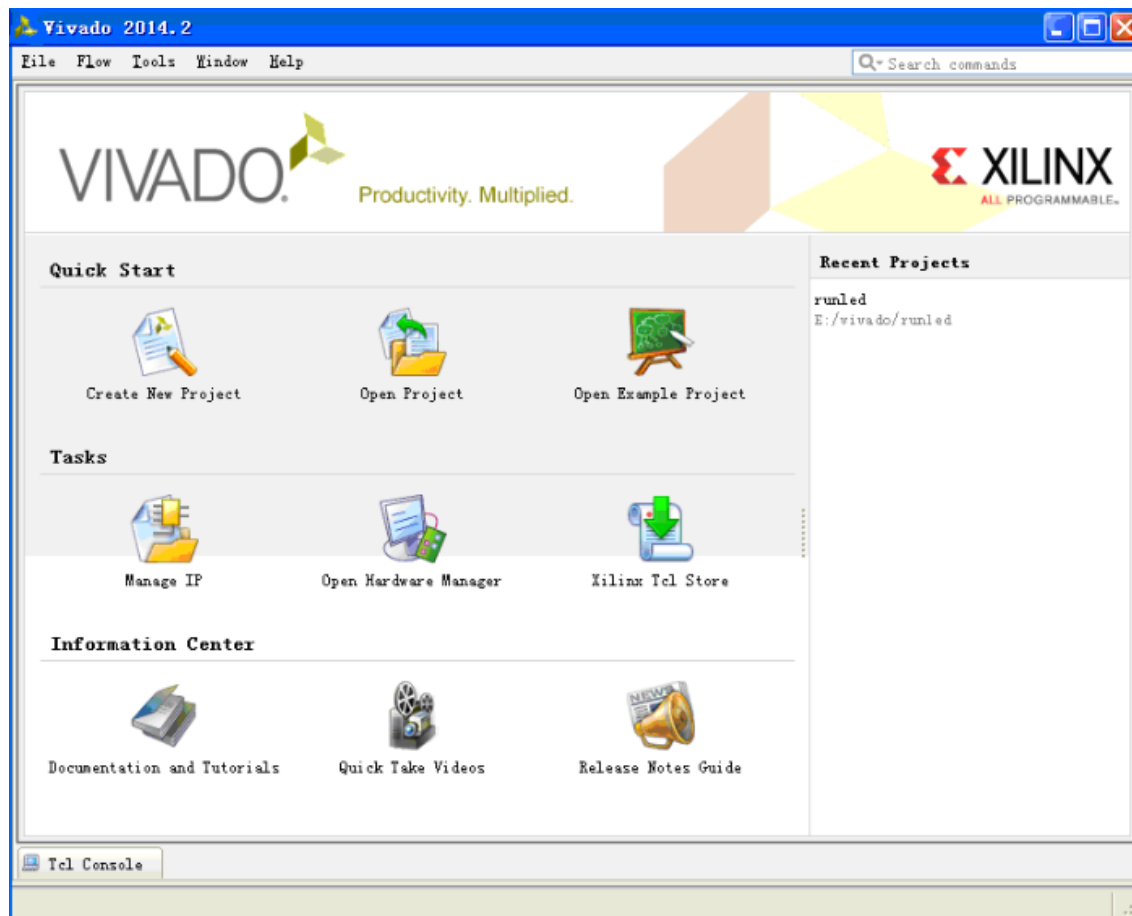






带符号位小数的加法设计

实验设计：后端





带符号位小数的加法设计

New Project

Project Name
Enter a name for your project and specify a directory where the project data files will be stored.

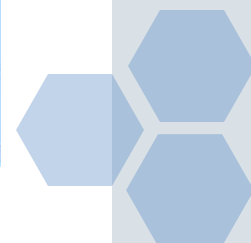
Project name:

Project location:

