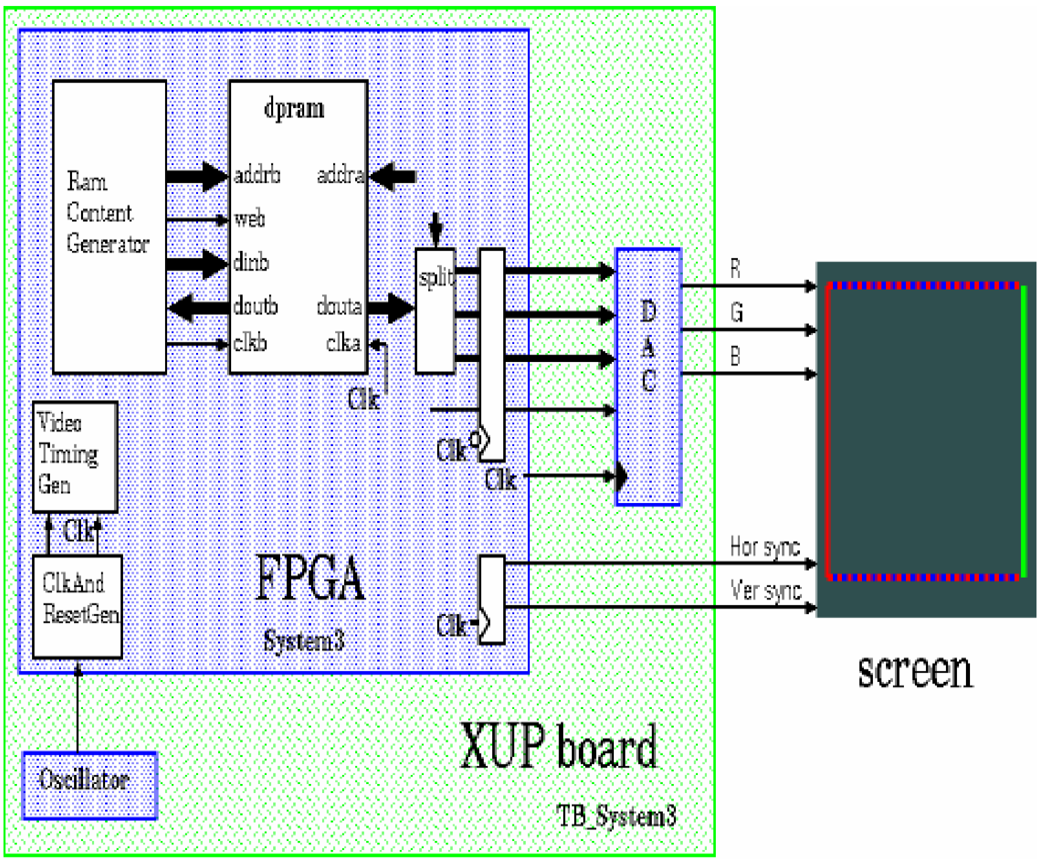
Lab-exercise

Lab6:

Generation of a dual port RAM

Introduction

In this lab, so far the RGB signals were driven directly the hardware. We can change this and read the RGB values from a RAM. We can now split the system in two parts, whereby the first subsystem writes the values to be shown on the screen in the RAM and the second subsystem reads those values from the RAM and sends them to the screen. This means that two independent systems are reading from and writing to the RAM. In the following modules we will use a processor to write values calculated by software into the RAM. The easiest solution to avoid read and write conflicts is using a dual port synchronous RAM. This is a RAM with can be read and written simultaneously on two different addresses. We will generate such a RAM in this module.



Goal

The goal of this lab is making the student realize that in a system not all modules have to be designed from scratch, but to use as many existing and verified modules as possible. After this lab the student will Have an idea about which modules can be automatically generated with the Xilinx CoreGen module generator. Know which output files CoreGen creates and what they are used for.

