

## 软件工程专业

### 计算机组成与体系结构 实验辅导

- 1、QuartusII使用指南
- 2、实验内容以及要求
- 3、提交方式以及截止时间

### • QuartusII图形方式使用指南

## 实验用QuartusII 9.0

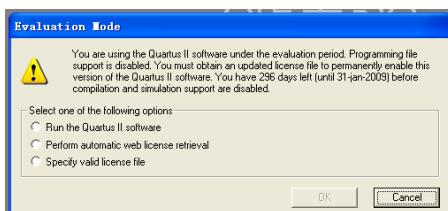
- QUARTUSII是美国Altera公司提供的可用于可编程片上系统(SOPC)开发的综合开发环境,是进行SOPC设计的基础.集成环境包括以下内容:系统级设计,嵌入式软件开发,可编程逻辑器件(PLD)设计,综合,布局和布线,验证和仿真.
- 其应用方法与设计流程对于其他流行的EDA工具的使用具有一定的典型性和一般性。

- <http://fpgasoftware.intel.com/>
- <https://www.intel.com/content/www/us/en/collections/products/fpga/software/downloads/quartus-prime.html>



## 一、准备

- 1、使用QuartusII软件之前,请确保软件能正常  
– 若启动QuartusII时看到如下界面,则说明软件尚未获得许可:

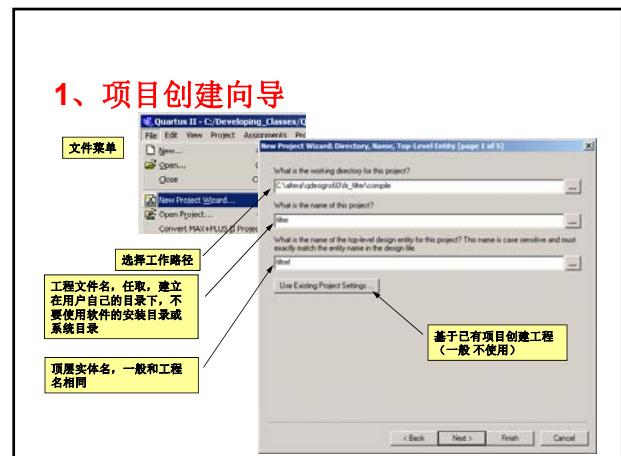
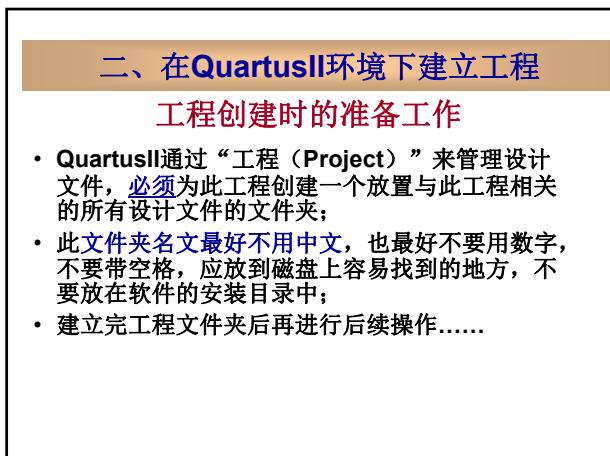
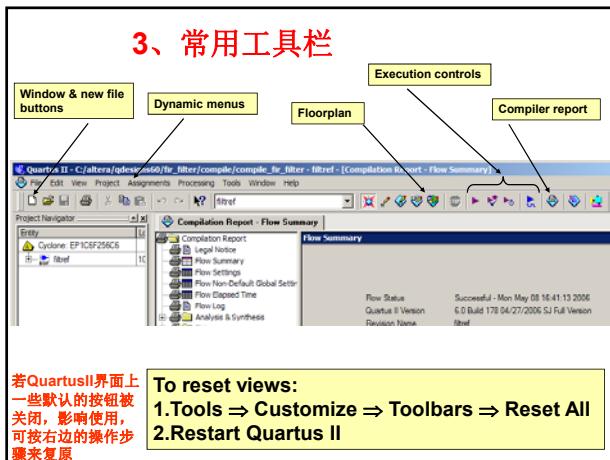
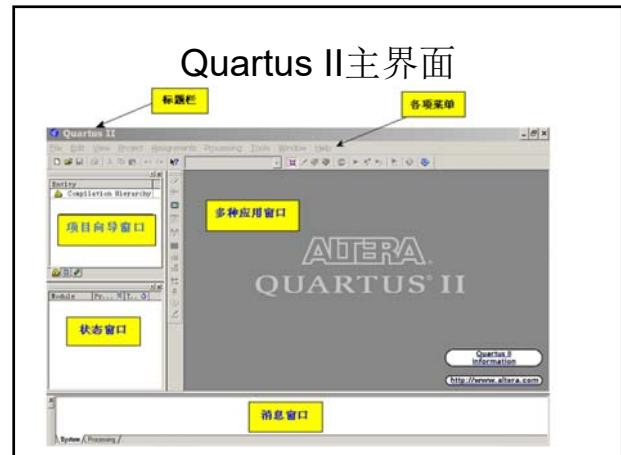


