一、实验目标

使用正弦波信号发生器，生成仿真波形。并进一步熟悉Quartus II，初步了解CPU功能模块设计，实现VHDL顶层设计

二、实验原理

正弦信号发生器的结构由4部分组成：数据计数器或地址发生器、波形数据ROM、D/A和滤波电路。数据计数器或地址发生器产生控制ROM波形数据表的地址，输出信号的频率由ROM地址的变化速率决定，变化越快，输出频率越高；波形数据表ROM用于存放波形数据，可以存放正弦波、三角波或者其他波形数据；D/A转换器将ROM输出的数据转换成模拟信号，经过滤波电路后输出。VHDL代码完成正弦信号发生器以及data\_rom的功能，生成一组正弦信号数据rom.mif。进行波形仿真，给正弦信号发生器一组周期为10ns的时钟信号，生成正弦仿真波形。

三、实验准备

f3源代码

LIBRARY IEEE; --正弦信号发生器源文件

USE IEEE.STD\_LOGIC\_1164.ALL;

USE IEEE.STD\_LOGIC\_UNSIGNED.ALL;

ENTITY f3 IS

PORT ( CLK : IN STD\_LOGIC; --信号源时钟

DOUT : OUT STD\_LOGIC\_VECTOR (7 DOWNTO 0) );--8位波形数据输出

END;

ARCHITECTURE DACC OF f3 IS

COMPONENT data\_rom --调用波形数据存储器LPM\_ROM文件：data\_rom.vhd声明

PORT(address : IN STD\_LOGIC\_VECTOR (5 DOWNTO 0);--6位地址信号

inclock : IN STD\_LOGIC ;--地址锁存时钟

q : OUT STD\_LOGIC\_VECTOR (7 DOWNTO 0) );

END COMPONENT;

SIGNAL Q1 : STD\_LOGIC\_VECTOR (5 DOWNTO 0); --设定内部节点作为地址计数器

BEGIN

PROCESS(CLK ) --LPM\_ROM地址发生器进程

BEGIN

IF CLK'EVENT AND CLK = '1' THEN Q1<=Q1+1; --Q1作为地址发生器计数器

END IF;

END PROCESS;

u1 : data\_rom PORT MAP(address=>Q1, q => DOUT,inclock=>CLK);--例化

END;

data\_rom源代码

LIBRARY ieee;

USE ieee.std\_logic\_1164.all;

LIBRARY altera\_mf;

USE altera\_mf.altera\_mf\_components.all; --使用宏功能库中的所有元件

ENTITY data\_rom IS

PORT (address : IN STD\_LOGIC\_VECTOR (5 DOWNTO 0);

inclock : IN STD\_LOGIC ;

q : OUT STD\_LOGIC\_VECTOR (7 DOWNTO 0) );

END data\_rom;

ARCHITECTURE SYN OF data\_rom IS

SIGNAL sub\_wire0 : STD\_LOGIC\_VECTOR (7 DOWNTO 0);

COMPONENT altsyncram --例化altsyncram元件，调用了LPM模块altsyncram

GENERIC ( --参数传递语句

intended\_device\_family : STRING; --类属参量数据类型定义

width\_a : NATURAL;

widthad\_a : NATURAL;

numwords\_a : NATURAL;

operation\_mode : STRING;

outdata\_reg\_a : STRING;

address\_aclr\_a : STRING;

outdata\_aclr\_a : STRING;

width\_byteena\_a : NATURAL;

init\_file : STRING;

lpm\_hint : STRING;

lpm\_type : STRING );

PORT ( clock0 : IN STD\_LOGIC ; --altsyncram元件接口声明

address\_a: IN STD\_LOGIC\_VECTOR (5 DOWNTO 0);

q\_a : OUT STD\_LOGIC\_VECTOR (7 DOWNTO 0) );

END COMPONENT;

BEGIN

q <= sub\_wire0(7 DOWNTO 0);

altsyncram\_component : altsyncram

GENERIC MAP ( intended\_device\_family => "Cyclone", --参数传递映射

width\_a => 8, --数据线宽度8

widthad\_a => 6, --地址线宽度6

numwords\_a => 64, --数据数量64

operation\_mode => "ROM", --LPM模式ROM

outdata\_reg\_a => "UNREGISTERED", --输出无锁存

address\_aclr\_a => "NONE", --无异步地址清0

outdata\_aclr\_a => "NONE", --无输出锁存异步清0

width\_byteena\_a => 1, -- byteena\_a输入口宽度1

init\_file => "D:\BaiduNetdiskWorkspace\intelFPGA\test\f3\rom.mif", --ROM初始化数据文件，此处已修改过

