

详细讲解Xilinx DDR3 的MIG IP生成步骤及参数含义

基于FPGA的DDR相关知识导航界面，点击查看。

前几篇文章讲解了SDRAM到DDR3 各自的变化，本文讲解如何使用DDR3，在Altera的Cyclone IV开发板上一般会使用SDRAM作为存储数据的芯片，而Xilinx的S6和7000系列一般使用DDR3作为存储数据的芯片。

从SDRAM芯片内部结构分析其原理，从内部结构讲解SDRAM与DDR的变化，从内部结构讲解DDR到DDR2的变化，从内部结构分析DDR2到DDR3的变化及DDR3原理图分析。

SDRAM的控制器还算比较简单，一般可以通过Verilog HDL去实现，也推荐初学者写一下SDRAM的驱动，相比UART、IIC这些驱动难度还是会大一点，能够强化状态机的代码能力，另外也能充分理解SDRAM的控制流程。

对于DDR3控制器来说，一般都是调用Altera或者Xilinx官方的MIG IP来完成设计，DDR3控制器的难度偏大，即使通过Verilog HDL写出来，性能和稳定性也未必能得到保障。

Vivado通过调用MIG实现对DDR3的控制，用户在使用时需要了解该IP的各个参数。Xilinx提供了ug586对该IP的功能和参数进行说明，由于该IP的功能比较多，导致手册有六百八十页。涉及到DDR3相关内容也只是一百多页内容，本文结合手册对IP配置的各个参数进行讲解。

1、生成MIG IP

首先在vivado中创建一个工程，然后添加MIG IP，如下图所示。

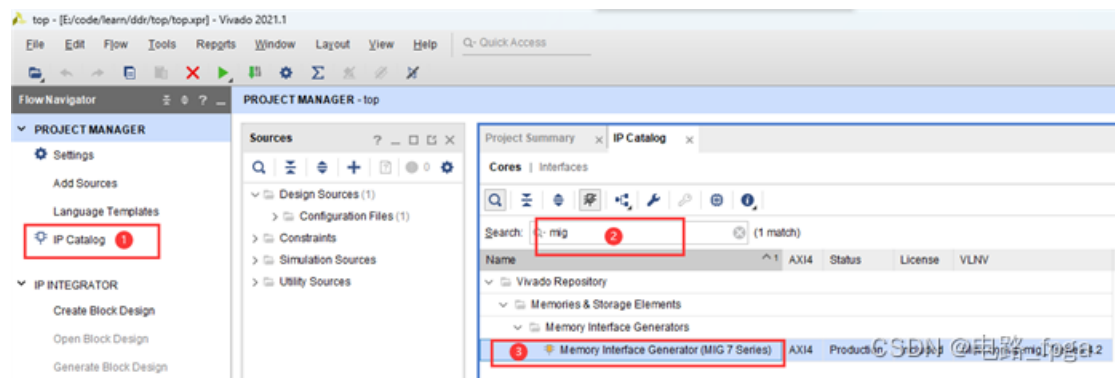


图1 添加MIG IP

双击进入MIG配置界面后，首先如下图所示，该界面显示MIG的功能以及本工程选定的 fpga 器件型号，我使用的是zynq7030器件，直接进入下一步即可。



图2 向导界面

进入MIG Output Option页面，该页有几个选项，意思分别如下：

1. Create Design：创建DDR3 IP工程设计，此处勾选。
2. 表示对已有工程引脚约束进行验证。
3. 这个IP生成后的模块名。
4. 由于这个IP支持多个控制器，所以这里需要选择主控的数量，FPGA使用的话一般只会存在一个。

5. 这个IP有两种接口，默认使用的接口比较简单，适合FPGA内部使用，勾选5处会使用AXI4接口，这个接口一般给ARM端控制DDR3会比较方便，FPGA内部使用AXI4接口就是把简单的问题复杂化且浪费资源。