## 申请授权许可证 (license)

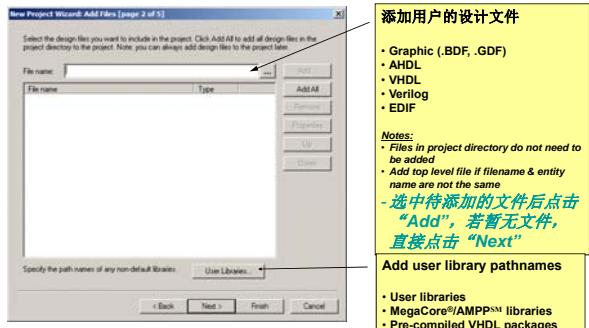
### — 准备工作：查找当前计算机的NIC — (Network Interface Card)

- ◆ 在Windos界面下, 使用命令 **开始 | 运行 | cmd ↴**,  
出现DOS命令提示符,
- ◆ 在DOS命令提示符下键入命令: **ipconfig /all**
- ◆ 在屏幕显示的结果中, 在 **physical address** 后面有一串  
**12位的16进制数**, 这就是本计算机的NIC (每2个数字之间有连字符隔开)。  
例如: **00-0F-7D-86-3E-25**。

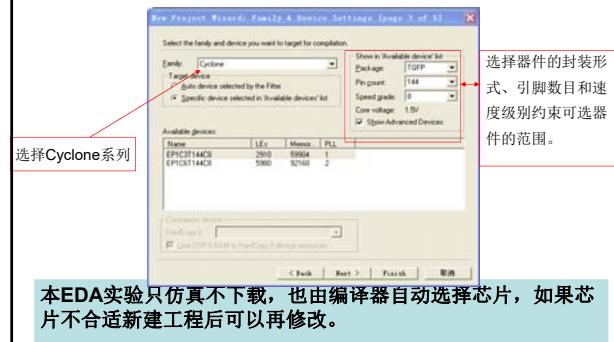
将本机D:\Altera目录下的License.Dat文件中的hostid后的内容改为MAC号即完成；  
图所示为查看本机MAC地址（实际地址）的方法。



## 2、为创建的工程添加设计文件

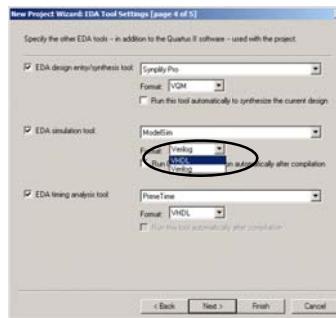


## 3、器件选择

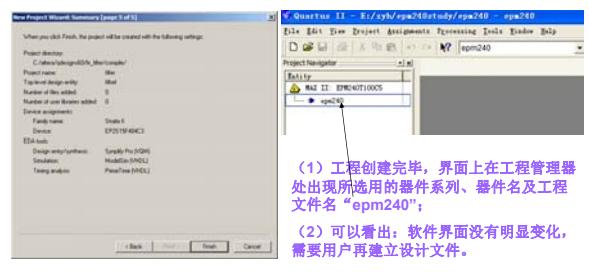


## 4、EDA 工具设置

选择第三方EDA工具  
(如ModelSim、  
Synplify等)  
这里不需要



## 5、完成!



## 关于创建工程的补充说明

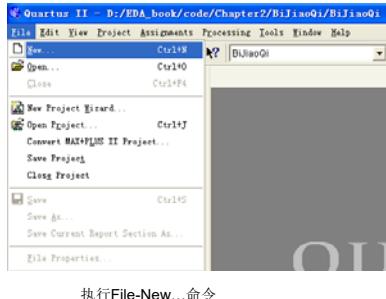
- （1）指定工程所在的工作库文件夹、工程名及设计实体名；
- （2）将设计文件加入工程中；
- （3）选择仿真器和综合器类型（默认“None”为选择QuartusII自带的）；
- （4）选择目标芯片（开发板上的芯片类型）；
- （5）工具设置（若都不选择，则使用QuartusII自带的所有设计工具）；
- （6）结束设置。
- 工程建立后，若需要新增设计文件，可以通过 Project /Add Remove.....在工程中添加新建立的设计文件，也可以删除不需要的设计文件。编译时将按此选项卡中列出的文件处理。

## 原理图输入设计方法流程

- 建立工程项目（工程目录、名称和选择合适器件）
- 编辑设计图形文件（放置元件、连线、设定输入输出管脚名称）
- 编译设计图形文件（检查电路是否有错误）
- 时序仿真设计文件（得到仿真波形验证设计结果）
- 生成元件符号