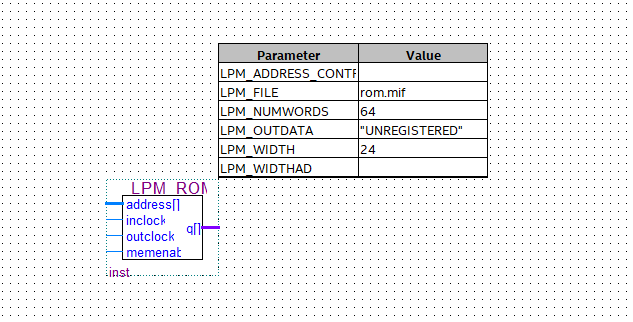
lpm\_hint => "ENABLE\_RUNTIME\_MOD=YES, INSTANCE\_NAME=NONE",

lpm\_type => "altsyncram" ) --LPM类型

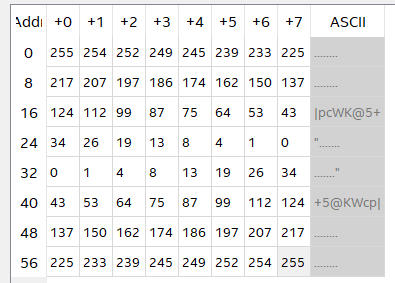
PORT MAP (clock0 => inclock, address\_a => address,q\_a => sub\_wire0 );

END SYN;