☒ Create project subdirectory

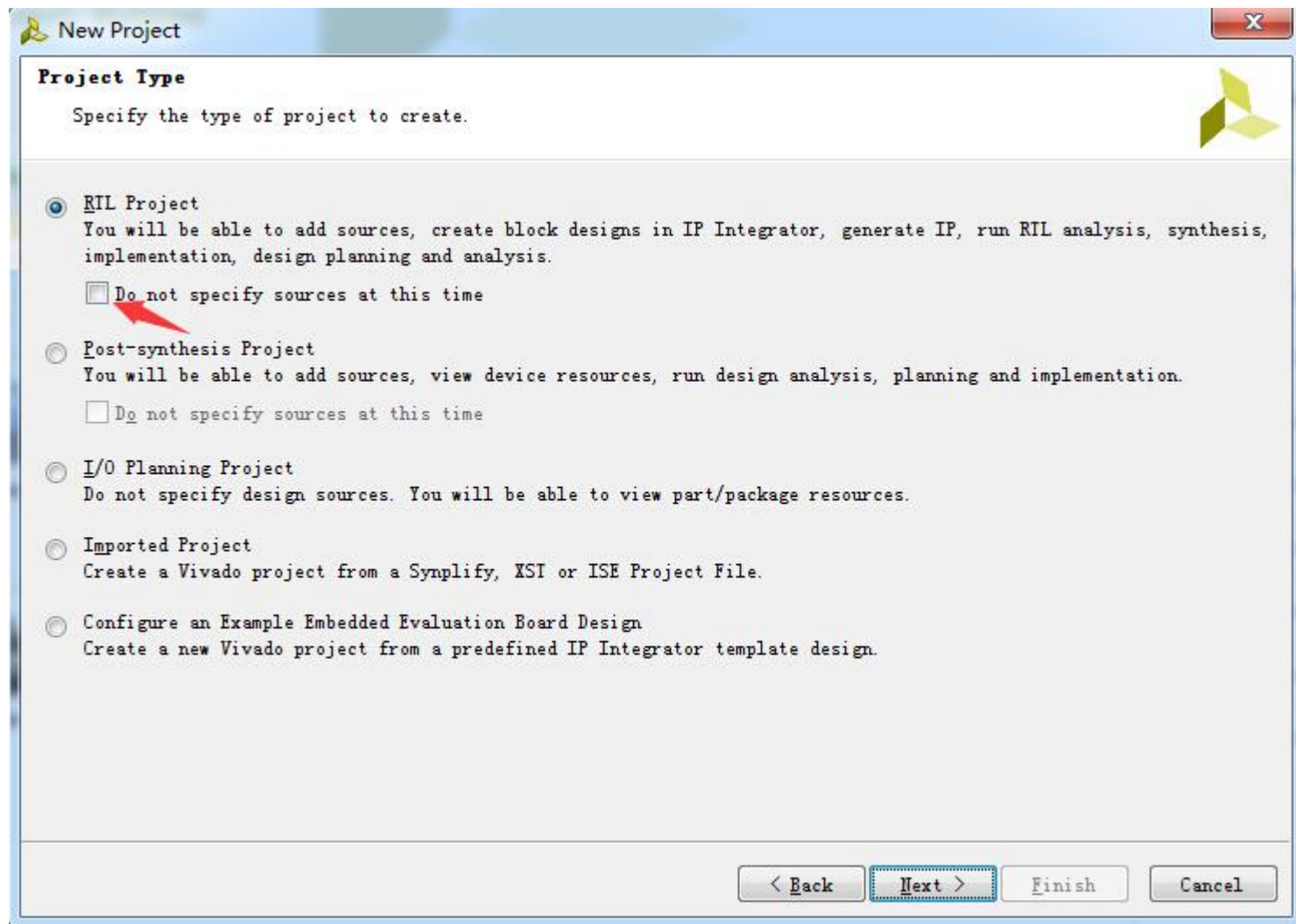
Project will be created at: E:/Robei_modules/calculate/lab1

