

实验 6-存储器实验

一、 实验准备

- 1.1 装有 Vivado2015.4 的电脑一台：本实验不对 Vivado 环境有硬性要求，但涉及 Xilinx 库中 IP 的实验，在不同版本环境下无法兼容，且低版本 Vivado 无法运行高版本生成的项目，为方便实验检查，应当尽量使用实验要求版本
- 1.2 熟悉 Vivado 的 IDE 环境，并能够使用其进行仿真、综合；（见第一次实验课件）
- 1.3 熟悉开发板（见第一次实验课件）

二、 实验目的

- 2.1 了解随机存取存储器 RAM 的原理；
- 2.2 掌握调用 Xilinx 库 IP(Block Memory Generator)实例化 RAM 的方法；

三、 实验设备

PC 机一台，Basys3 开发板，Xilinx Vivado 开发套件。

四、 实验任务

4.1 存储器 IP 实例化实验

本次实验使用 Vivado 的 Block Memory Generator 模拟数据在存储器中的存取过程。实验使用单端口 ROM。初始化 ROM 存储器中的内容，通过开关选择相应的地址，将对应的存储器中内容读出来，并通过七段数码管显示。实验原理如图 1 所示：

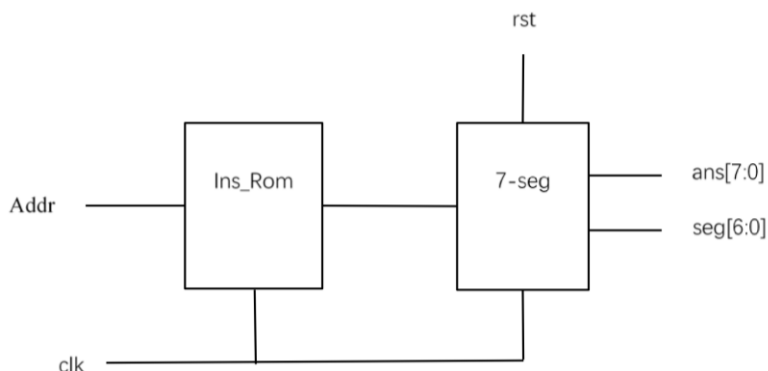


图 1

4.1.1 实验要求

1. 使用 Block Memory Generator 生成单端口 ROM，并将指令 coe 文件加载；
2. 将 ROM 中对应的 32 位指令取出并送往 7-seg 数码管显示。
3. 将地址线接到拨码开关上，验证 ROM 读取的值。

