

IP核使用

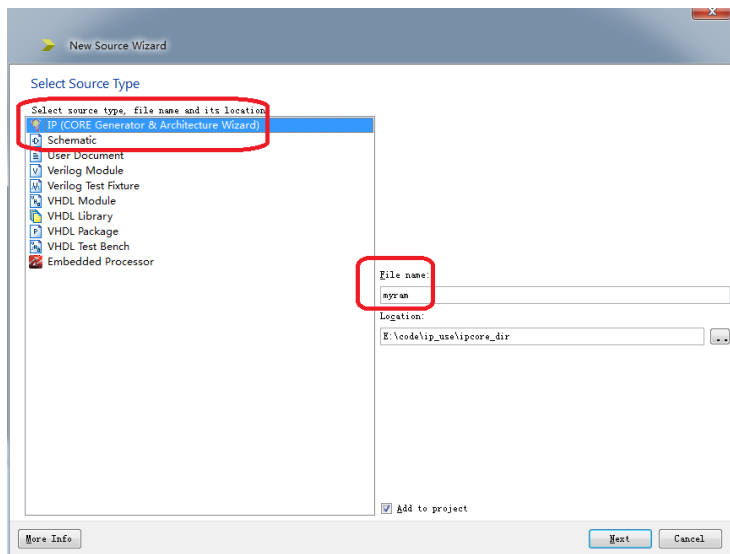
目的

- 掌握IP核的使用方法
- 熟悉RAM的接口及时序
- 能熟练设计时序逻辑电路

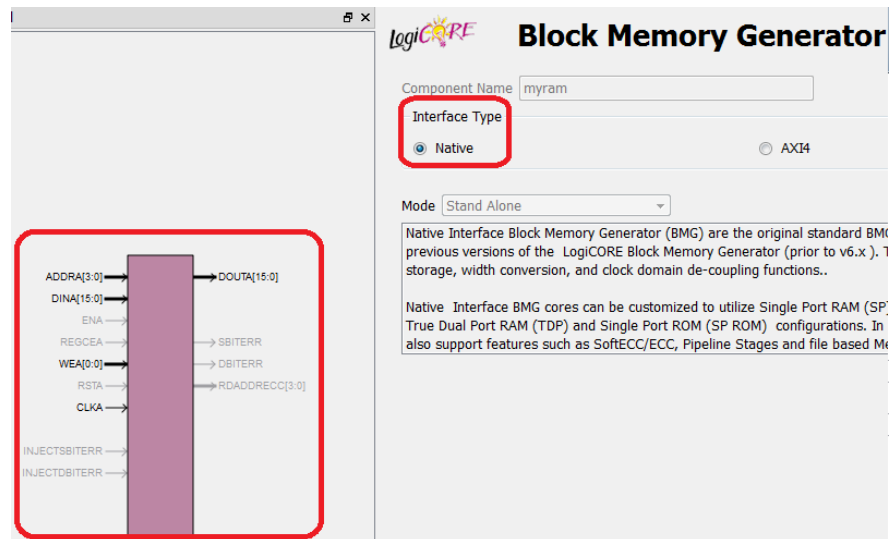
- **FPGA**厂商在其开发工具中提供了许多通用的功能模块，称为“**IP核**”（知识产权核），用户可以像调用普通模块一样来调用它们。
- 本实验中，我们以时序逻辑电路中较为常见的“简单双端口**RAM**”为例来学习**IP核**的使用方法