< Back Next > Finish Cancel



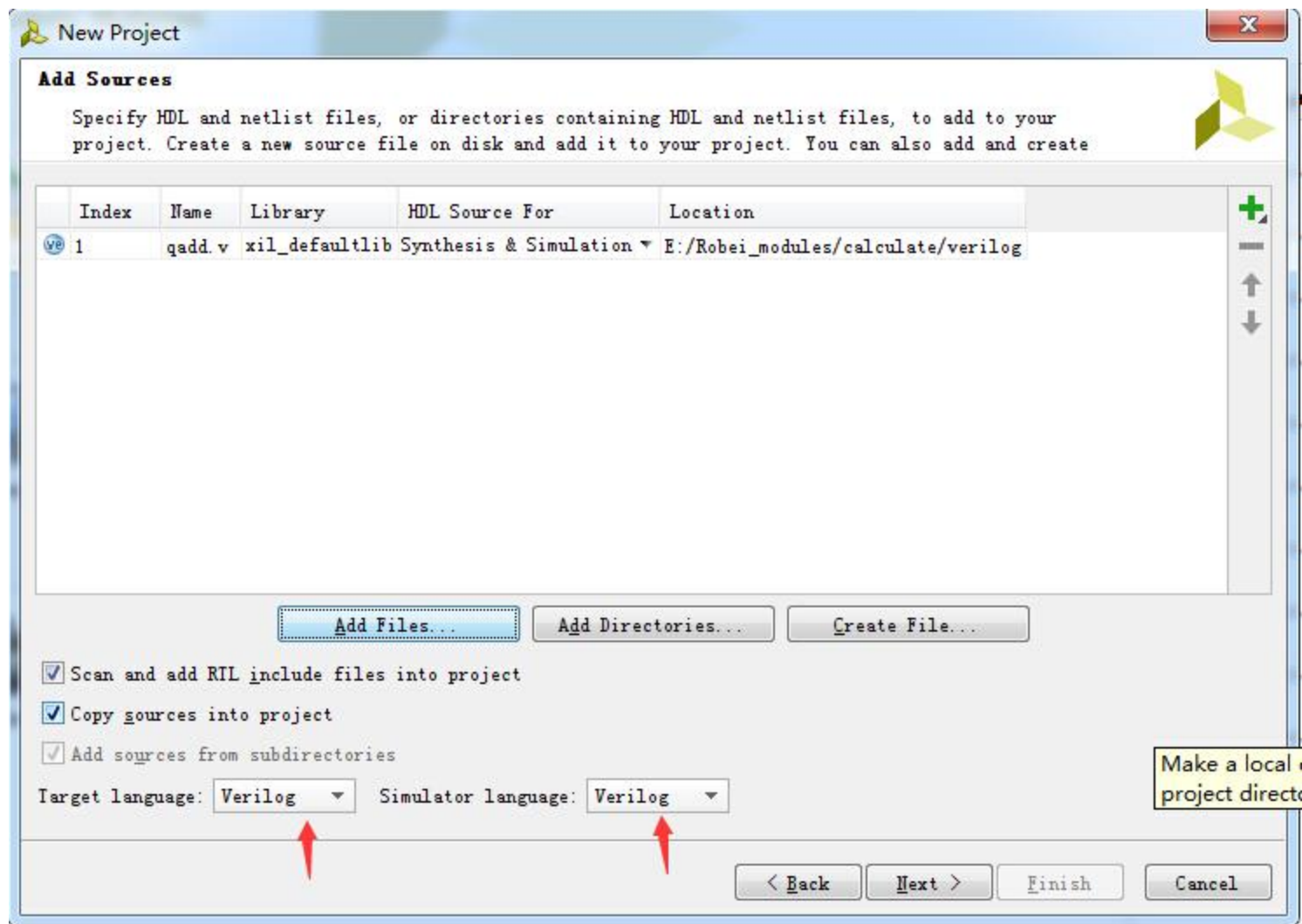


带符号位小数的加法设计



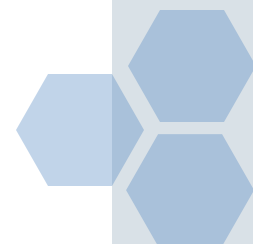
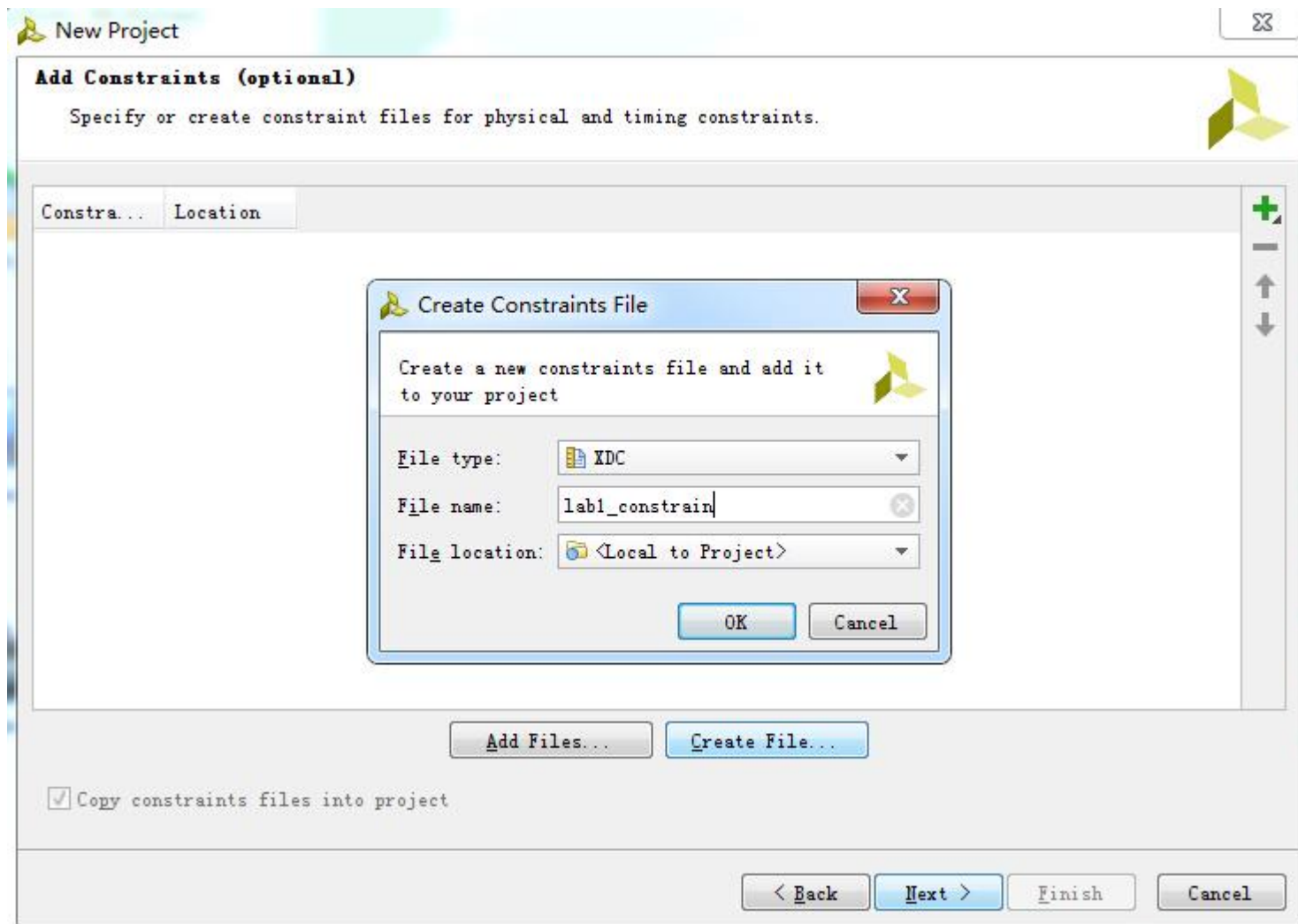