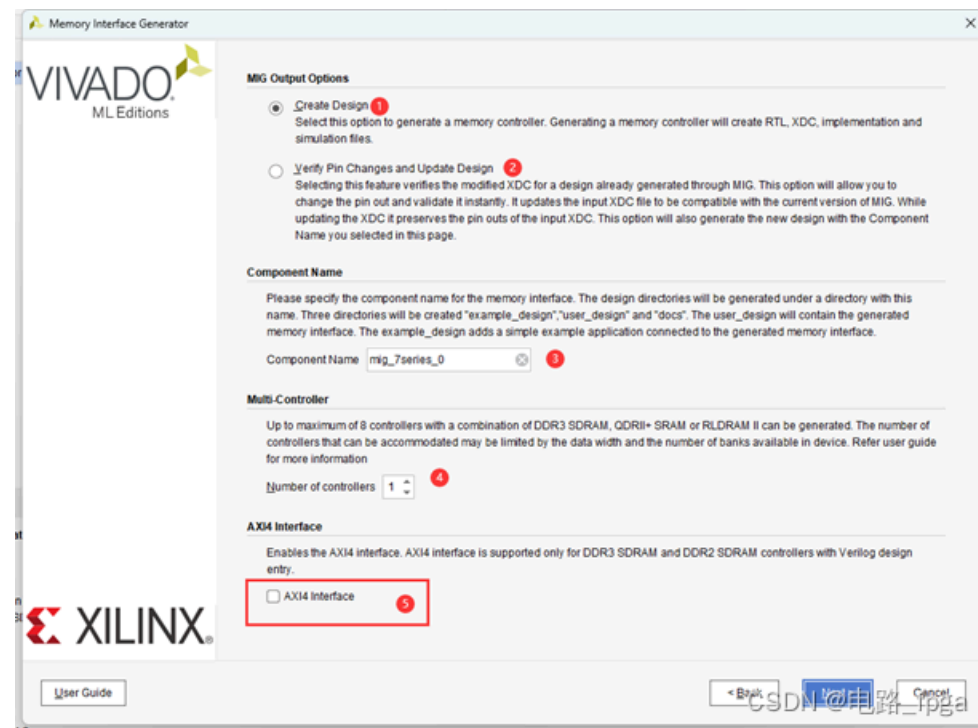


图3 MIG Output Option配置页面

点击Next到Pin Compatible FPGAs 配置页面，如下图所示该界面可用于配置选择和当前所设定的唯一器件型号做引脚兼容的其它FPGA型号。对于可能升级器件型号的应用而言，这个功能是很实用的。本设计保持默认不做配置。

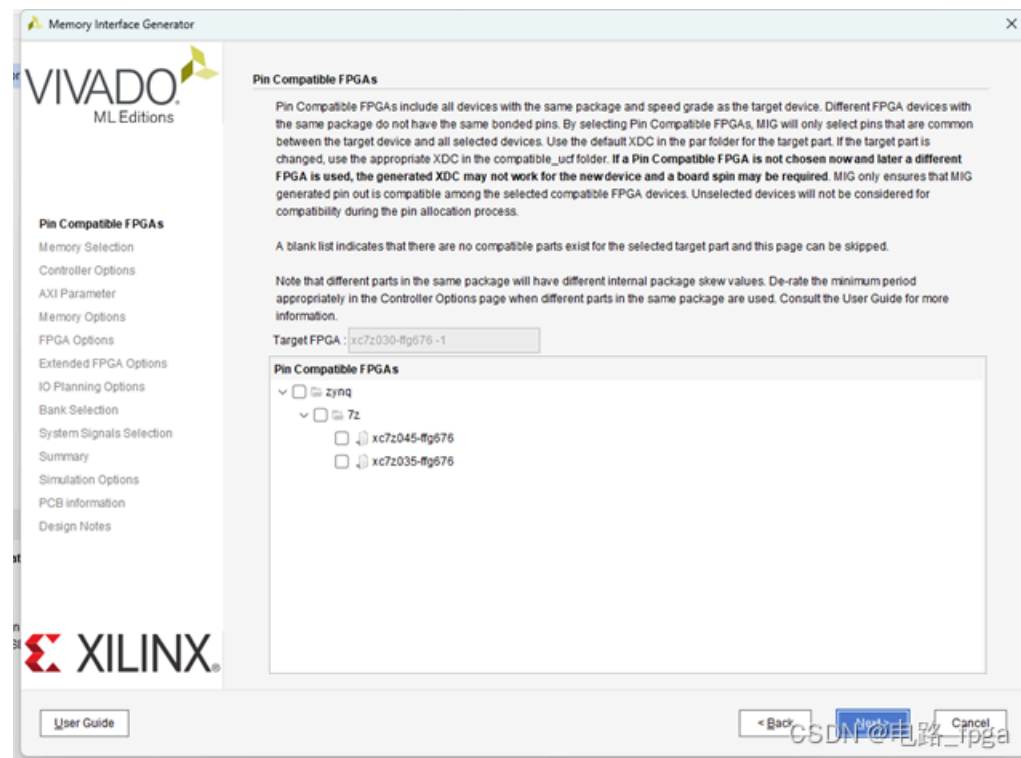


图4 Pin Compatible FPGAs配置页面

点击Next进入到如下图所示的Memory Selection配置页面，该IP可以控制DDR3、DDR、LPDDR2等多种器件，所以这里需要选择这个IP具体控制哪种芯片，**开发板** 上板载的是DDR3，这里选择“DDR3 SDRAM”即可。