Knowledge background

None

Classification

Difficulty level (from 1 tot 5): 1
Time needed (without support): 0.5 hours

Input

You'll find the following folders/files for module4a:

- coregen/ram_vga_init.coe : file containing initialization values for the RAM

RAM parameters

1. Size of the RAM.
The Xilinx VirtexII FPGAs contains a limited amount of block RAM, each 18kbit in size(including the parity). In the XC2VP30 there are 136 of these block RAMs. This hence means you can use136x18kbit – 306kB block RAM. To drive a VGA screen with eight possible colors per pixel (a pixel is represented by 3 bits) we need 3x 640x480bits. That's 3x37.5kB = 112.5 kB for the video RAM, leaving 306 – 112.5 =193.5 kB we can use to implement other functionality later.
2. RAM Organisation
In the following modules we will connect the RAM to a processor via a 32 bits wide bus. Per word we can hence store the color information of 10 pixels. We are not going to use the 2 remaining bits in each word. For our 640x480 pixels we hence need 307200/10 (=30720) 32 bits wide memory spaces.This comes down to a minimum address width of 15 bits (2**15=32768).

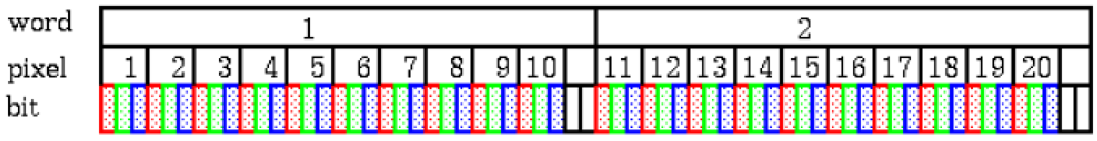


Figure 2

3. IO of the RAM
A side: is read and sent to the screen
clka : in
addra : in 15 bit
douta : out 32 bit
B side: is written and read by something that decides what has to be put on the screen.
clkb : in
addrb : in 15 bit

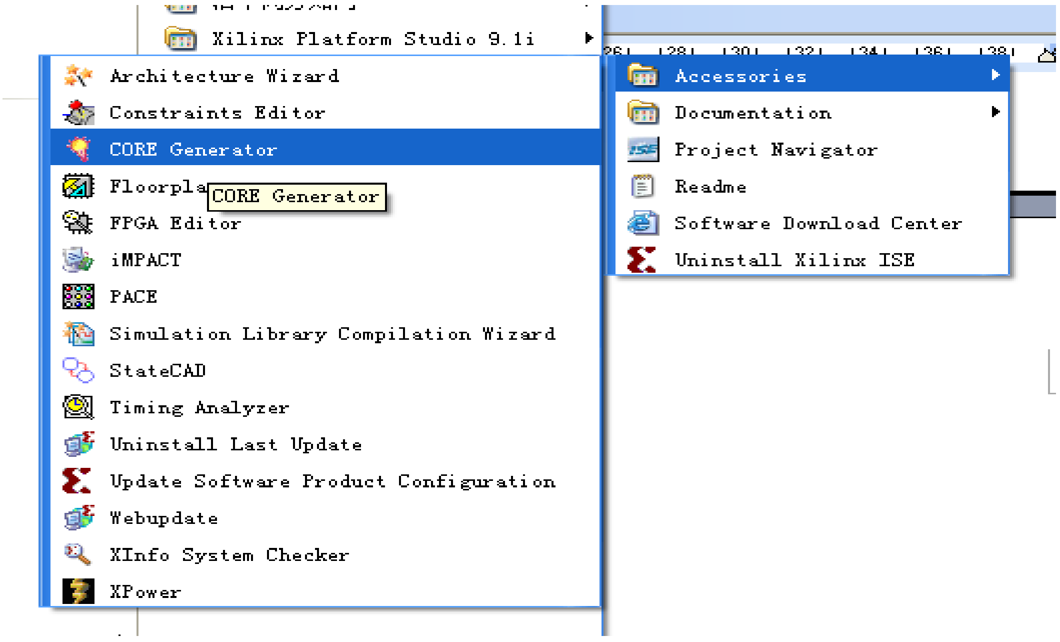
dinb : in 32 bit
web: in -- write enable
doutb: out 32 bit
Question : Do you know why the PowerPC processor should be able to write to AND read from the video RAM?