RAM例化

1



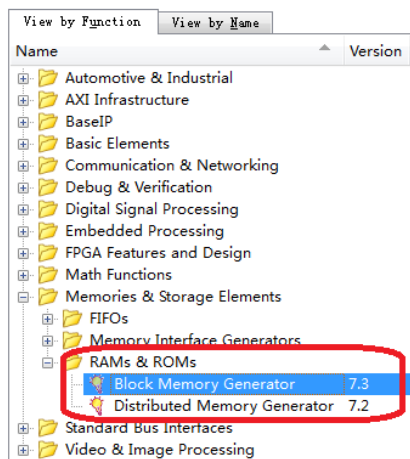
3



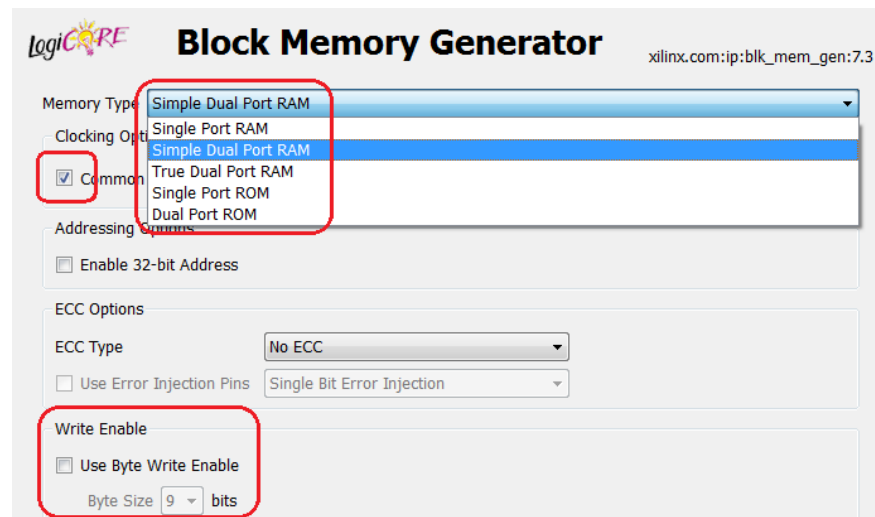
2

Select IP

Create Coregen or Architecture Wizard IP Core.



4



RAM例化

- RAM的读写位宽可设置
- 输出数据寄存可设置
- 可加载初始化文件

Port B

- ☒ Register Port B Output of Memory Primitives
- ☒ Register Port B Output of Memory Core
- ☐ Register Port B Output of SoftECC logic
- ☐ Use REGCEB Pin (separate enable pin for Port B output registers)

Pipeline Stages within Mux: 0 Mux Size: 1x1

Memory Initialization

- ☒ Load Init File

Coe File: no_coe_file_loaded [Browse] [Show]

LogiCORE Block Memory Generator

Port A Options

Memory Size

Write Width: 32 Range: 1..4608

Write Depth: 64 Range: 2..9011200

Operating Mode

☒ Write First

☐ Read First

☐ No Change

Enable

☒ Always Enabled

☐ Use ENA Pin

Port B Options

Memory Size

Read Width: 32

Read Depth: 64

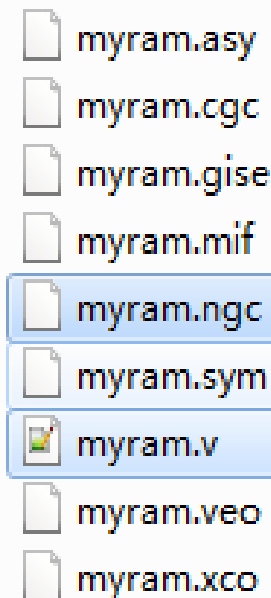
coe文件格式

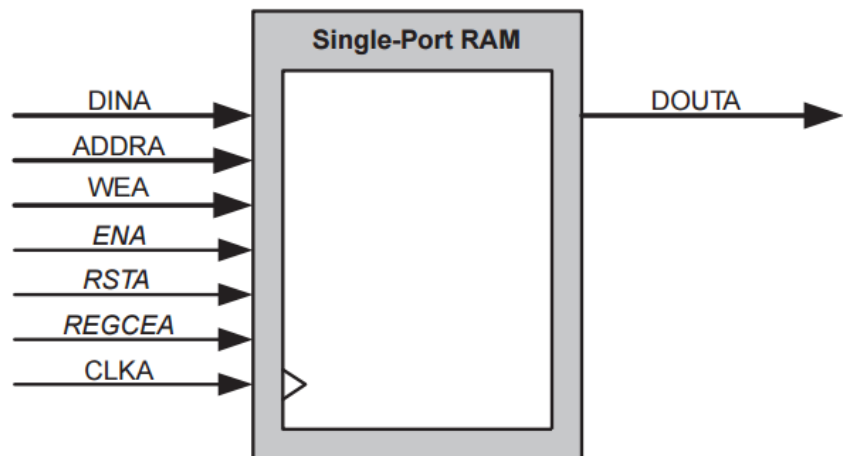
```
memory_initialization_radix = 16;  
memory_initialization_vector =  
1,2,3,4,5,6,7,8,9,a,b,c,d,e,f,10,11,12,13,14,15,16,17  
,18,19,1a,1b,1c,1d,1e,1f;
```