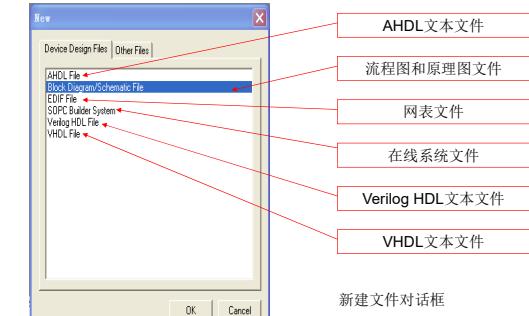
## 二、编辑设计图形文件

### 1. 建立原理图文件



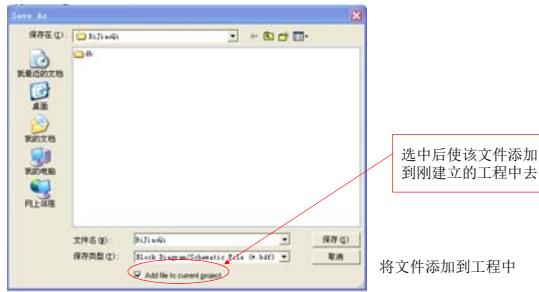
### 1. 建立原理图文件

弹出新建文件对话框如图

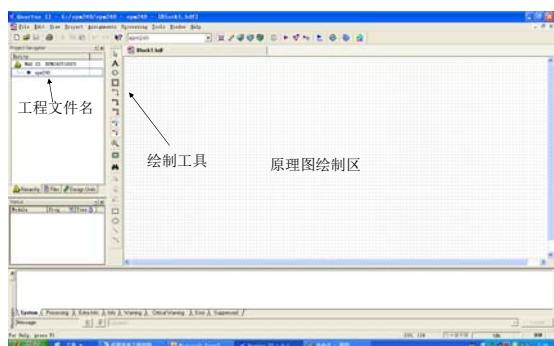


### 1. 建立原理图文件

执行File-Save as...命令，把它另存为文件名是XXX.bdf的原理图文件，后缀名为.bdf

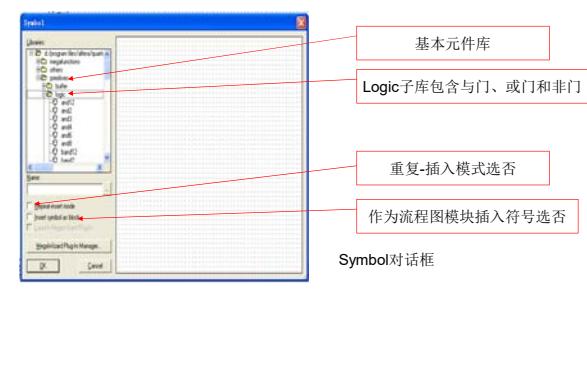
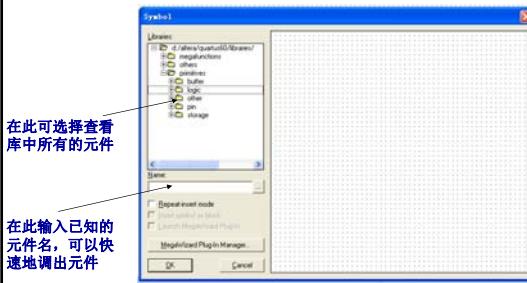


## 2、建立原理图设计文件

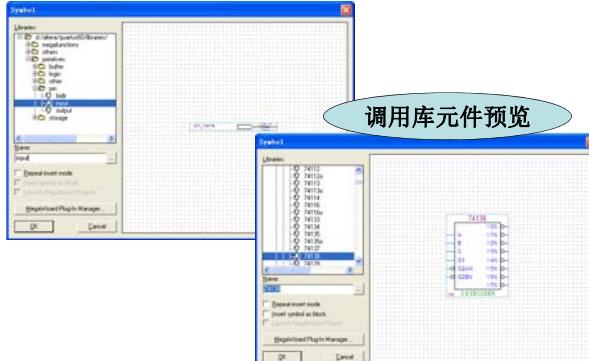


## 3、调用参数化元件

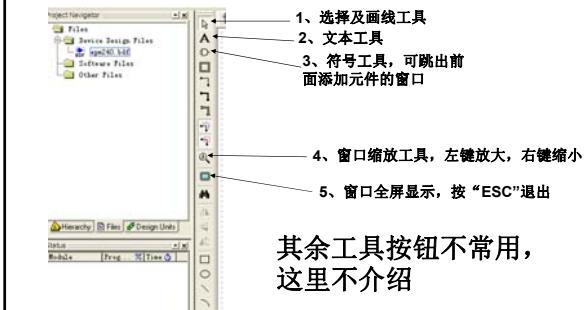
- 在绘图区双击鼠标左键，即弹出添加符号元件的窗口



- 分别调用了输入端口“input”和逻辑器件“74138”