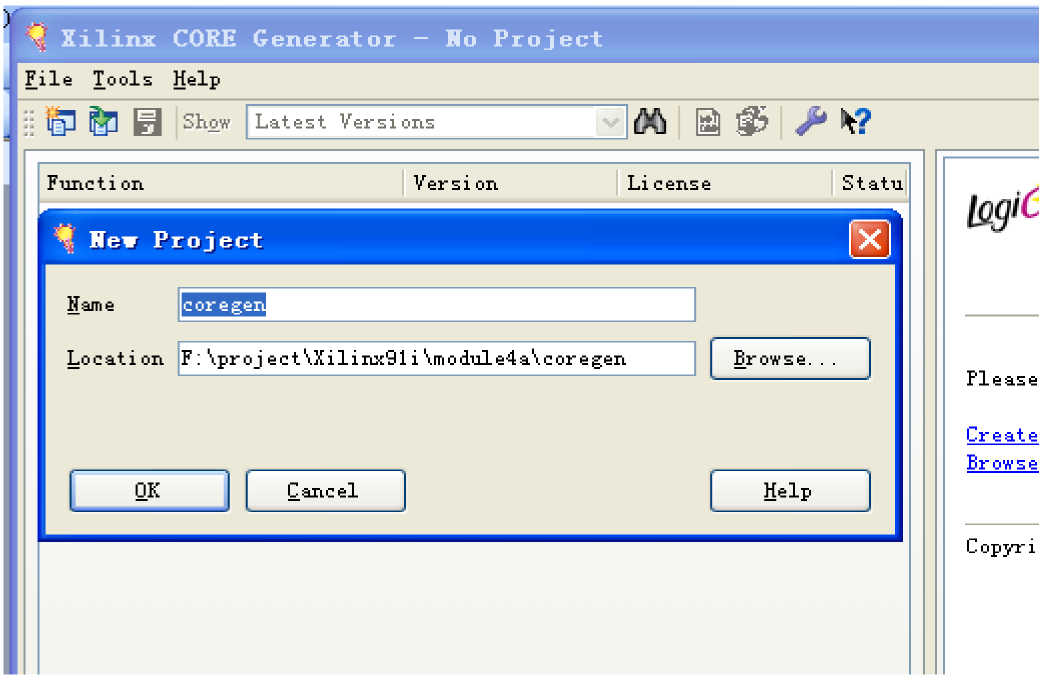
The lab

Use the Xilinx "CORE Generator" tool to generate the above mentioned RAM. You'll find the tool under the Programs>Xilinx ISE >Accessories menu.

1. Start a new project in the ....module4a\coregen directory.

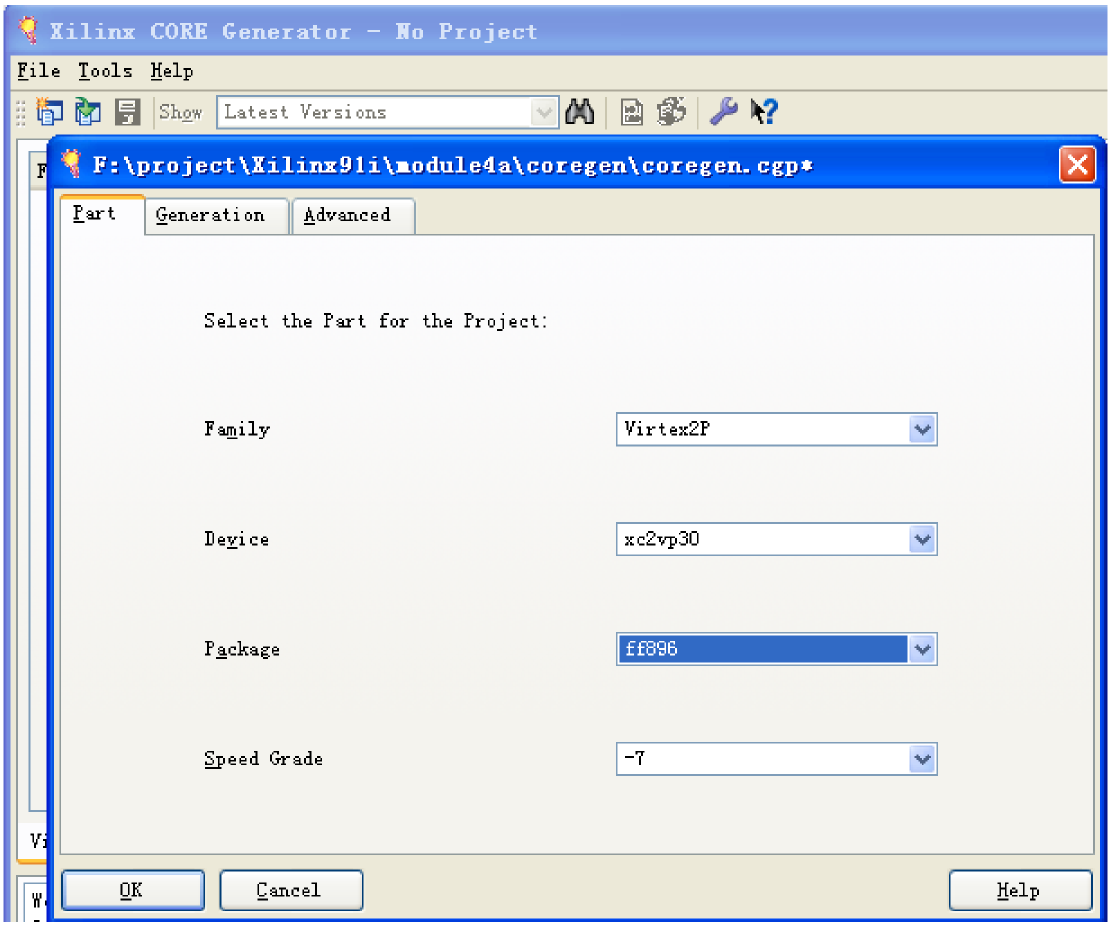
Make sure your Target Architecture is Virtex2P. Don't change the other options.