带符号位小数的加法设计





带符号位小数的加法设计





带符号位小数的加法设计

New Project

Default Part

Choose a default Xilinx part or board for your project. This can be changed later.

Select: ☒ Parts ☐ Boards

Filter

Product category: All Package: clg400

Family: Zynq-7000 Speed grade: -1

Sub-Family: Zynq-7000 Temp grade: All Remaining

Reset All Filters

Search: Q

Part	I/O Pin Count	Available IOBs	LUT Elements	FlipFlops	Block RAMs	DSPs	Gb Transceivers
xc7z010clg400-1	400	100	17600	35200	60	80	0
xc7z020clg400-1	400	125	53200	106400	140	220	0

< Back Next > Finish Cancel



带符号位小数的加法设计

lab1 - [E:/Robei_modules/calculate/lab1/lab1.xpr] - Vivado 2014.4

File Edit Flow Tools Window Layout View Help

Search commands

write_bitstream Complete

Flow Navigator

- Project Manager
 - Project Settings
 - Add Sources
 - Language Templates
 - IP Catalog
- IP Integrator
 - Create Block Design
 - Open Block Design
 - Generate Block Design
- Simulation
 - Simulation Settings
 - Run Simulation
- RIL Analysis
 - Open Elaborated Design
- Synthesis
 - Synthesis Settings
 - Run Synthesis
 - Open Synthesized Design
- Implementation
 - Implementation Settings
 - Run Implementation
 - Open Implemented Design
- Program and Debug
 - Bitstream Settings

Project Manager - lab1

Sources

- Design Sources (1)
 - qadd (qadd.v)
- Constraints (1)
 - constrs_1 (1)
 - lab1_constrain.xdc
- Simulation Sources (1)

Hierarchy Libraries Compile Order

Sources Templates

Properties

lab1_constrain.xdc

Location: E:/Robei_modules/calculate/lab1/lab1.srcs/constrs_1/lab1_constrain.xdc

Type: XDC

General Properties

Log

Creating bitmap...
Creating bitstream...
Writing bitstream ./qadd.bit...
INFO: [Vivado 12-1842] Bitgen Completed Successfully.
INFO: [Project 1-120] WebTalk data collection is mandatory for users of free Webpack licenses. To see the specific WebTalk data collected for your design, open the Vivado Webpack License page.
INFO: [Common 17-83] Releasing license: Implementation
write_bitstream: Time (s): cpu = 00:00:32 ; elapsed = 00:00:41 . Memory (MB): peak = 794.184 ; gain = 343.605
INFO: [Common 17-206] Exiting Vivado at Thu Jun 04 15:44:14 2015...

Project Summary

qadd.v

E:/Robei_modules/calculate/lab1/lab1.srcs/sources_1/imports/verilog/qadd.v

```
7 //---Ports declaration---  
8 input [3:0] a;  
9 input [3:0] b;  
10 output [3:0] c;  
11  
12 wire [3:0] a;  
13 wire [3:0] b;  
14 wire [3:0] c;  
15  
16 //---Code starts here---  
17 parameter N = 4;  
18 reg [N-1:0] res;  
19 assign c = res;  
20  
21 always @(a or b)  
22 begin  
23     if(a[N-1] == b[N-1])
```