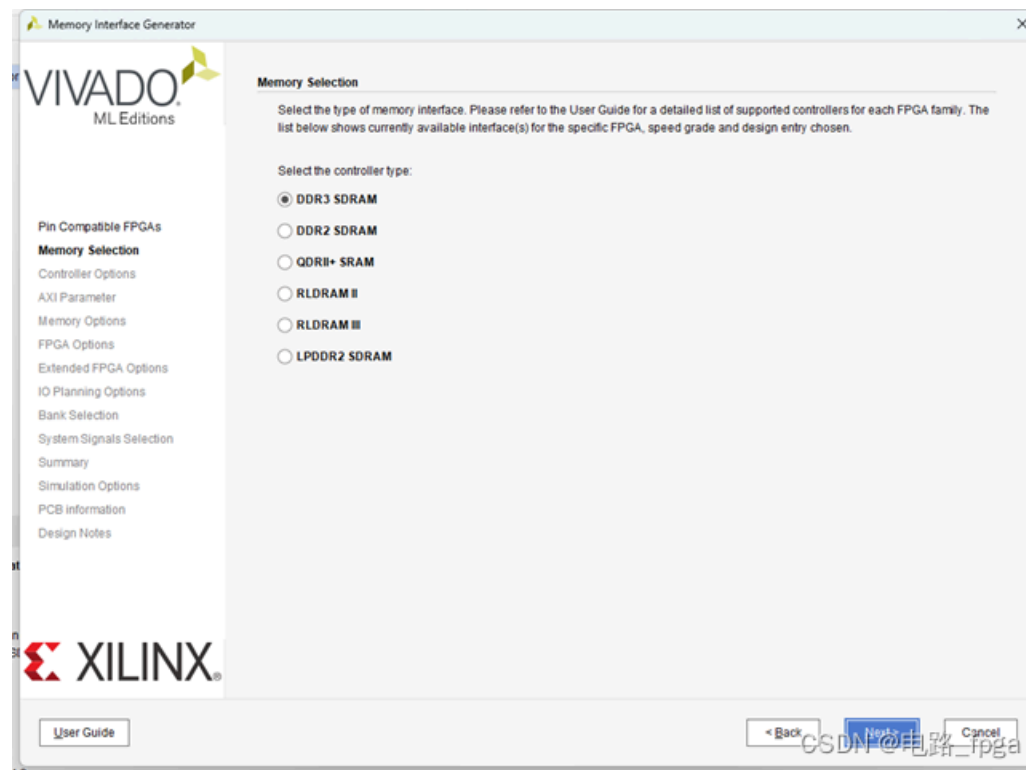


图5 Memory Selection配置页面

点击Next进入到如下图所示的Controller Options配置页面该配置，该界面是主要参数的配置界面之一，涉及的内容比较多。

1、Clock Period：设置DDR3的IO时钟频率，根据开发板上实际DDR3支持的最大时钟频率进行设置，MT41K256M16RE-125最大支持800MHz的接口时钟，在加上ZYNQ 7030是K7架构，DDR3运行在800MHz时钟频率下是没有问题的。但是本文只是对IP进行测试，所以选用400MHz即可。

2、PHY to Controller Clock Ratio：FPGA给DDR3芯片提供的时钟频率为Clock Period，但是FPGA内部其余模块的时钟一般是运行不到Clock Period这么高的时钟频率的，所以需要给FPGA内部提供一个较低的时钟，这个参数就是用来设置DDR3接口时钟与MIG IP提供给FPGA内部其他模块的时钟(ui_clk)频率比值。

如果fpga是A7系列的芯片，那么在设置Clock Period为400MHz之后，该选项应该会被锁定为4:1，也就是说内部逻辑时钟只支持100MHz。

由于ui_clk与DDR3接口时钟频率比值为4:1，且DDR3的数据在双沿传输，导致用户每次读写MIG IP的数据位宽是DDR3芯片数据位宽的8倍，DDR3如果数据位宽为16位，那么用户每次读写IP的数据为128位，这是不是与DDR3芯片内部的预取思路差不多？通过位宽换取时钟频率。

3、Vccaux_io：设置bank的供电电压，一般都是锁死的，会根据DDR3时钟频率变化而变化。

4、Memory Type：一般选择Components即可，其余几种与内存条有关吧。

5、Memory Part：选择DDR3的型号，芯片类型不同会涉及具体的DDR3时序参数不同以及地址线和数据线的位数不同。

6、Memory Voltage：选择芯片引脚的电平标准，DDR3一般只有1.5V和1.35V两种，根据原理图选择支持的电平的类型即可。

7、Data Width：DDR3读写数据位宽，根据原理图选择DDR3位宽。前文介绍过核心板上有两颗DDR3芯片，采用数据线扩展方式连接，支持32位读写数据。但是本文只使用其中一块DDR3作为存储体，所有读写数据位宽为16。