Name: coregen
Location: F:\project\Xilinx91i\module4a\coregen

Click on Ok


Xilinx CORE Generator - No Project
File Tools Help
Show Latest Versions
F:\project\Xilinx91i\module4a\coregen\coregen.cgp*
Part Generation Advanced
OK Cancel Help

Select the Part for the Project:

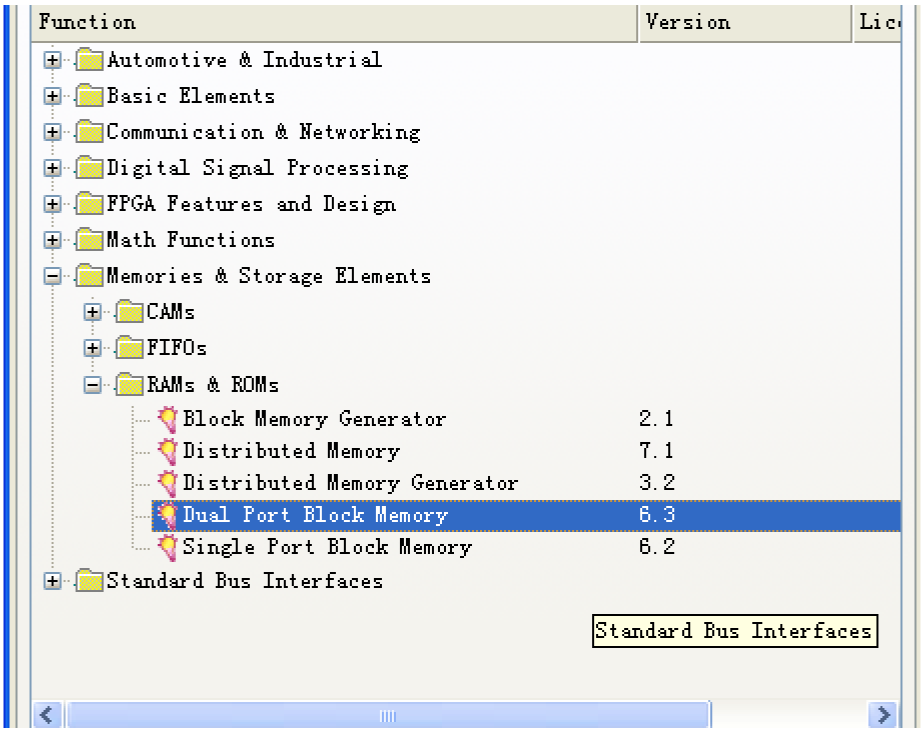
Family Virtex2P

Device xc2vp30

Package ff896

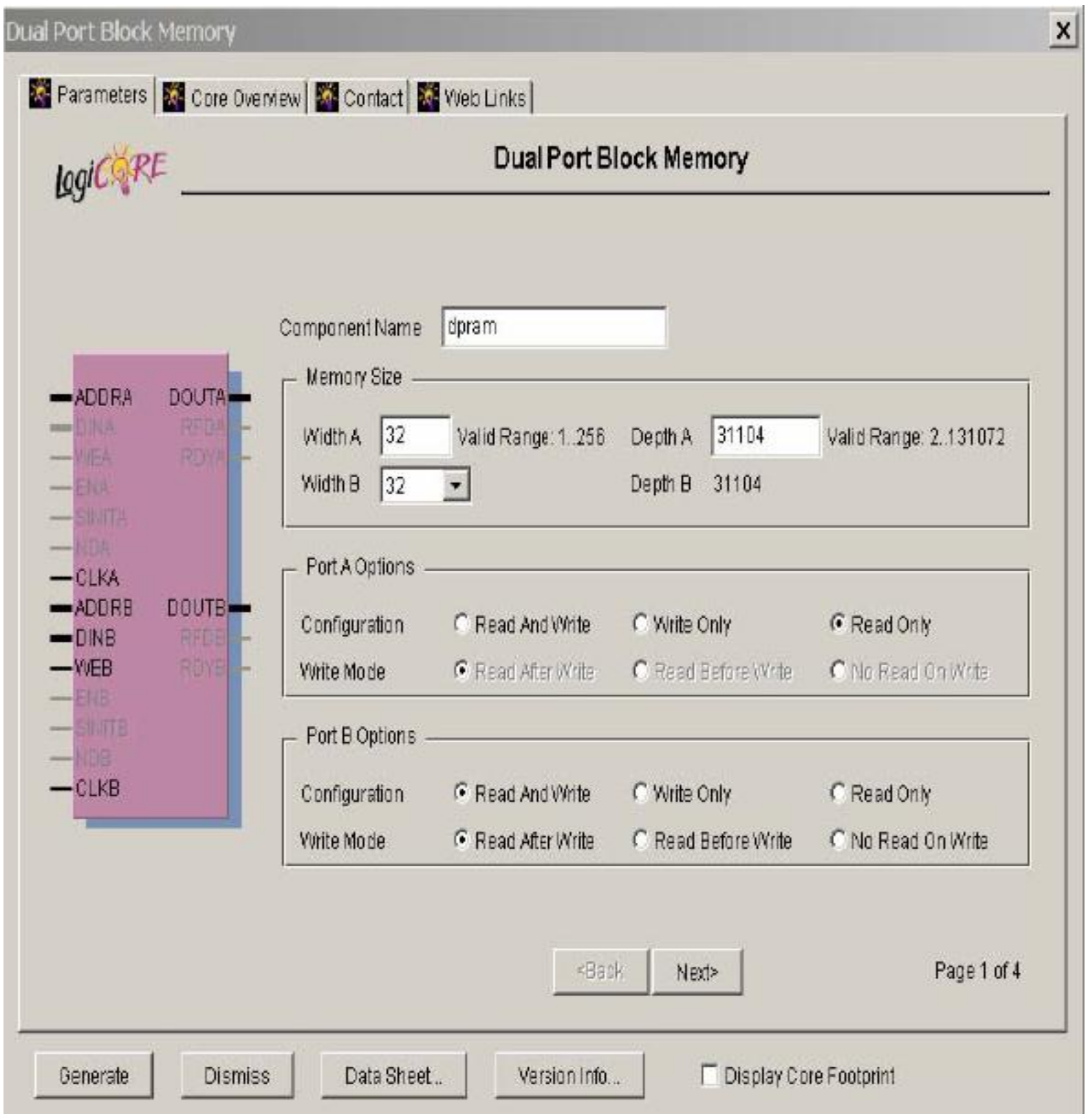
Speed Grade -7

Choose the right device. And click on OK.



Function	Version
Automotive & Industrial	
Basic Elements	
Communication & Networking	
Digital Signal Processing	
FPGA Features and Design	
Math Functions	
Memories & Storage Elements	
CAMs	
FIFOs	
RAMs & ROMs	
Block Memory Generator	2.1
Distributed Memory	7.1
Distributed Memory Generator	3.2
Dual Port Block Memory	6.3
Single Port Block Memory	6.2
Standard Bus Interfaces	

Double click on Dual Port Block Memory.