必要文件LPM\_ROM.bdf

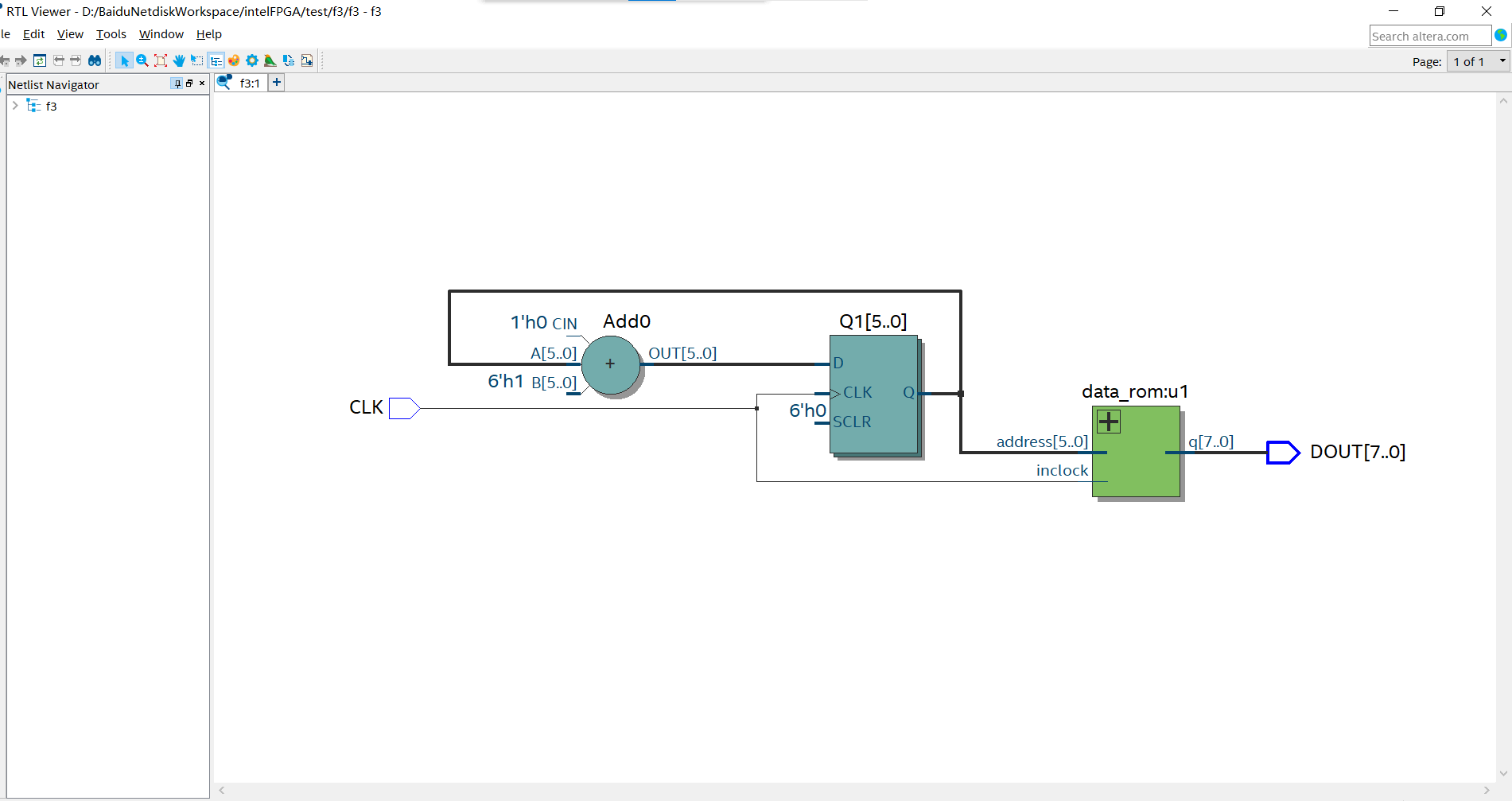


rom.mif

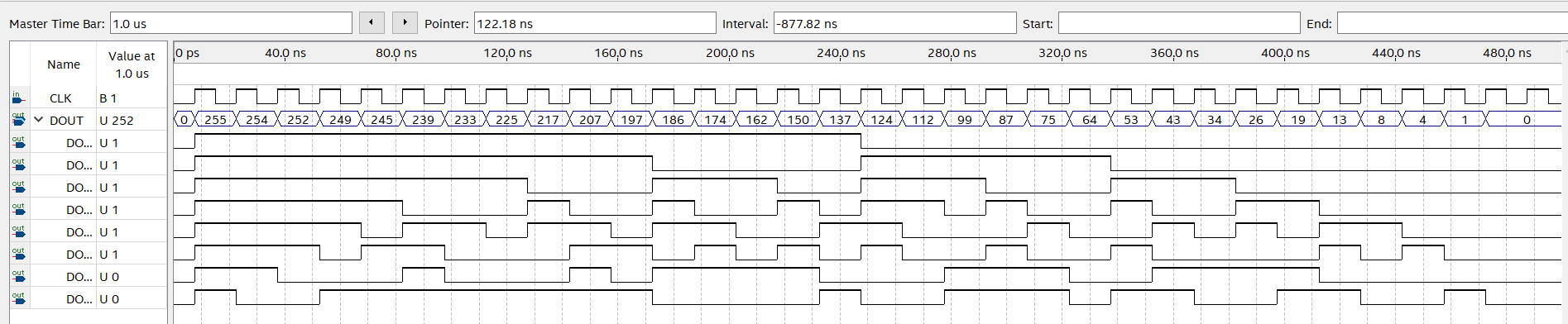


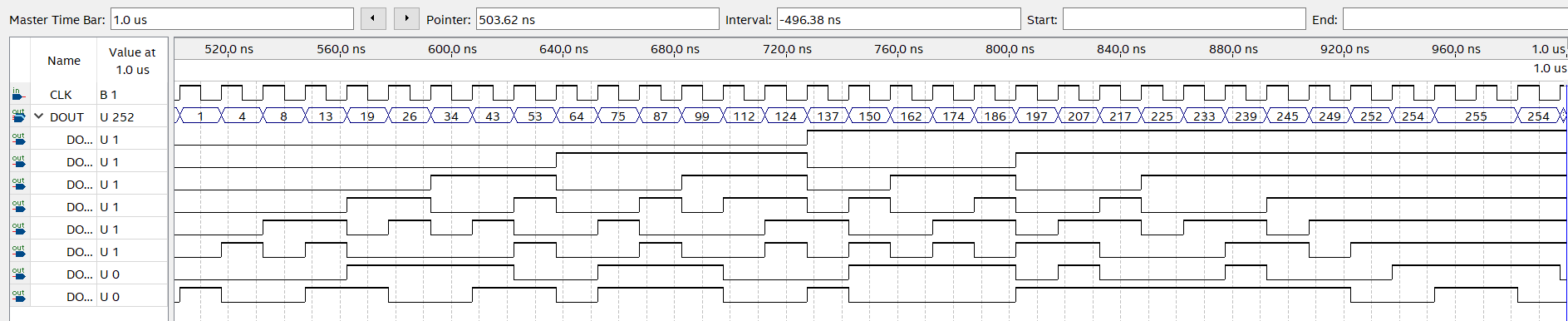
四、实验过程

元件图



测试波形





测试：使用data\_rom模块读入手动生成的数据rom.mif,给CLK输入端输入周期为50ns的时钟信号，如图成功生成正弦信号的仿真波形。

实验结果：由rom.mif数据文件可知，从255到0再到255为一个完整的正弦波波形，图中可以看出成功进行了一个完整的正弦波波形仿真。

五、实验总结

本次实验做的是正弦信号发生器，输出仿真波形。其中最明显的感知就是本次实验的难度陡然上升，除了要写VHDL代码，还要设计其他文件，如.bdf和.mif，在写VHDL时，要注意文件的路径不能写错，否则程序会无法正确获取数据。在完成本次实验后，我对Quartus Ⅱ中各类文件之间的关系理解更加清晰了，比如.bdf为格式文件，.mif为数据文件。实验虽然做了很久，但我并没有感觉到枯燥，我喜欢这种遇到困难然后自己寻找解决办法最终攻克困难的过程。