实验报告

实验题目：IP模块的调用　　姓名：牟真伟　　学号：PB20051061

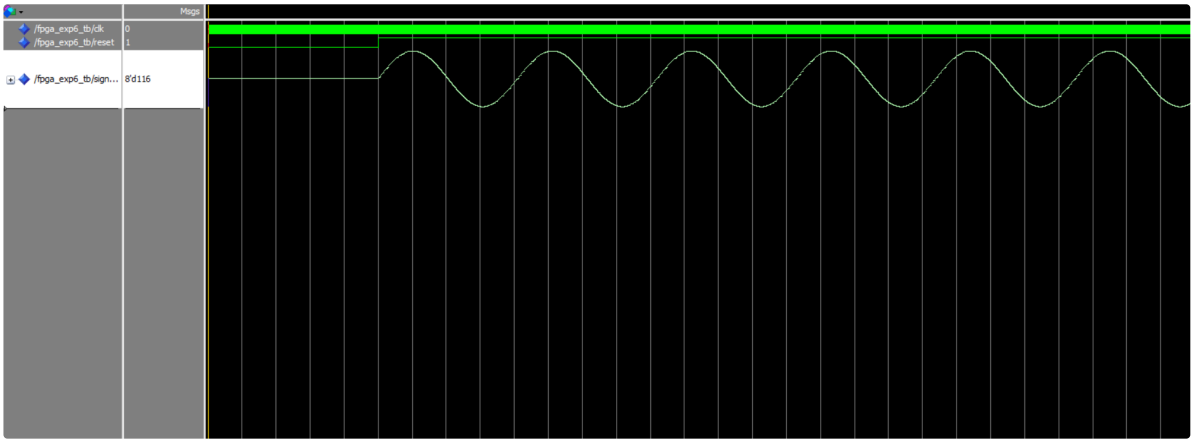
实验内容

利用宏功能模块调用（***MegaWizard***）产生一个存储器模块***mystorage***，并向其中填入数据，使得存储器的输出为正弦波，并使用Signaltap II捕获和显示正弦波信号。

设计分析

利用quartus的宏功能模块调用（***MegaWizard***）调用IP产生一个存储器模块***mystorage***，存储器模块为1k x 8bit大小，地址为10位，输出位8位，利用matlab产生存储器数据，用0-255分别表示正弦波从最小值到最大值，将存储器数据装入存储器模块***mystorage***，用顶层实体调用改模块，存储器模块地址输入位10进制计数器输出，地址每个一个时钟周期加1，输出端利用Signaltap II捕获和显示正弦波信号。

仿真结果记录



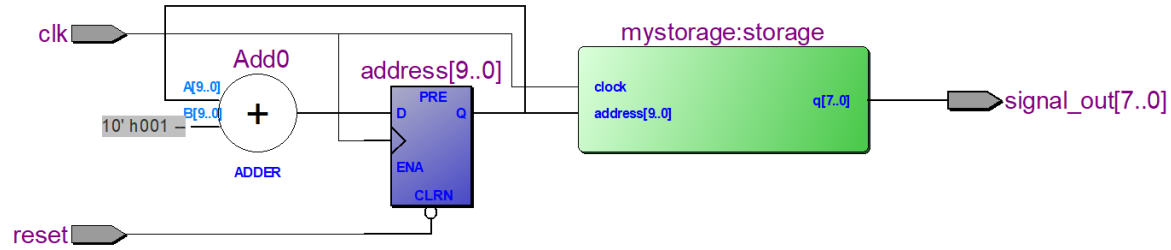
复位信号为低电平有效，当复位信号有效时，存储器输出为0地址的值128，复位信号无效时，存储器地址循环递增，信号输出为正弦波。