Component Name: dpram

Memory Size

Width A: 32 Valid Range: 1..256 Depth A: 31104 Valid Range: 2..131072

Width B: 32 Depth B: 31104

Port A Options

Configuration: Read And Write, Write Only, [✓ Read Only]

Write Mode: [✓ Read After Write], Read Before Write, No Read On Write

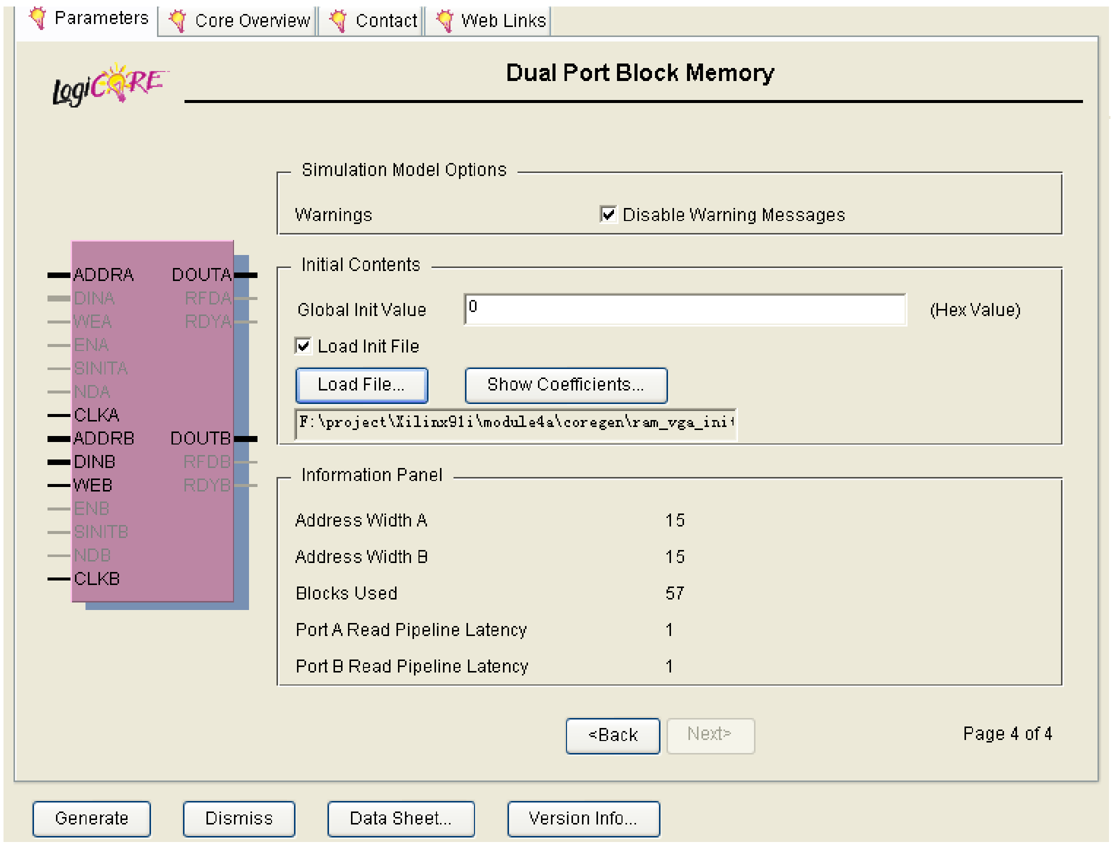
Port B Options

Configuration: [✓ Read And Write], Write Only, Read Only

Write Mode: [✓ Read After Write], Read Before Write, No Read On Write

On the next two pages (page 2 and 3) leave the defaults as they are. In the datasheet you can find the exact meaning of all these parameters. Keeping the default parameters in short means that the interface is kept as simple as possible and that both sides are triggered by the rising edge of the clock and that the control signals (web) are active high. On the last page (page 4) you select “Load Init File” and fill in ram_vga_init.coe as filename. This results in 2 files being adapted:.