带符号位小数的加法设计

约束文件代码:

```
# ZYBO board Pins
# QADD
#PMOD JE1~4, 7~10. Switches  a[0~3], b[0~3]

set_property PACKAGE_PIN V12 [get_ports {a[0]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[0]}]

set_property PACKAGE_PIN W16 [get_ports {a[1]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[1]}]

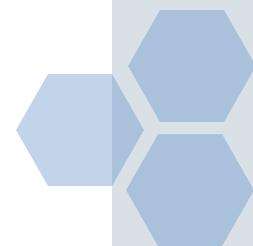
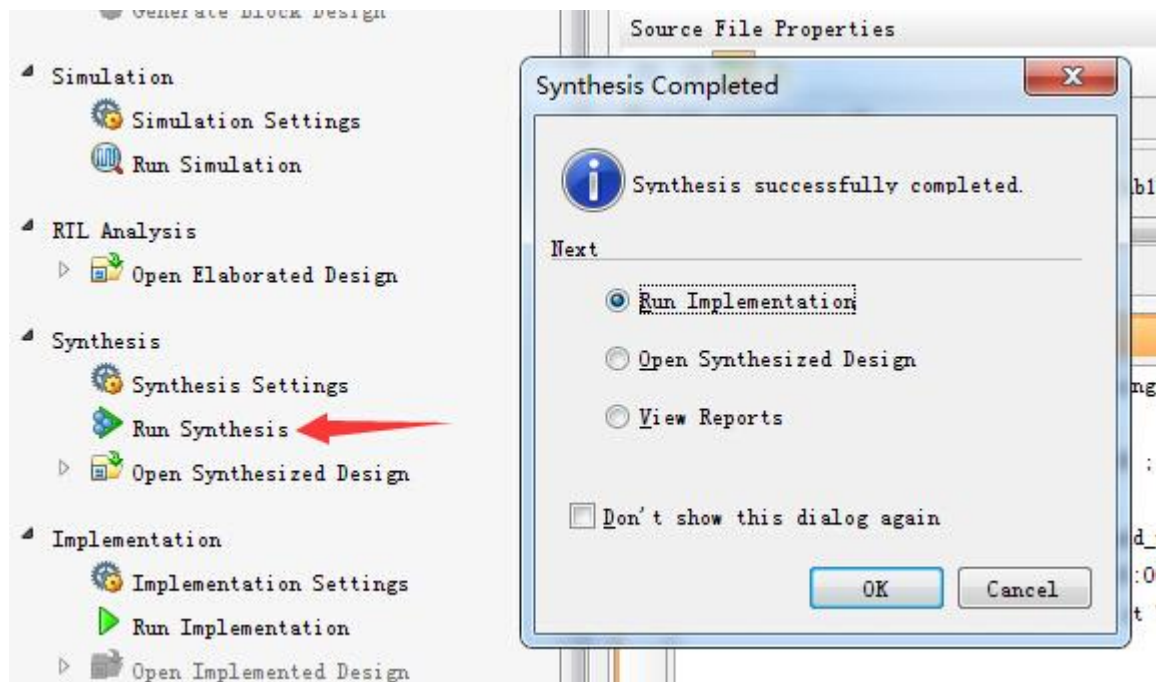
set_property PACKAGE_PIN J15 [get_ports {a[2]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[2]}]

set_property PACKAGE_PIN H15 [get_ports {a[3]}]
set_property IOSTANDARD LVCMOS33 [get_ports {a[3]}]
.....
.....
```