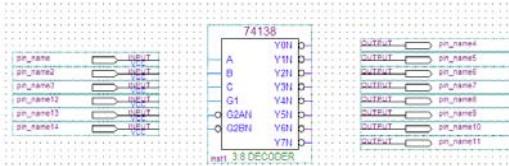
#### 4、绘图控制操作



说明：使用图示的工具按钮后，请切换回1按钮（选择及画线工具），才能对绘图进行编辑。

#### 5、设计74138，并进行功能验证测试

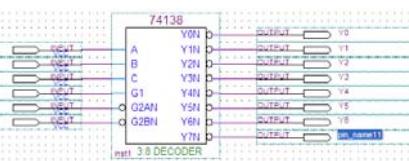
- 从符号库中调出74138及需要的输入、输出端口，排放整齐



- 完成画线连接操作（鼠标放到端点处，会自动捕捉，按下左键拖动到目标处，释放后即完成一次画线操作）

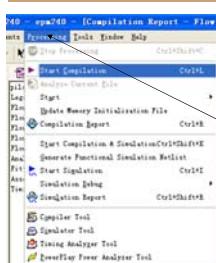
#### 为端口命名

- 鼠标左键双击端口号，如图示74138电路Y7N端所示，直接输入用户自定义的名字即可。
- 74138逻辑测试电路原理图设计完毕！



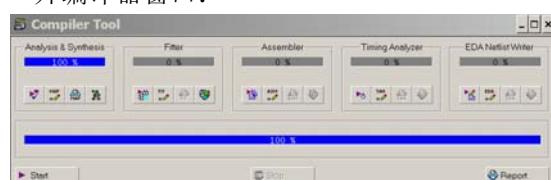
#### 四、全程编译

在下拉菜单“Processing”中选择“Start Compilation”，启动全程编译

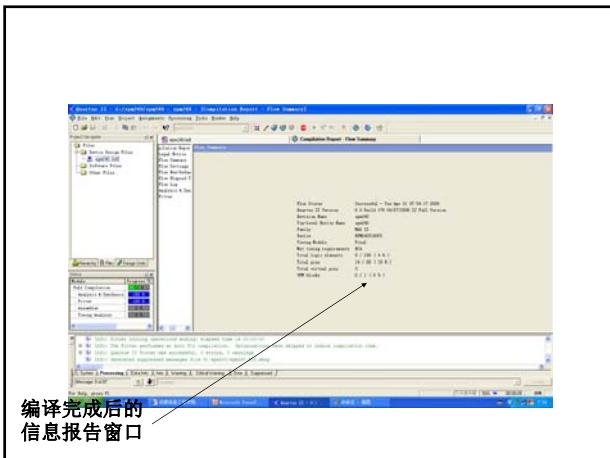


#### 进入编译器

- 选择命令 Processing | Compiler Tool，打开编译器窗口：



- 编译器包含5个主模块，可以连续运行5个模块，也可以单独运行某模块。



## 关于全程编译

- QuartusII的编译器由一系列处理模块构成;
- 这些模块负责对设计项目的检错、逻辑综合、结构综合、输出结果的编辑配置，以及时序分析;
- 在这一过程中，将设计项目适配到FPGA/CPLD目标器件中，同时产生多用途的输出文件，如功能和时序信息文件，器件编程的目标文件;
- 编译器首先检查出工程设计文件中可能的错误信息，以供设计者排除，然后产生一个结构化的网表文件表达的电路原理图文件;
- 启动全程编译：
  - 选择Processing/Start Compilation，自动完成分析、排错、综合、适配、汇编及时序分析的全过程。
- 编译过程中，错误信息通过下方的信息栏指示（红色字体）。双击此信息，可以定位到错误所在处，改正后在此进行编译直至排除所有错误；
- 编译成功后，会弹出编译报告，显示相关编译信息。

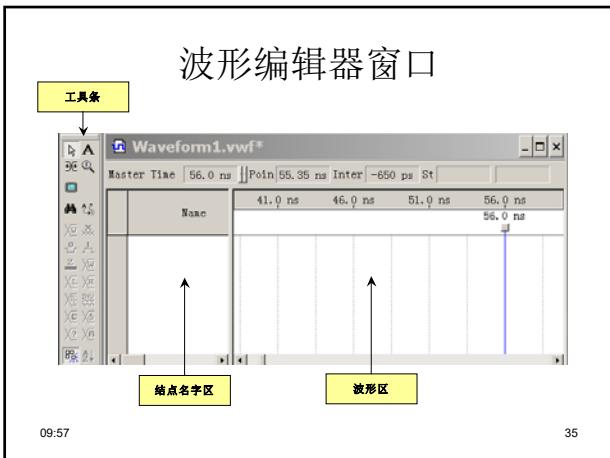
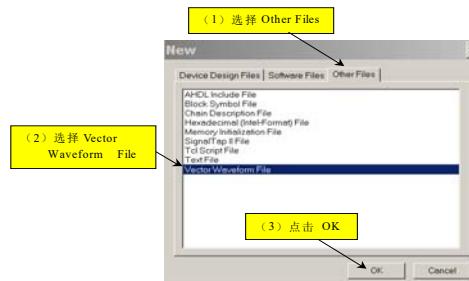
**五、时序仿真**