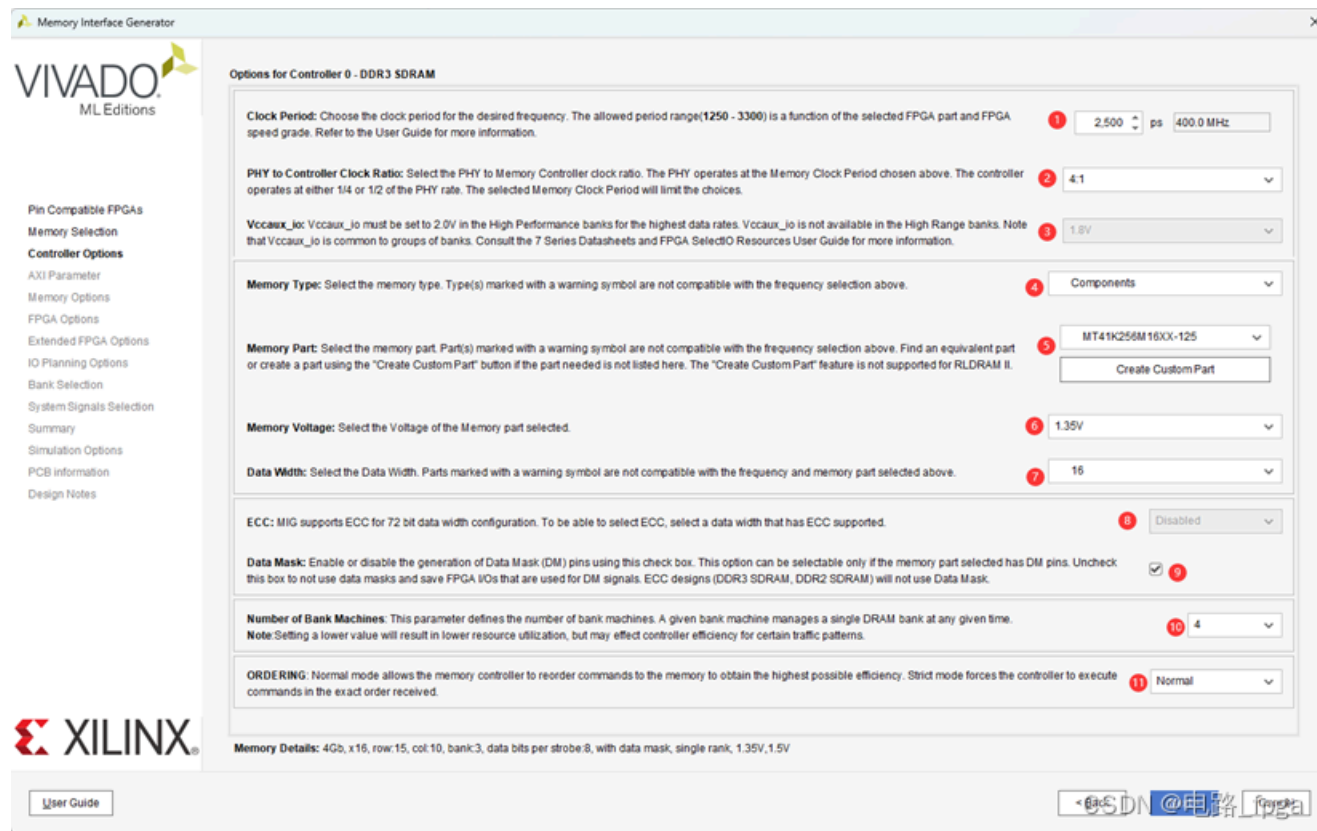


图6 Controller Options配置页面

8、ECC：数据线多余72为才能使用的功能。

9、Data Mask：启用之后，可以通过一个掩膜信号去控制DDR3写入数据，本质就是去控制DDR3的DM引脚，勾不勾选无所谓，一般也不会使用此功能。

10、Number of Bank Machies：选择bank数量，DDR3虽然有8个bank，但是如果不需要全部使用时，可以只选择部分bank，降低功耗，节约资源，默认使用4个bank。

11、ORDERING：DDR控制器调度命令的顺序的配置，当选择strict时，严格按照命令先后顺序执行；选择normal时，为了得到更高的效率，可能对命令重排序。

点击Next进入到如下图所示的Memory Options配置页面下。

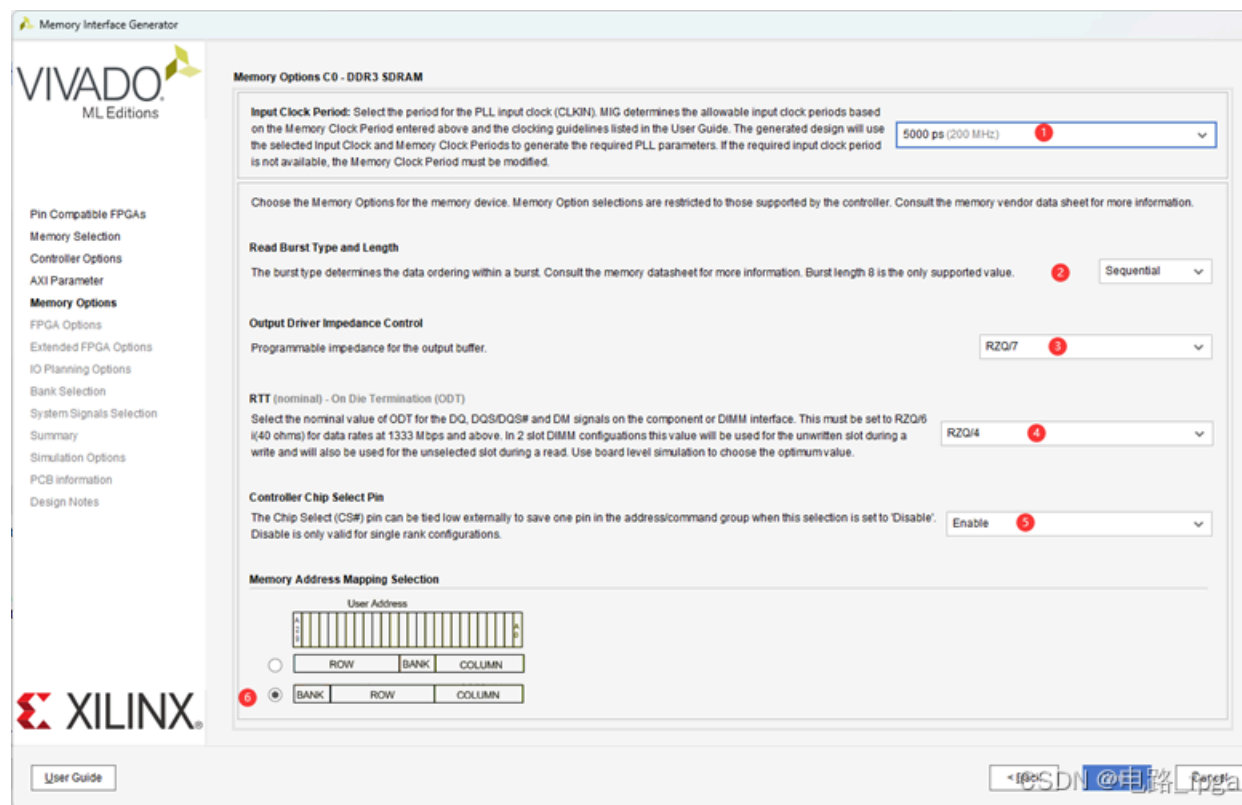


图7 Memory Options配置完成后页面

1. Input Clock Period: 这个时钟是提供给MIG IP的时钟sys_clk，如下图所示为MIG IP内部时钟的结构图，sys_clk进入IP之后会先经过一个锁相环PLL，锁相环会根据这里的输入时钟产生MIG IP内部各种所需的时钟，所以sys_clk的频率选择并没有限制，下拉框中的各种频率值都可以选择。