五、 实验步骤

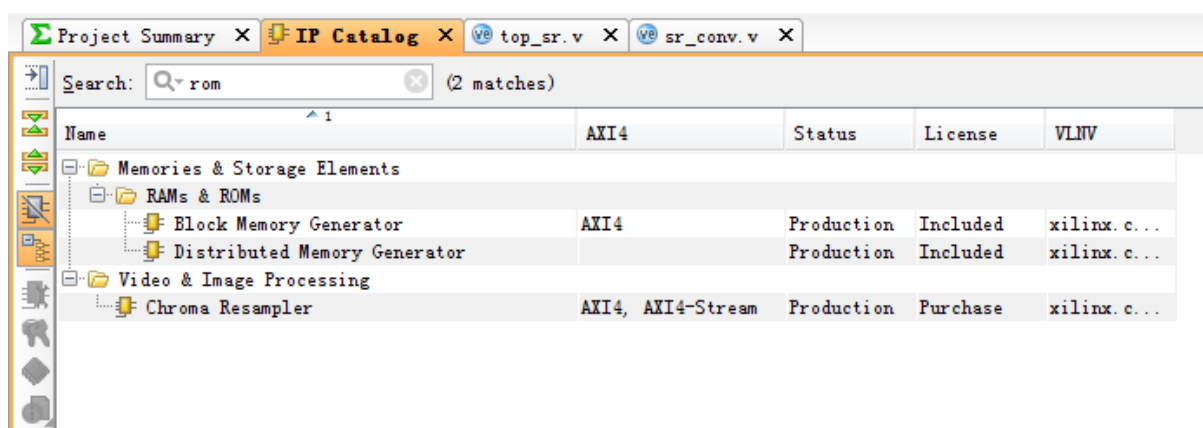
5.1 添加调用 rom IP 核

在 IP Catalog > Memories & Storage Elements > RAMs & ROMs> Block Memory Generator

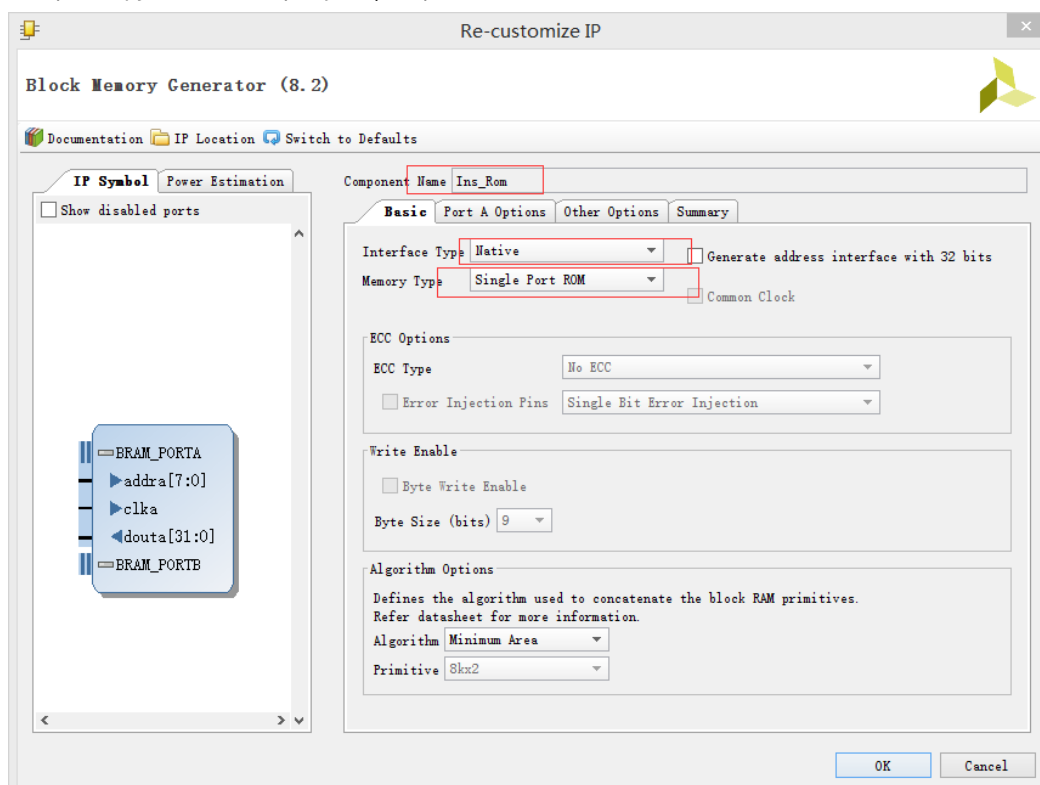
目录下，找到系统自带的 rom 核并添加。点击 Flow Navigator 中的 IP Catalog，打开窗口添加 IP 核。