顺序并不是唯一的

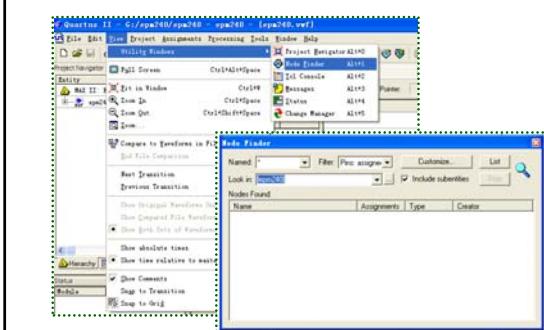
- 工程编译完成后，设计结果是否满足设计要求，可以通过时序仿真来分析；
- 时序仿真主要包含如下的设置步骤：
  - 打开波形编辑器；
  - 设置仿真时间区域；
  - 波形文件存盘；
  - 将端口节点信号选入波形编辑器中；
  - 编辑输入波形（输入激励信号）；
  - 总线数据格式设置
  - 启动仿真器
  - 观察仿真结果（波形编辑文件及产生的波形报告文件分开显示）
  - 若无法观察完整波形，可以使用热键Ctrl+W，即可看到完整的仿真波形。也可使用鼠标左右键，方法如下：

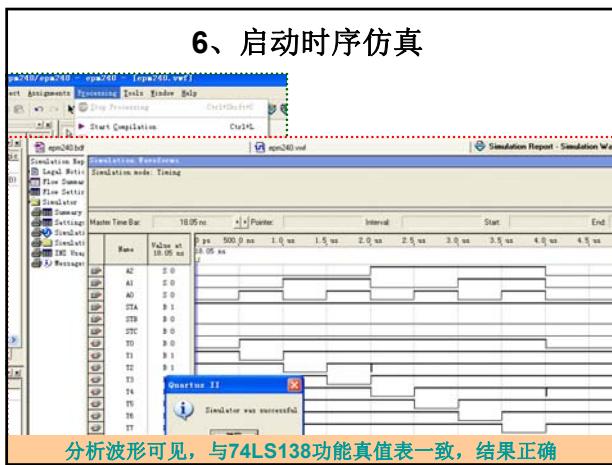
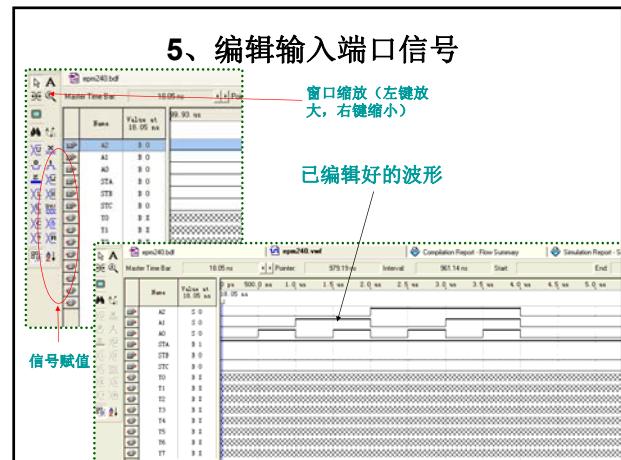
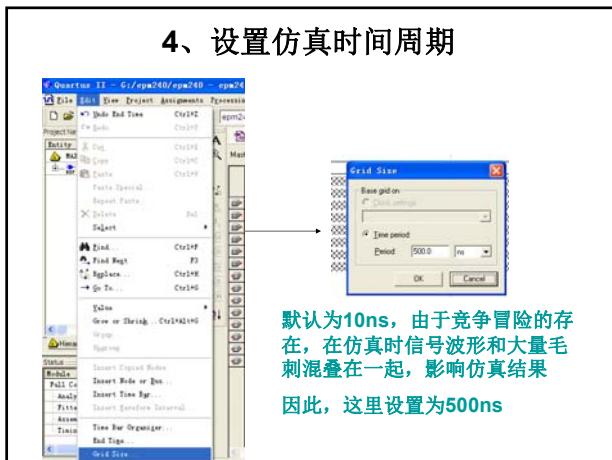
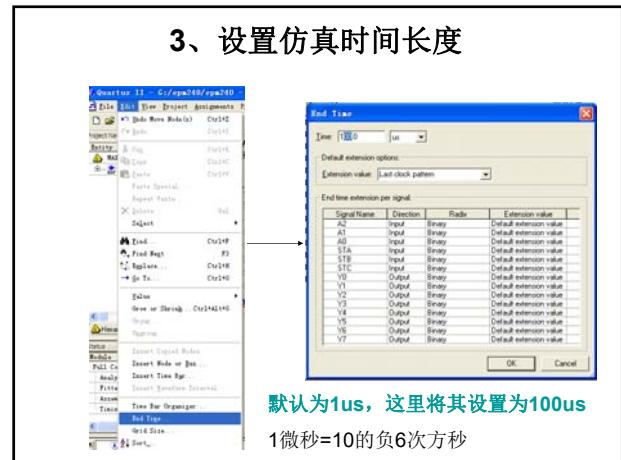
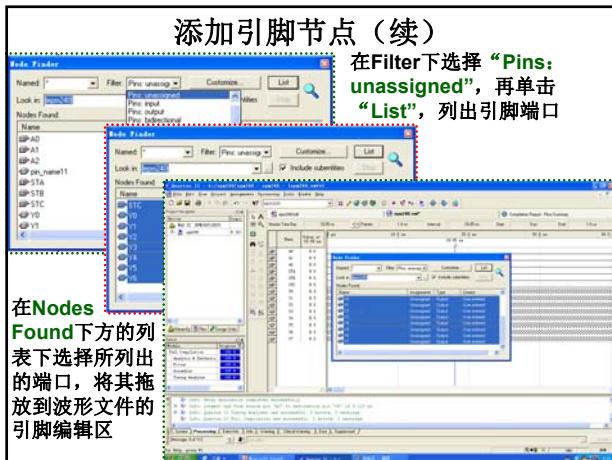
（1）选中后，右键放大，左键缩小