但是下图中还需要给MIG IP提供一个200MHz的IDELAY参考时钟，至于IDELAY的功能相比不用多说了吧，前文讲解原语时详细讲过。那么是不是可以把这两个时钟进行合并，那么FPGA就只需要给MIG IP提供一个200MHz的时钟就行了，这个200MHz时钟既是IDELAY的参考时钟又作为该IP锁相环输入时钟。

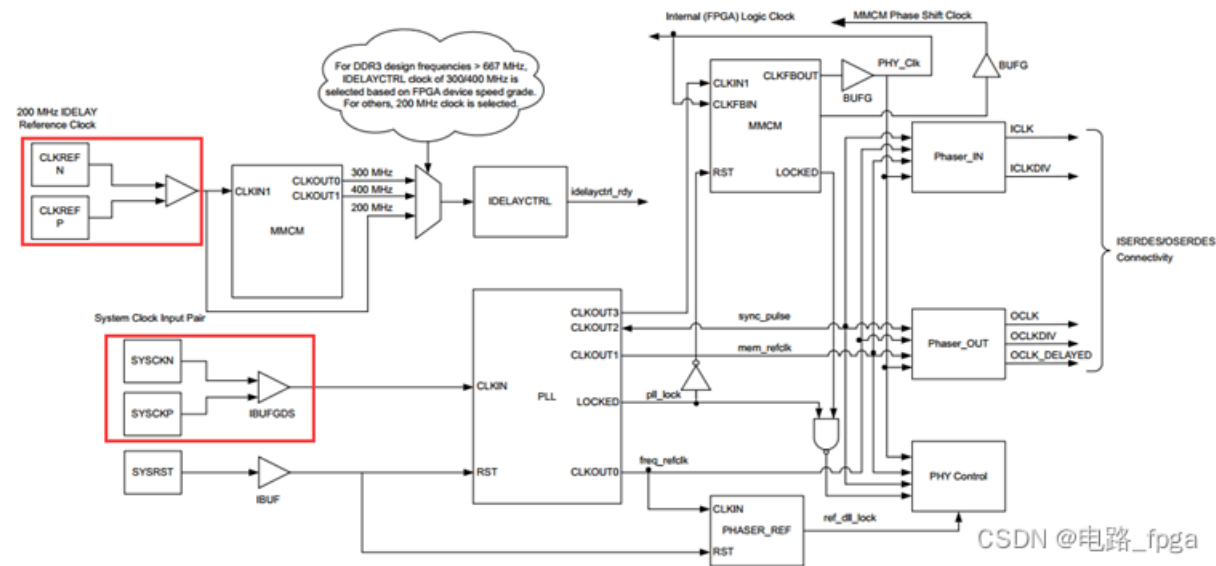


图8 MIG时钟结构图

2. Read Burst Type and Length: 设置为顺序读写Sequential, 从SDRAM到DDR3都支持该功能, 该参数在模式寄存器中进行设置, 两种读写方式的区别在SDRAM的文章中详细分析过, 需要了解的可以查看。