中间结果记录

全编译后资源占用情况：

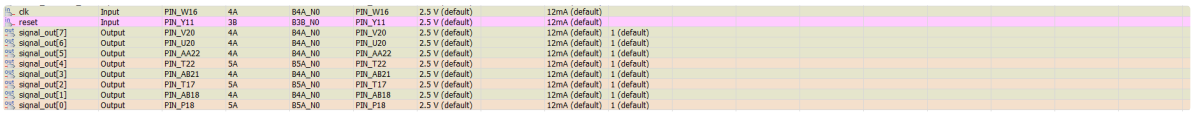
Flow Summary	
Flow Status	Successful - Sun Nov 13 18:53:35 2022
Quartus II 64-Bit Version	13.0.1 Build 232 06/12/2013 SP 1 SJ Full Version
Revision Name	FPGA_EXP6_mzw
Top-level Entity Name	FPGA_EXP6_mzw
Family	Cyclone V
Device	5CEFA2F23C8
Timing Models	Final
Logic utilization (in ALMs)	309 / 9,430 (3 %)
Total registers	644
Total pins	10 / 224 (4 %)
Total virtual pins	0
Total block memory bits	86,016 / 1,802,240 (5 %)
Total DSP Blocks	0 / 25 (0 %)
Total HSSI RX PCSs	0
Total HSSI PMA RX Deserializers	0
Total HSSI PMA RX ATT Deserializers	0
Total HSSI TX PCSs	0
Total HSSI PMA TX Serializers	0
Total HSSI PMA TX ATT Serializers	0
Total PLLs	0 / 4 (0 %)
Total DLLs	0 / 4 (0 %)

电路总RTL结构图：



FPGA验证结果记录

实验管脚约束情况如下：



clk为50Mhz时钟信号，reset为DIP0复位信号，低电平有效，由于实验箱上没有DA转换器，输出端采用Signaltap II捕获和显示正弦波信号。

- Signaltap II捕获和显示正弦波信号

  复位信号有效时

  复位信号无效时

实验总结

本次实验通过调用IP核产生一个存储器模块，并用该存储器模块存储正弦波数据，输出正弦波信号。学习了如何调用厂商提供的IP核和如何调用Signaltap II捕获和显示正弦波信号。

VHDL源代码

```
-- 顶层实体
library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_unsigned.all;

entity FPGA_EXP6_mzw is
    port(
        clk :IN std_logic;
        reset :IN std_logic;
        signal_out :OUT std_logic_vector(7 downto 0)
    );
end FPGA_EXP6_mzw;

architecture arch_FPGA_EXP6 of FPGA_EXP6_mzw is
    component mystorage port(
        address     : IN STD_LOGIC_VECTOR (9 DOWNTO 0);
        clock       : IN STD_LOGIC;
        q       : OUT STD_LOGIC_VECTOR (7 DOWNTO 0)
    );
    end component;
    signal address :STD_LOGIC_VECTOR (9 DOWNTO 0):="0000000000";


begin
    storage: mystorage port map(address,clk,signal_out);

    process(clk,reset)
    begin
        if(reset = '0') then
            address <= "0000000000";
        else
```

```
            if(clk'event and clk = '1') then
                address <= address + 1;
            end if;
        end if;
    end process;

end arch_FPGA_EXP6;

-- 仿真 test bench
library ieee;
use ieee.std_logic_1164.all;

entity FPGA_EXP6_tb is
end FPGA_EXP6_tb;

architecture arch_FPGA_EXP6_tb of FPGA_EXP6_tb is
    component FPGA_EXP6_mzw port(
        clk :IN std_logic;
        reset :IN std_logic;
        signal_out :OUT std_logic_vector(7 downto 0)
    );
    end component;
    signal clk :std_logic;
    signal reset :std_logic;
    signal signal_out :std_logic_vector(7 downto 0);
begin
    fpga_exp6 :FPGA_EXP6_mzw port map(clk,reset,signal_out);

    process
    begin
        clk <= '0';
        wait for 20 ns;
        clk <= '1';
        wait for 20 ns;
    end process;

    process
    begin
        reset <= '0';
        wait for 50000 ns;
        reset <= '1';
        wait for 500000 ns;
    end process;
end arch_FPGA_EXP6_tb;
```