- 打开波形编辑器建立测试向量波形文件
- 选择命令 File | New → 执行以下操作 → 打开波形编辑器窗口：



## • 2、添加引脚节点





**LPM(Library Parameterized Modules) 参数化的宏功能模块库**

- Altera提供的宏功能模块
  - 算术组件：累加器，加法器等
  - 门电路：多路复用器，门函数等
  - I/O组件：千兆位收发器，PLL等
  - 存储器编译器：FiFo Partitioner、RAM和ROM等
  - 存储器组件：移位寄存器宏模块，LPM存储函数。

## 调用带参数的库元件

实例：在原理图编辑器中调用库元件 lpm\_ram\_d，  
它是一个带参数的存储器元件，本例  
字长8位，地址码长度5位（存储容量32字）。

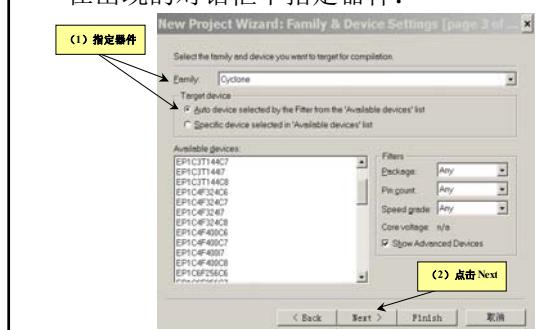
### 调用库元件 lpm\_ram\_d

- 使用命令File | New Project Wizard ➔ 出现对话框 在该对话框中创建新项目：



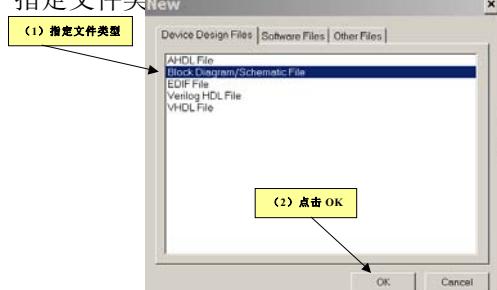
### 调用库元件 lpm\_ram\_d (续)

- 在出现的对话框中指定器件：



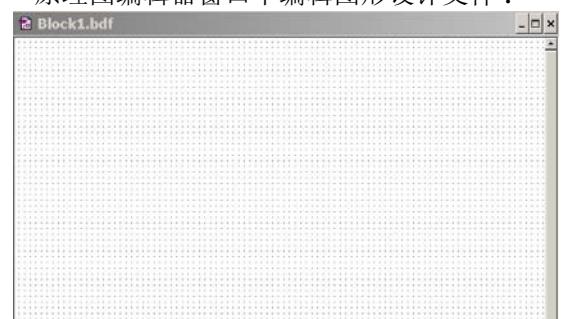
### 调用库元件 lpm\_ram\_d (续)

- 使用命令File | New ➔ 在出现的对话框中指定文件类型



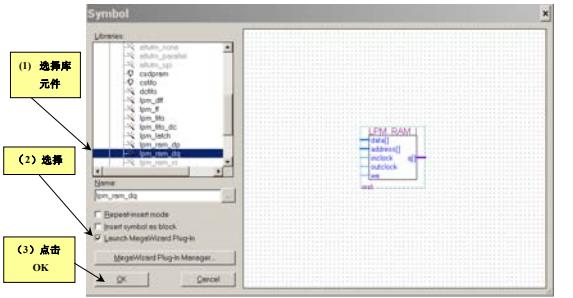
### 调用库元件 lpm\_ram\_d (续)