3. Output Drive Impedance Control: 输出驱动阻抗控制选择RZQ/7。

4. On Die Termination: 片上终端电阻设置为R ZQ/4。

5. Controller Chip Select Pin: 片选信号设置为Enable, 如果开发板上DDR3的CS管脚未连接到FPGA, 设置为Disable。

6. Memory Address Mapping Selection: DDR和AXI总线之间的地址映射存储器地址映射选择。默认选择后者, 在调用IP时, 其实不会关心bank地址和行、列地址。

点击Next进入到如下图所示的FPGA Options配置页面中, 如下图所示。

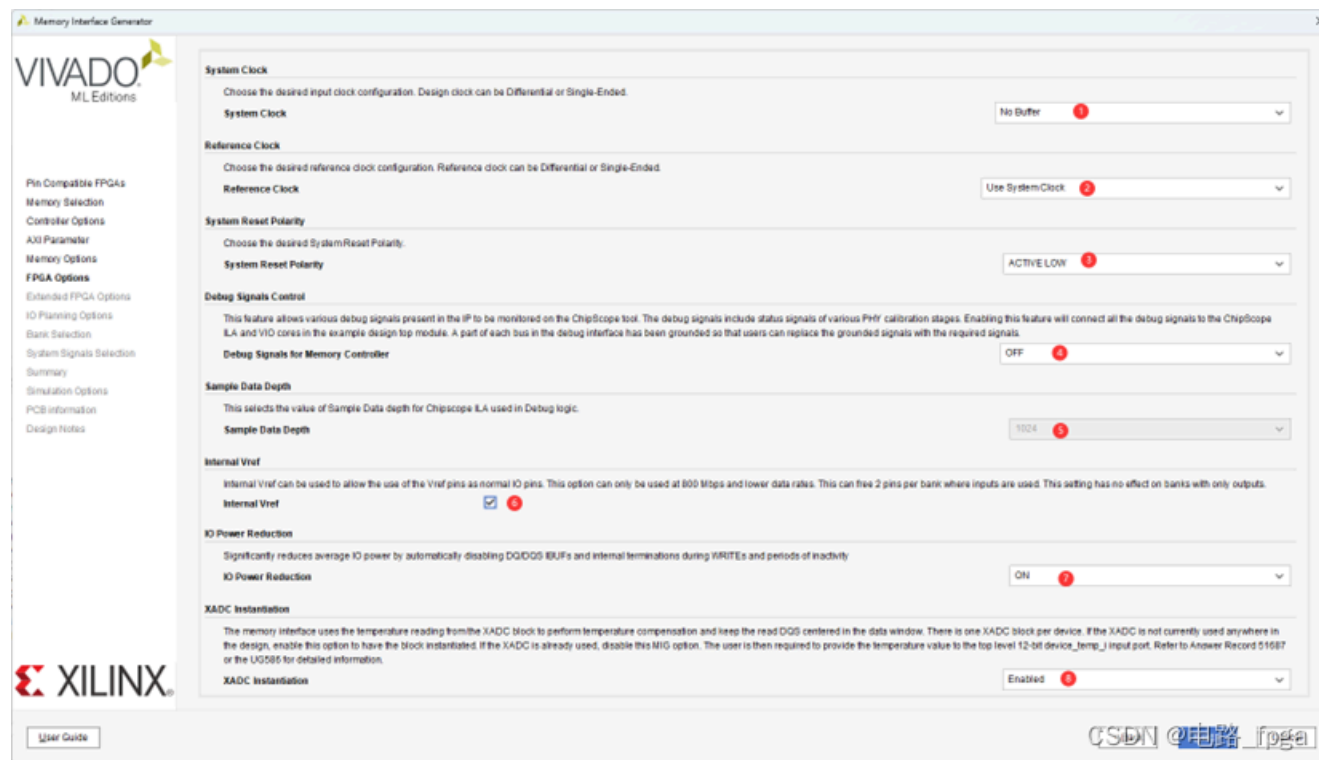


图9 FPGA Options配置页面

- 1、System Clock：设置系统时钟sys_clk的来源，图7设置为200MHz。由FPGA内部提供，不由管脚输入，选择No Buffer。如果实际硬件管脚有提供200MHz时钟，也可以选择Differential（差分输入）或Signal-Ended（单端输入）。
- 2、Reference Clock：设置图8中IDELAY参考时钟200MHz的来源，由于在图7中将系统时钟设置为200MHz，可以选择Use System Clock，将两个输入时钟合并一个共用的200MH输入。如果图7中系统时钟设置不是200MHz，这里配置选项就没有“Use System Clock”，只能由管脚端口输入时钟或者FPGA内部产生这个200MHz时钟。
- 3、System Reset Polarity：系统复位极性设置为ACTIVE LOW（低电平）。
- 4、Debug Signal for Memory Controller：关闭存储器控制器的调试信号，如果选择ON，就会在IP中添加ILA作为调试，本文并不需要，关闭即可。
- 5、Sample Data Depth：只有4设为ON，此处才能设置，就是设置ILA采集数据的深度。
- 6、勾选internal Verif，允许把参考引脚作为正常的IO引脚使用。
- 7、IO Power Reduction：相当于不操作时进入休眠状态降低功耗，这个一般关闭。

8、XADC Instantiation: 使能MIG IP通过XADC来获取温度, 实现温度补偿, 没有单独使用XADC时启用该功能。

点击Next进入到如下图所示的Extended FPGA Option 配置页面中, 其中1和2只有在HP Bank的FPGA中才会出现, HR bank没有这两个设置内容, 一般默认即可。设定内部终端阻抗 (Internal Termination Impedance) 为50 Ohm。