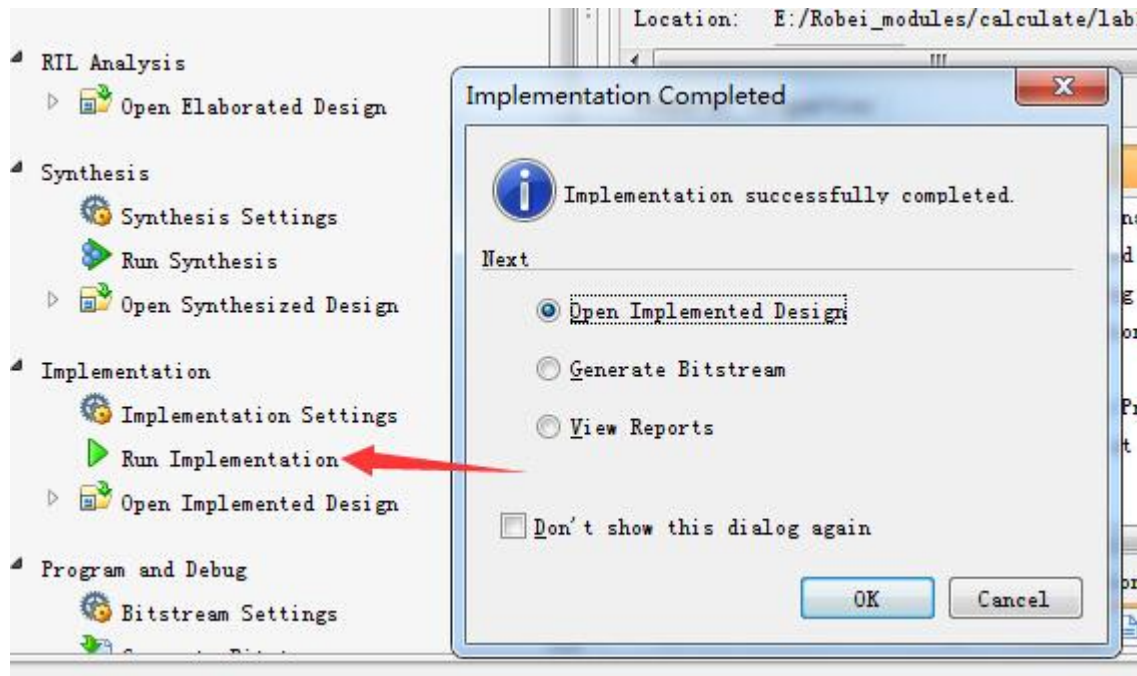


带符号位小数的加法设计



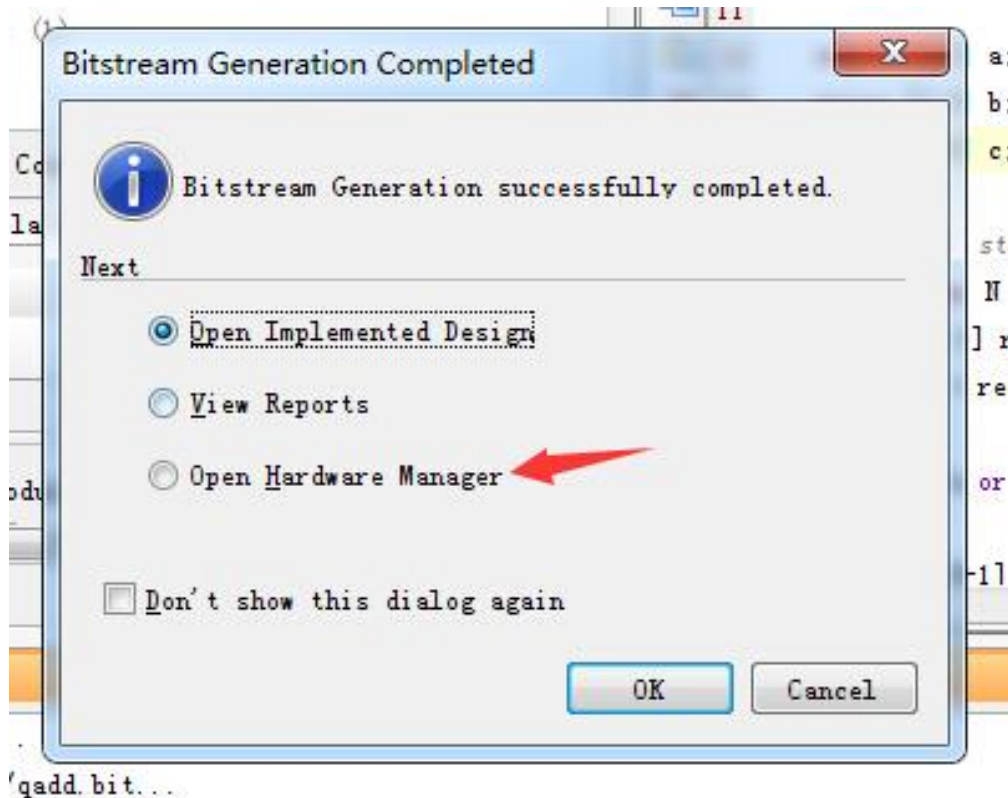


带符号位小数的加法设计