- 第一行表示数据格式，16代表16进制，10代表10进制
- 第二行则是初始化向量，是RAM的初始化值

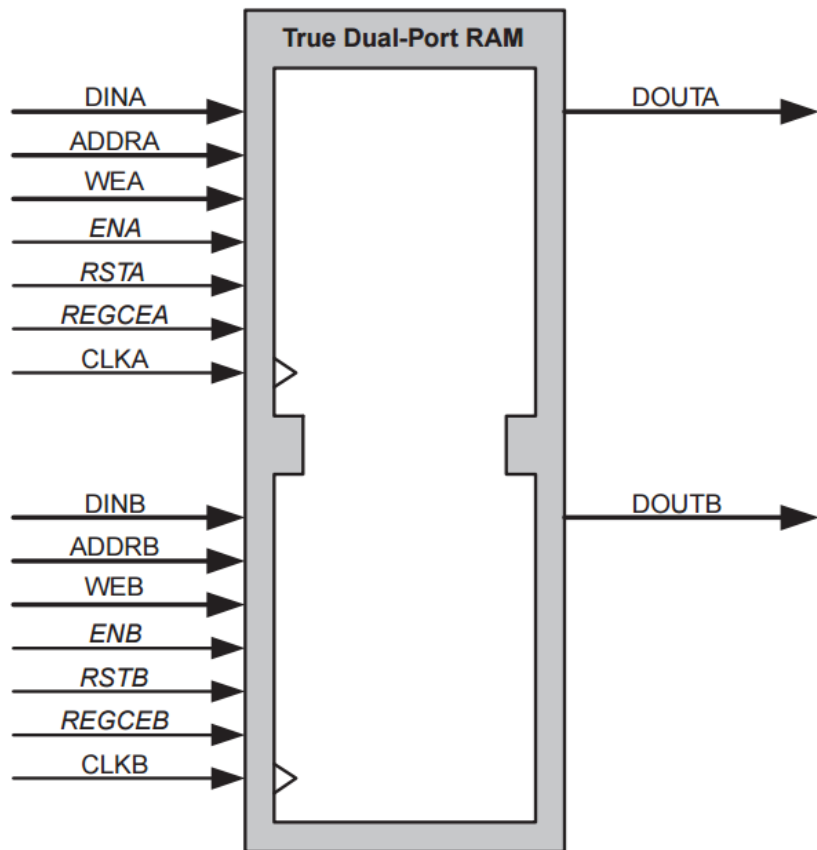
IP核使用

- IP核生成好之后，在工程目录下会产生一个.v文件，给出了IP核的端口信号，上层模块可根据该文件调用IP核
- 如在**其他工程**中使用该IP，只需将.v、.ngc两个文件加入工程，即可像普通模块那样使用