Block Memory 为块存储设备，这里需要的是 Block Memory Generator

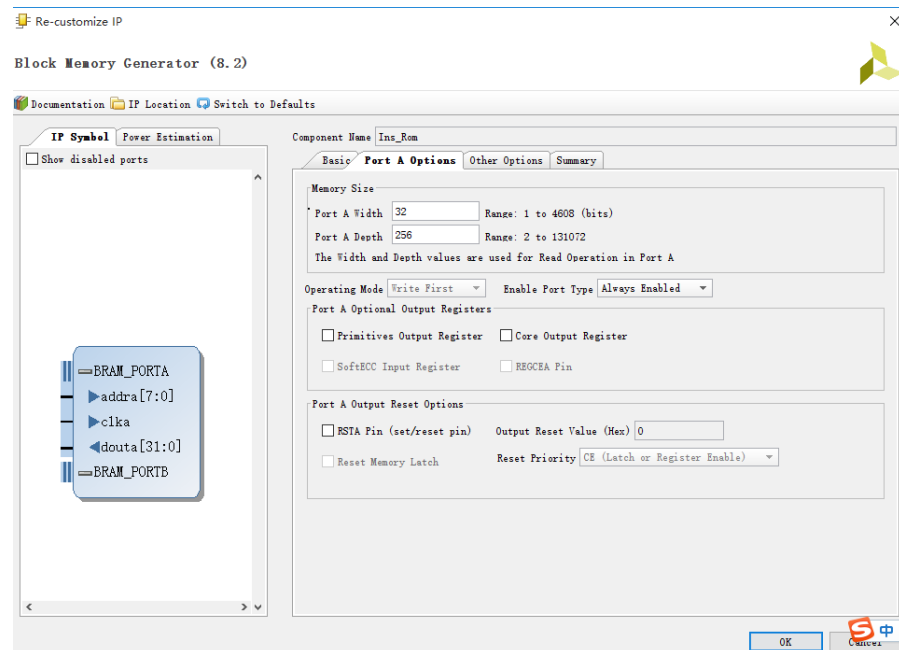


双击之后得到 Basic 一栏的基本属性：

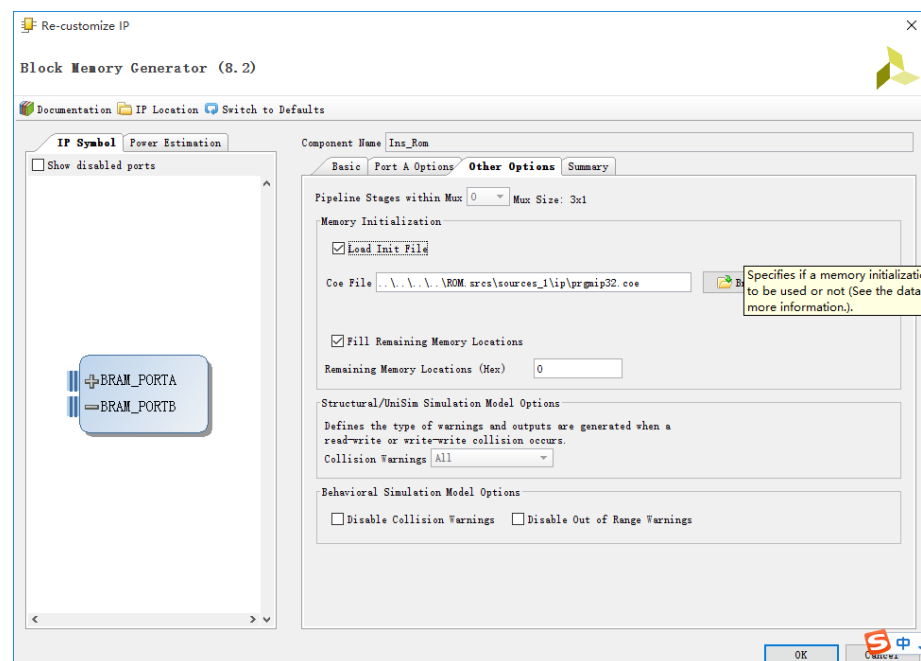


Component Name 命名为：Ins_Rom。Native、单端口、ROM 不要 ECC 校验、最小面积算

法。接着设置 Port A Option 的基本属性



数据宽度 32 位，深度要 256 个数据，地址线 8 根。接着设置 option 选项属性：



在 LOAD INIT FILE 项上打勾，点击 LOAD FILE，要你选择一个文件。这个文件就是你要放在 ROM 中的数据，文件命名 prgmip16.coe，文本文件。你可以自己建一个文本文件，然后把后缀名改为.COE 即可。(文件中放入自己的学号或你想显示任意内容)。RST & Initialization：使用 coe 文件可以给 ROM 输入初值，格式如下：

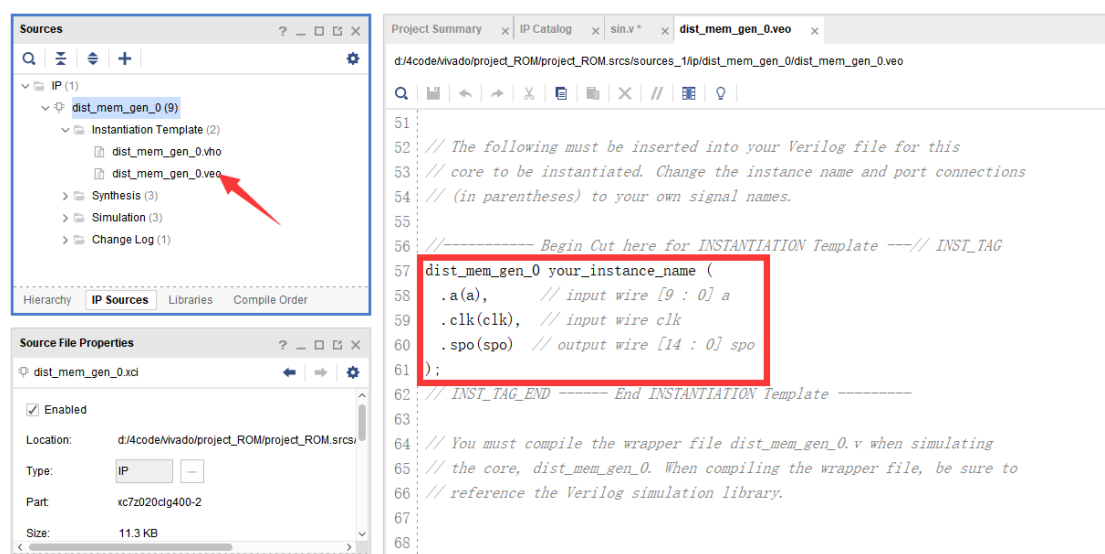
```
memory_initialization_radix=16;  
memory_initialization_vector=  
0108,0304,  
0506,0907,  
0001,0004,
```

0005,0009,
0007,0005,
0000,0005,
8888,8888,
9999,9999,
Aaaa,aaaa,
Bbbb,bbbb,
1234,5678,
12ab,cdef;

可以直接用文本编辑器写好 COE 文件，radix=16 进制定义，这里是 16 进制，第二行照抄，第三行开始是数据（我们定义的 16 位数据，rom 数据宽度），数据间隔用逗号，最后一个数据分号。COE 文件最后会生成 MIF 文件用于初始化 ROM。点击 ok，生成 rom ip 核相应的文件。

5.2 ROM 例化文件的方法

ip 核的例化，在 source 下面有一个 IP Source 选项卡点击进去，再点击 Instantiation Template 里的.veo 文件就是 verilog 例化文件了，默认名称 blk_mem_gen_0.veo。如图所示，



把实例化代码复制到 top 文件中，在此基础上进行修改。编写好源代码后，进行综合，在 Flow Navigator 选项卡选择 synthesis，点击“run synthesis”，综合后可以在下面的 message 一栏查看综合信息，是否有错误，不要有 critical warning 和 error。