1. The EDIF netlist you need to implement the RAM on the FPGA. It will also contain the initial values for the RAM.
2. The .mif file used by the VHDL model during simulation. This file also contains the intial values for the RAM.



Dual Port Block Memory

Simulation Model Options

Warnings ☑ Disable Warning Messages

Initial Contents

Global Init Value 0 (Hex Value)

☑ Load Init File

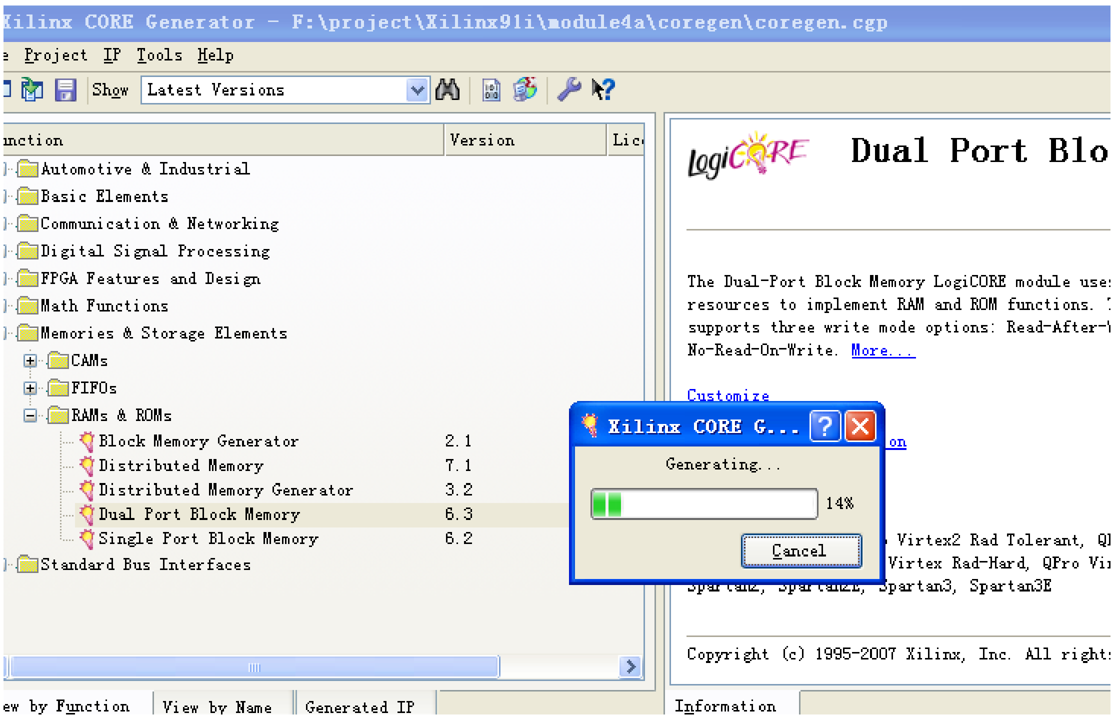
F:\project\Xilinx91i\module4a\coregen\ram_vga_init

Information Panel

Address Width A	15
Address Width B	15
Blocks Used	57
Port A Read Pipeline Latency	1
Port B Read Pipeline Latency	1

Page 4 of 4

Then click on Generate.



Xilinx CORE Generator - F:\project\Xilinx91i\module4a\coregen\coregen.cgp

Function	Version
Automotive & Industrial	
Basic Elements	
Communication & Networking	
Digital Signal Processing	
FPGA Features and Design	
Math Functions	
Memories & Storage Elements	
CAMs	
FIFOs	
RAMs & ROMs	
Block Memory Generator	2.1
Distributed Memory	7.1
Distributed Memory Generator	3.2
Dual Port Block Memory	6.3
Single Port Block Memory	6.2
Standard Bus Interfaces	

Dual Port Blo

The Dual-Port Block Memory LogiCORE module use resources to implement RAM and ROM functions. supports three write mode options: Read-After-W No-Read-On-Write. More...

Customize

Virtex2 Rad Tolerant, Q Virtex Rad-Hard, QPro Vi Spartan2, Spartan2E, Spartan3, Spartan3E

Copyright (c) 1995-2007 Xilinx, Inc. All right

Xilinx CORE G...

Generating...

14%

Wait about ten minutes. The lab is finished.