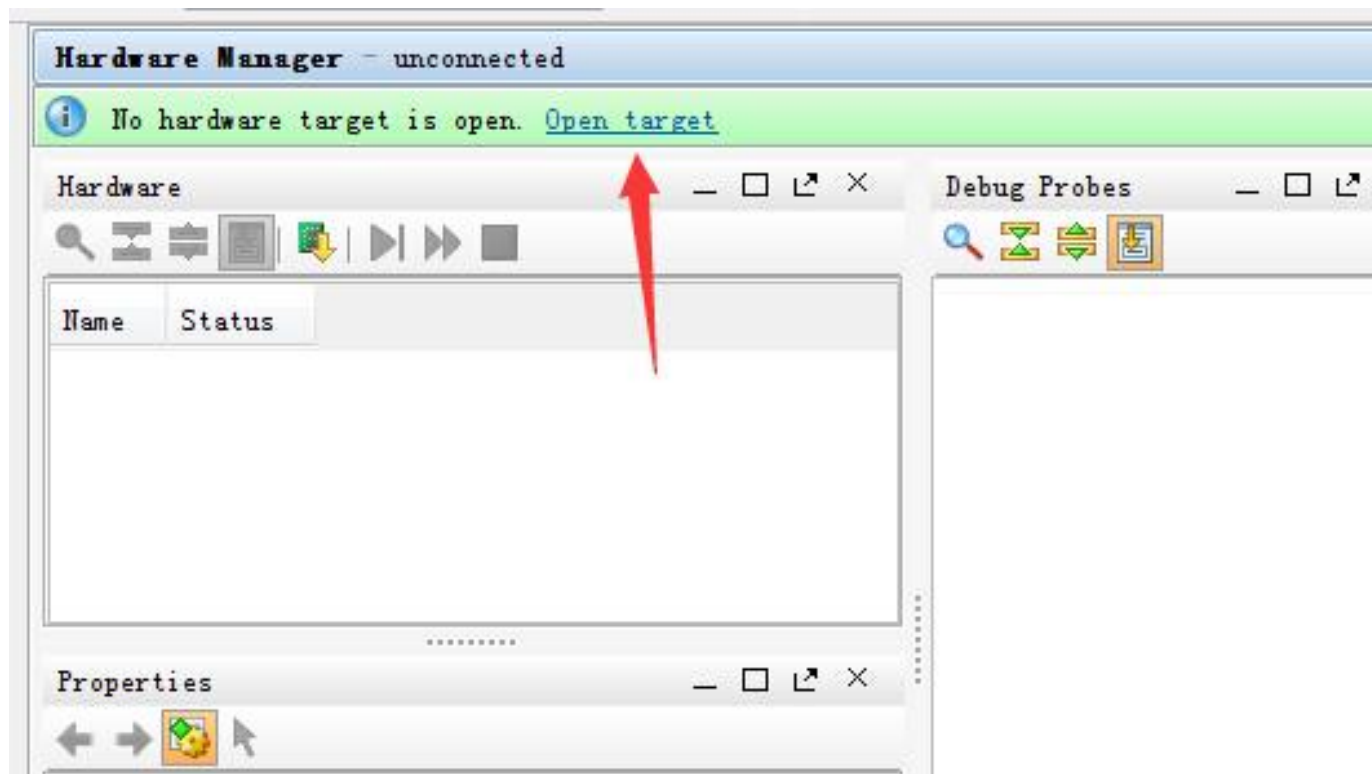


带符号位小数的加法设计



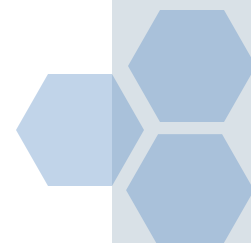
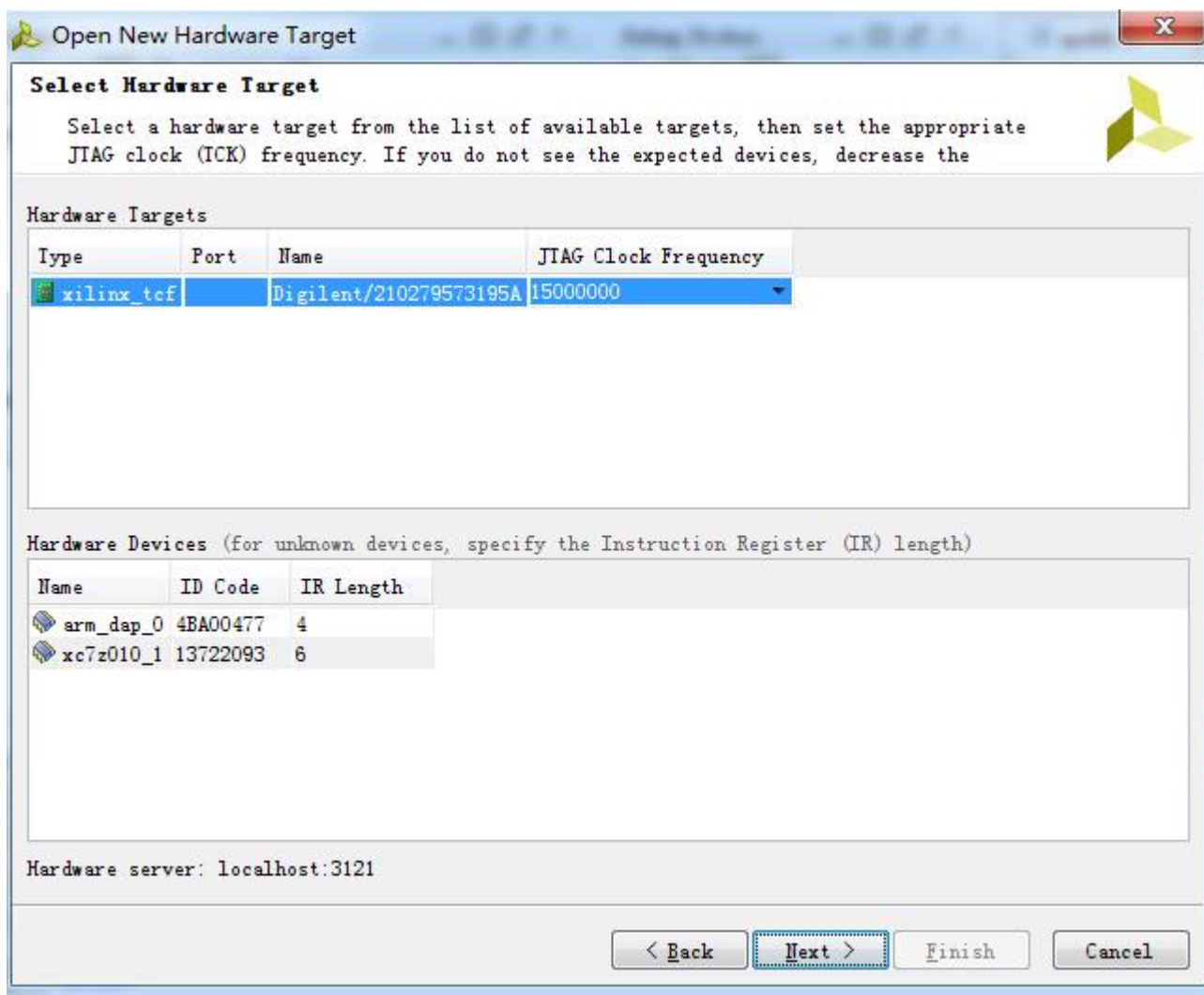