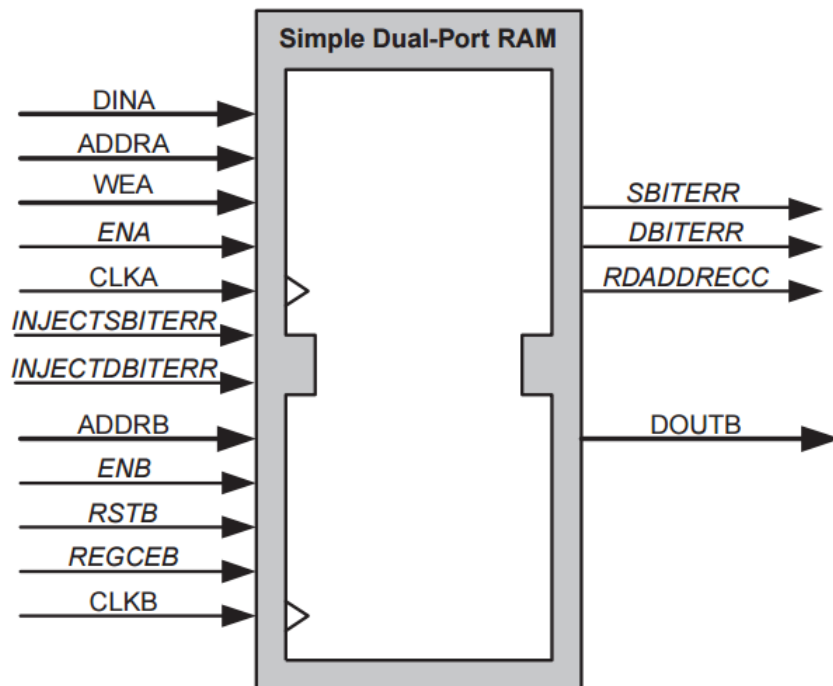




单端口



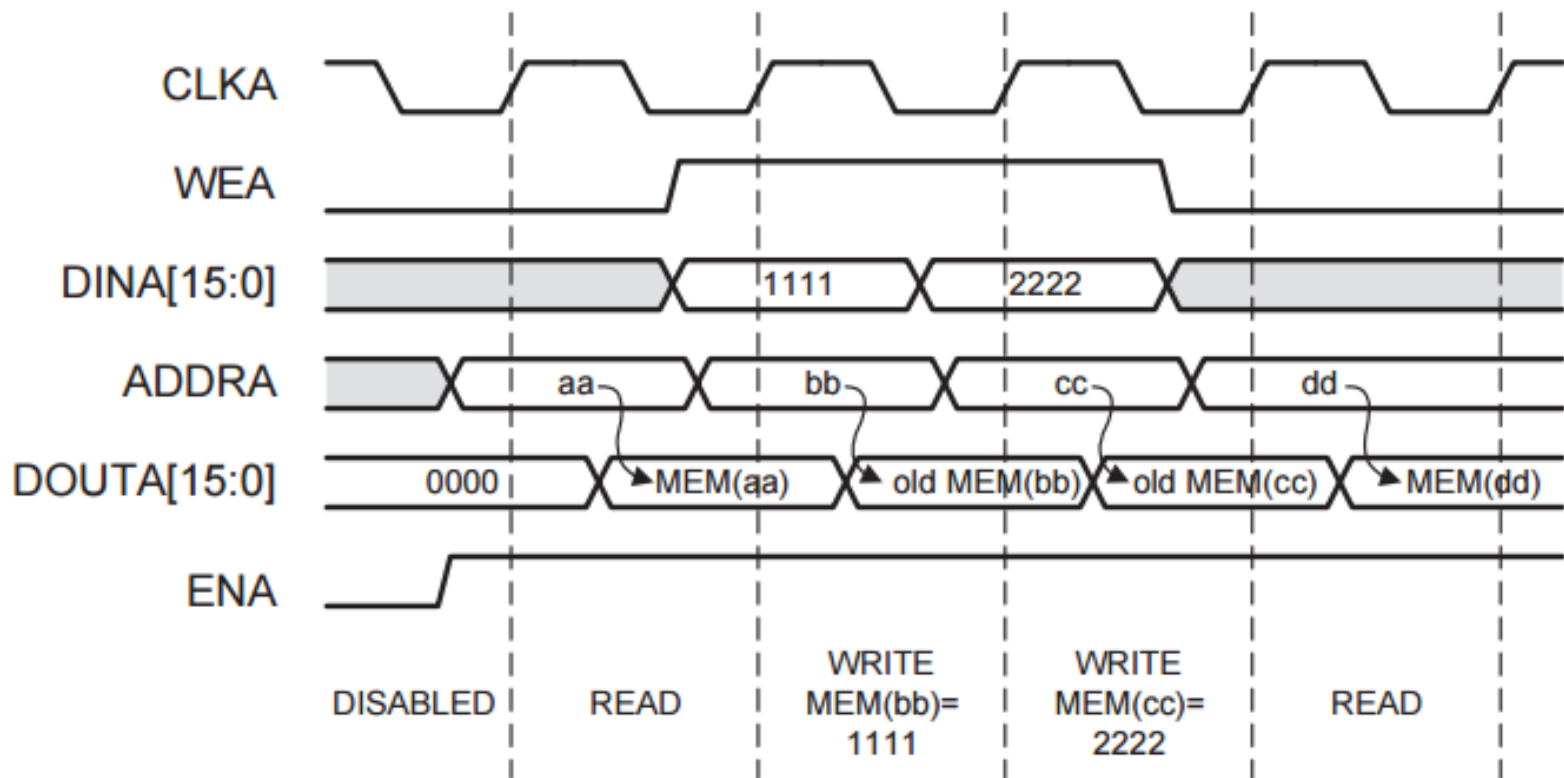
真双端口RAM



简单双端口

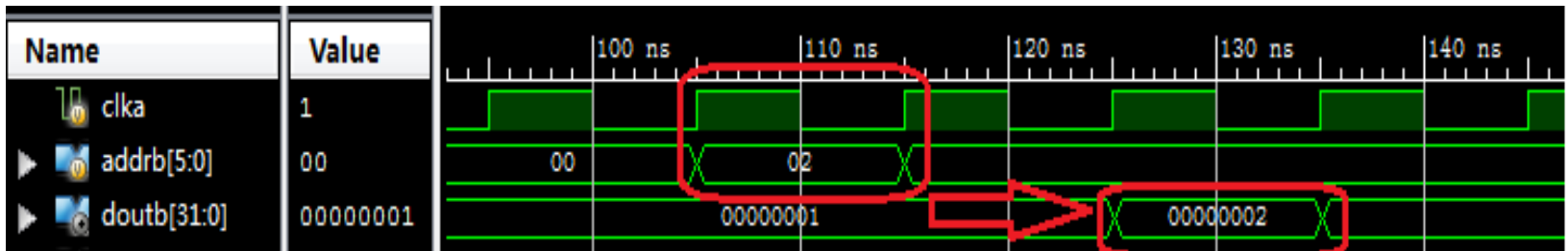
RAM读写时序

- 读优先模式
 - 同时对一个地址进行读写时，输出旧数据

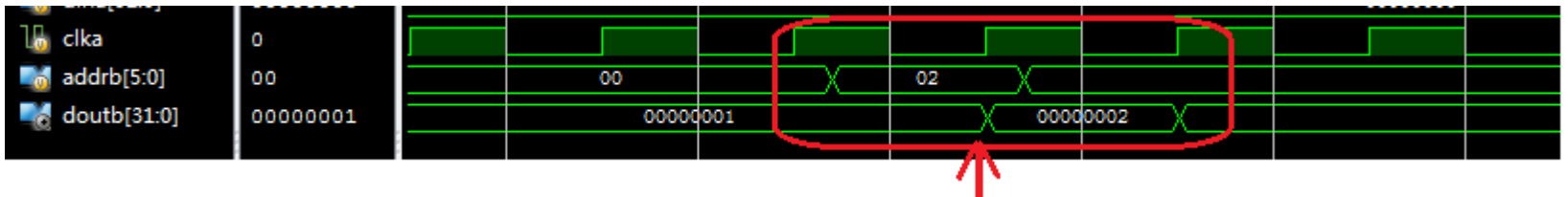


RAM读时序仿真

- 对端口B数据输出寄存



- 不寄存(时钟的上升沿之后输出数据)



内容

- 例化一个数据位宽为32bit，深度为32的简单双端口ram
- 读写端口采用同一时钟
- 将ram初始化为1,1,0,0,0,0.....
- 利用有限状态机实现斐波那契数列的计算
 - 读出0号地址、读出1号地址，将两个数据相加，写入2号地址
 - 读出1号地址、读出2号地址，将两个数据相加，写入3号地址
 - 读出2号地址、读出3号地址，将两个数据相加，写入4号地址
 -
 - 读出29号地址、读出30号地址，将两个数据相加，写入31号地址
 - 停止

要求

- 正确理解RAM读写接口时序
- 设计出正确的有限状态机
- 画出电路设计框图
- 仿真结果正确
 - RAM的所有接口信号波形
 - 状态机跳转波形
 - RAM内容结果
- 编写实验报告
 - 实验完成后一周之内上传至FTP

实验报告模板

- 实验报告命名格式（文件名）：
 - lab06_学号.pdf
- 内容包括但不限于：
 - 实验目的
 - 实验内容
 - 实验结果
 - 实验分析
 - 意见建议
 - 附录:原理图、源代码