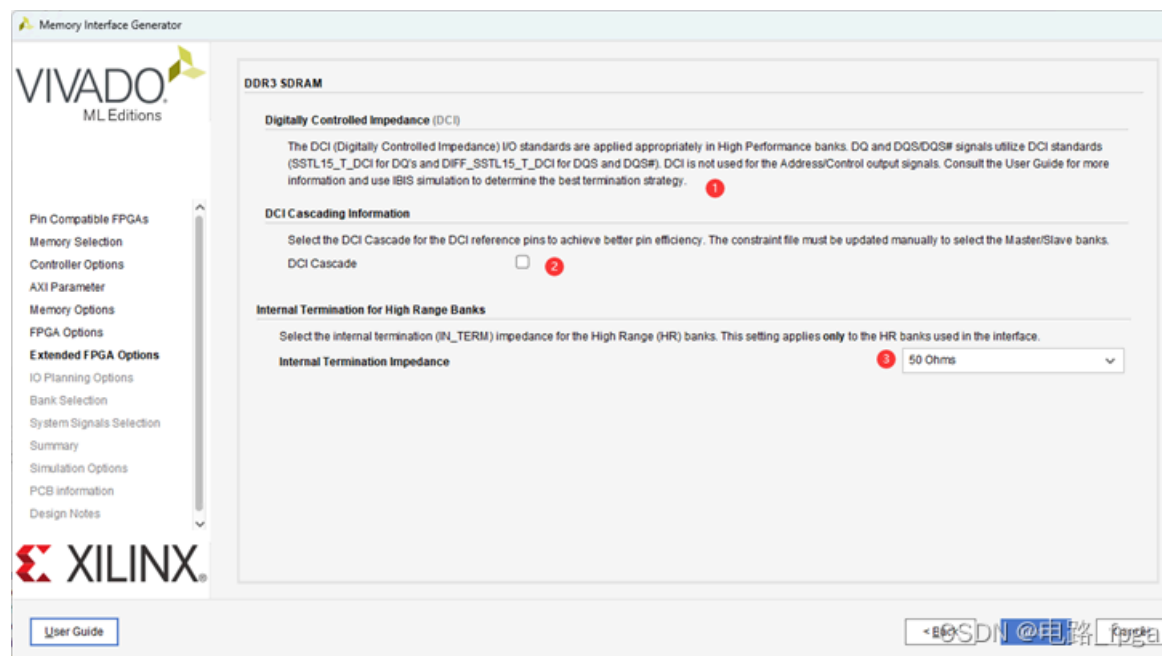


图10 Extended FPGA Option配置页面

点击Next进入到如下图所示的IO Planning Option配置页面中, 如果当前IP仅用于仿真, 可以选择New Design。本次实例最终是要进行上板测试的, 则选择“Fixed Pin Out: Pre-existing pin out is known and fixed”。

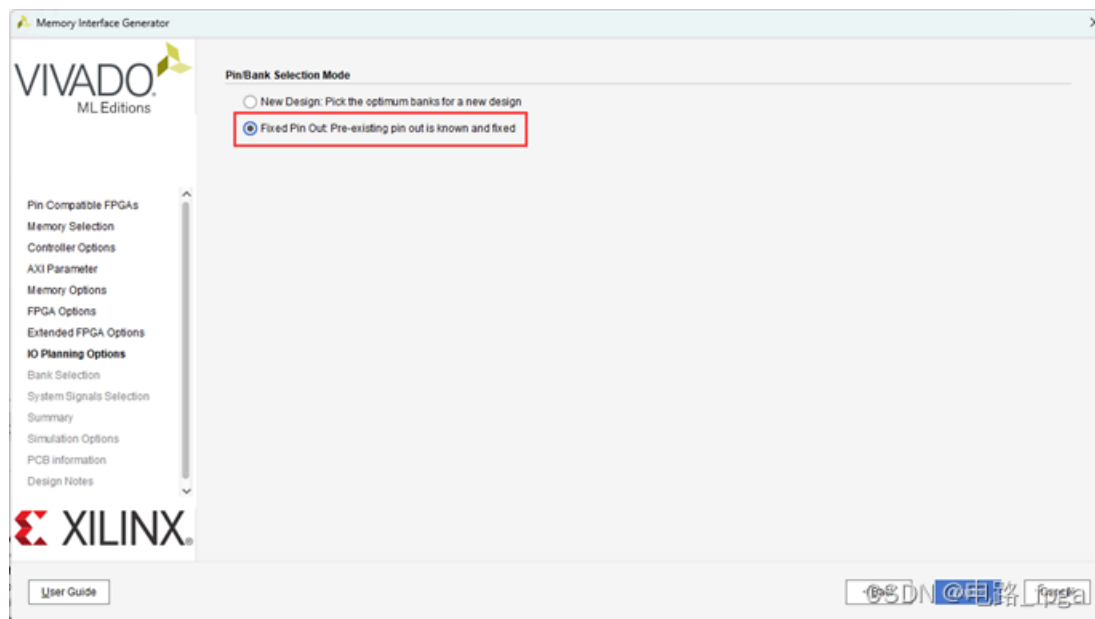


图11 IO Planning Option配置页面

点击Next进入到如下图所示的Pin Selection配置页面中，所有DDR3存储器相关的引脚定义引脚号以及IO电平标准的配置，需要和原理图连接相一致。有2种配置方式，其中一种配置方式是采用手动输入的方式，另一种采用导入.uc约束文件方式，更加方便，添加约束的步骤如下图所示。

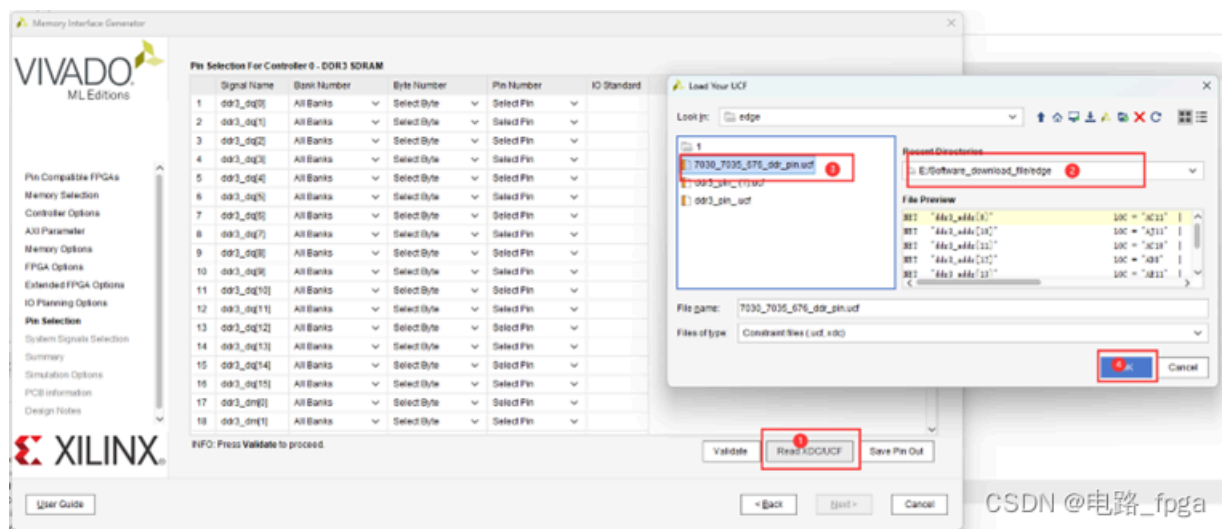


图12 Pin Selection配置页面

点击OK后，如下图所示，Pin Selection页面中所有DDR3存储器相关的引脚都已经配置完成。点击按钮“Validate”做语法检查，弹出如下图所示的DRC Validation Log message窗口，表示引脚分配通过DRC检查，点击OK。