带符号位小数的加法设计



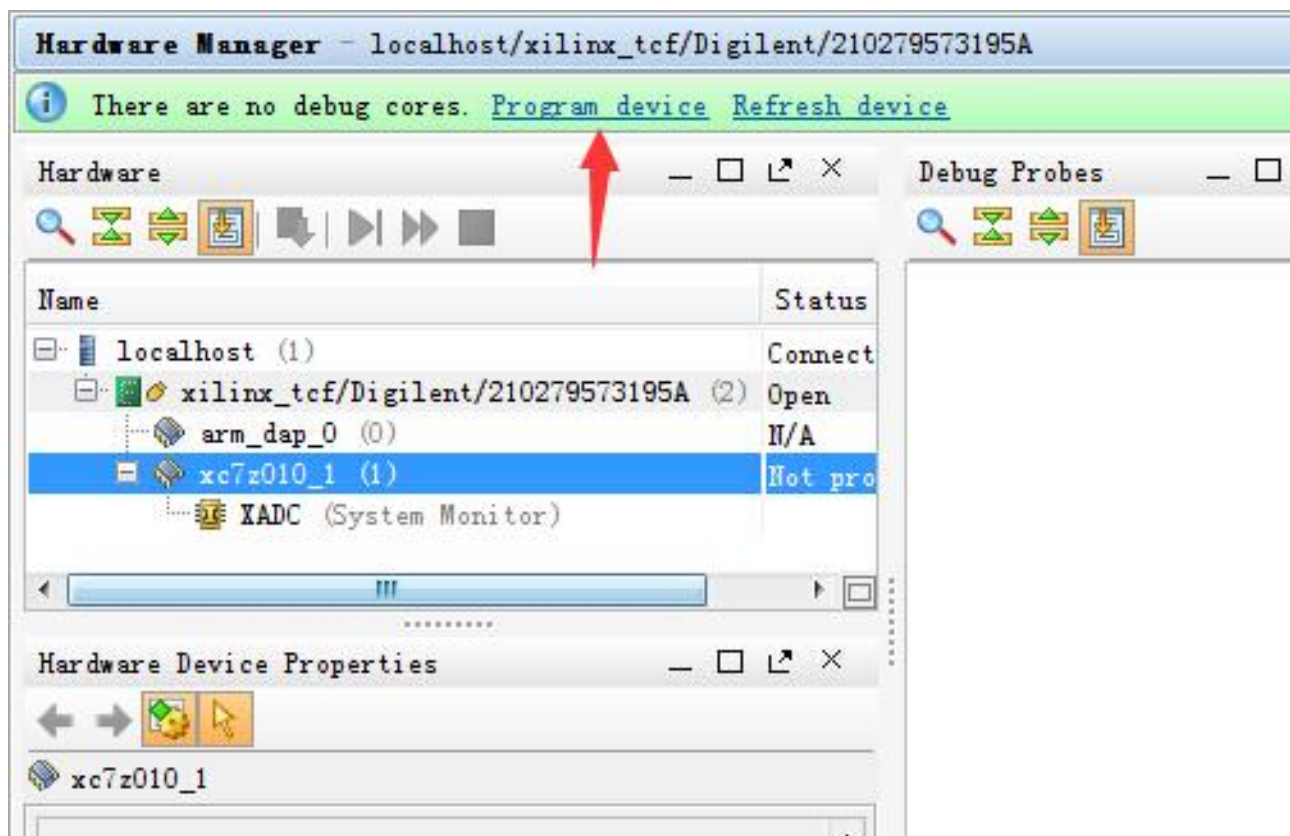


带符号位小数的加法设计





带符号位小数的加法设计



www.robei.com



Thank You!