- 原理图编辑器窗口中编辑图形设计文件：



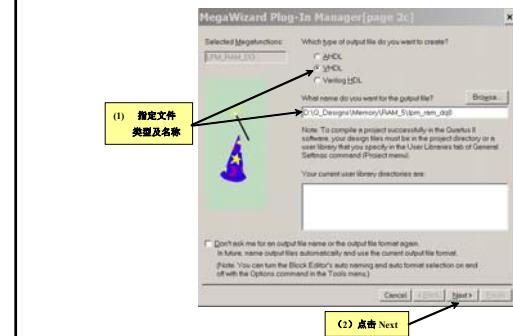
## 调用库元件 lpm\_ram\_d (续)

- 双击鼠标左键（或使用命令Edit | Insert Symbol）在出现的对话框中选择库元件



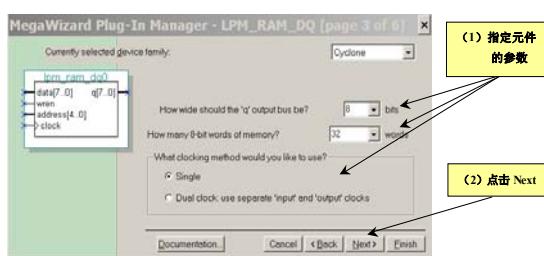
## 调用库元件 lpm\_ram\_d (续)

- 指定输出文件类型和名称



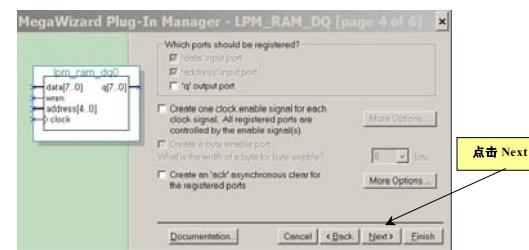
## 调用库元件 lpm\_ram\_d (续)

- 指定元件lpm\_ram\_dq0的参数:



## 调用库元件 lpm\_ram\_d (续)

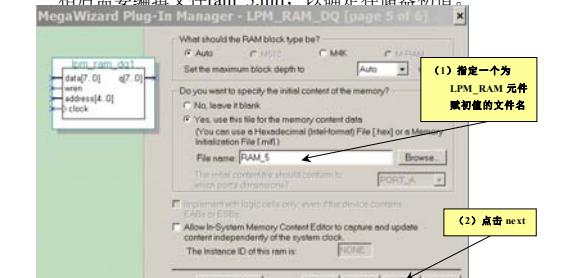
- 在出现的对话框中点击Next:



## 调用库元件 lpm\_ram\_d (续)

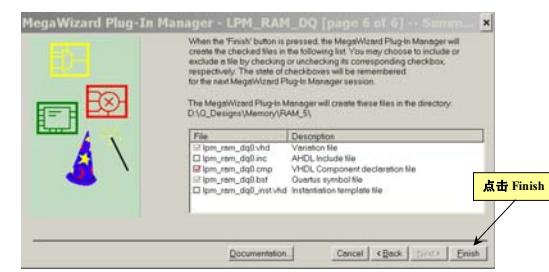
- 决定是否为存储器元件赋初值（否则，默认值为全0）本例赋初值文件名ram\_5.mif, (memory initialization file , mif)

- 稍后需要编辑文件ram\_5.mif, 以确定存储器初值。



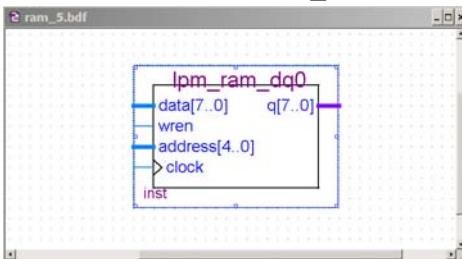
## 调用库元件 lpm\_ram\_d (续)

- 在出现的对话框中点击Finish



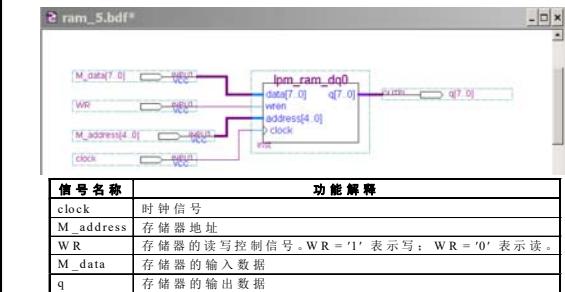
### 调用库元件 lpm\_ram\_d (续)

- 自动返回到原理图编辑器 ➔  
原理图编辑器中出现被调用的元件 ➔  
**lpm\_ram\_dq0**  
将此图形文件以文件名ram\_5.bdf保存起来



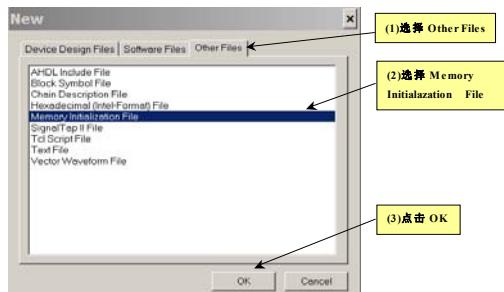
### 调用库元件 lpm\_ram\_d (续)

- 添加输入 / 输出引脚 ➔  
将此图形文件以文件名ram\_5.bdf保存起来



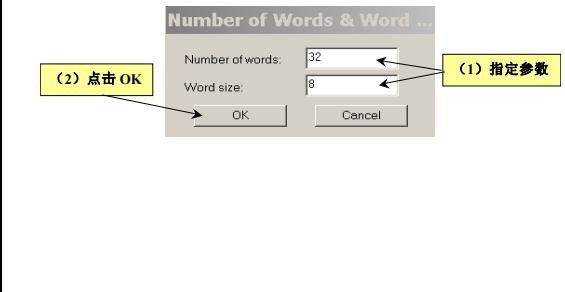
### 初始化存储器的内容

- 使用命令File | New ➔  
在出现的对话框中操作 ➔



### 初始化存储器的内容 (续)

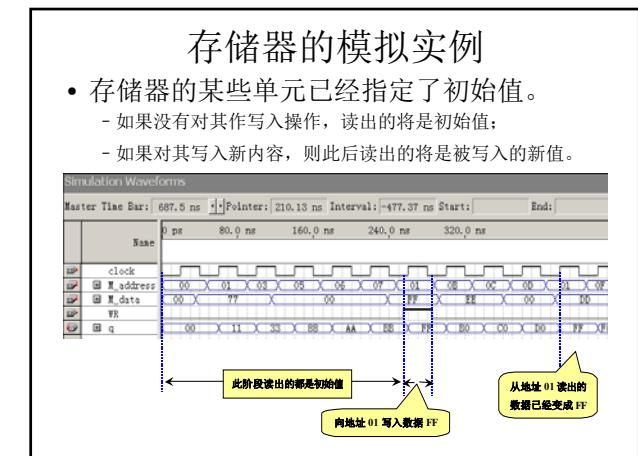
- 在出现的对话框中指定存储器的参数 ➔



### 初始化存储器的内容 (续)

- 在出现的对话框中为每一个存储单元指定初始值 ➔  
指定完毕后，以文件名ram\_5.mif将其保存。

Addr	+0	+1	+2	+3	+4	+5	+6	+7
00	00	11	22	33	77	88	AA	BB
08	CC	DD	AA	BB	CO	DO	E0	FO
10	00	00	00	00	00	00	00	00
18	00	00	00	00	00	00	00	00



- 实验内容以及要求

## 实验: 算数逻辑运算实验

- 一、实验目的
  - 1. 熟悉QuartusII环境。
  - 2. 学习在QuartusII中新建项目、文件、编译、仿真。
  - 3. 学习原理图方式自定义元件的输入，封装，调用
  - 4. 掌握运算器的功能
  - 5. 学会使用现成的alu 元件

- 二. 实验原理
  - 利用74181和74182芯片采用组间并行和组间串行两种方式构建16位加法器。参加运算的两个16位数据分别为A[15..0]和B[15..0]，运算模式由S[3..0]=1001决定做加法运算。设M=0，选择算术运算，CN为低位的进位位；F[15..0]为输出结果，CN16为运算后的输出进位位。画出逻辑图，并观察进位延迟。

- 三. 实验内容
  - (1) 在QuartusII环境下以原理图方式建立顶层文件工程
  - 选择图形方式，根据输入实验电路图，从Quartus II 的基本元件库中将各元件调入图形编辑窗口、连线，添加各输入输出引脚名称。将所设计的图形文件 ALU.bdf 保存到原先建立的文件夹中，将当前文件设置成工程文件，以后的操作就都是对当前工程文件进行的。
  - (2) 建立仿真文件，ALU.vwf，进行软件仿真，观察输出结果。

- 四. 实验要求
  - 1. 做好实验预习，掌握运算器的数据传送通路和ALU的功能特性。
  - 2. 比较组间串行方式和组间并行方式的进位延迟。

## 实验报告要求

- 1. 记录设计原理图。
- 2. 记录综合结果（逻辑单元<Le>的消耗情况等）。
- 3. 记录仿真结果（画出仿真波形）。
- 4. 分析结果（实验现象结论）。

## 提交方式以及截止时间

- 注意：需独立完成实验。
- 提交文件：**rar**压缩文件提交，压缩内容应包括实验的所有内容：
- **1、实验报告电子档（.doc格式）A4排版宋体四号字（格式见模板）。**
- **2、实验工程目录下所有文件，包括源代码、波形文件等。（要求可以直接仿真看结果的，提交后如果不能综合，看不到结果为无效提交）。**
- **3、压缩内容清单。**

### • 提交方法：

- 截至时间：2026年1月25日24时
- 提交方式：
  - 通过电子邮件提交个人作业，邮件发至 [ronbit@163.com](mailto:ronbit@163.com) (有自动回复)
- 提交格式：
  - 邮件主题：学号\_姓名
- 文件命名方式：
  - 学号\_姓名.rar