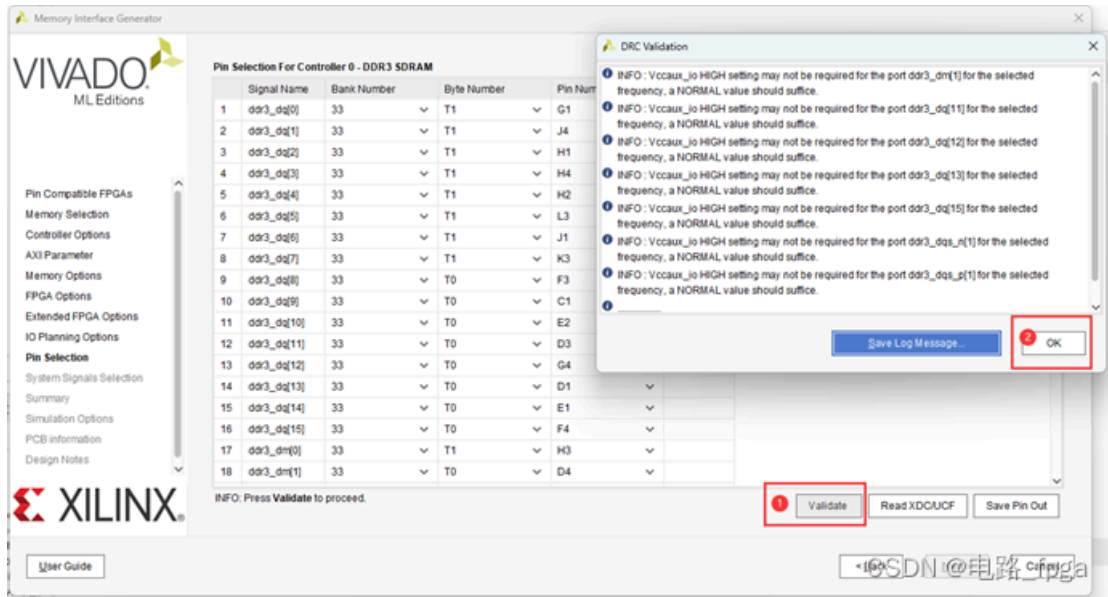


图13 Pin Selection配置页面管脚分配验证

到此MIG IP的配置就基本完成了，之后的界面均点击“Next”，直到生成IP为止。

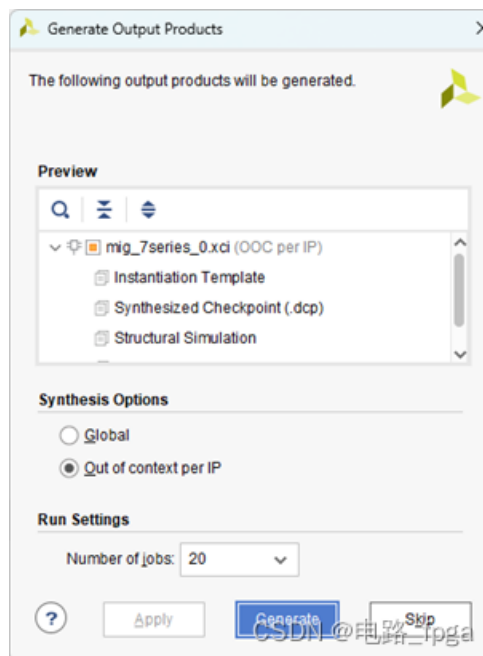


图14 生成MIG IP

2、总结

生成MIG IP需要特别关注的就是时钟的设置，主要包括4个时钟信号。

图6中1处决定DDR3芯片工作时端口时钟频率，也就是DDR3芯片的差分输入时钟频率。

图6的2处决定MIG IP输出给用户端口的相关信号的时钟频率与DDR3芯片差分输入时钟频率的比值。根据FPGA的架构不同，该处能够设置的比值不同，对于A7架构，用户逻辑很难运行在200MHz时钟频率下，所以当1处设置为400MHz时，2处不支持设置成2:1，只能设置为4:1。但是如果FPGA架构为K7，当1处设置为400MHz，那么2处可以设置为2:1，即用户时钟接口信号的频率支持200MHz。

图6中两处设置分别决定DDR3芯片差分时钟频率，MIG IP的用户接口信号ui_clk的频率。

然后MIG IP还需要输入两路时钟信号，一路作为内部IDELAY延时的参考信号，固定为200MHz。

另一路为MIG IP系统时钟，通过内部锁相环生成需要的各种频率信号，由于这一路时钟信号作为锁相环的输入，所以对频率的具体数值没有要求。因此图7的1处可以设置为任意频率的时钟信号。

既然系统时钟信号频率可以是任意值，那么就可以将频率与参考时钟的频率设为一致，之后就可以只给MIG IP提供一路200MHz的时钟信号，这一路时钟既作为IDELAY的参考时钟信号，又作为MIG IP的系统时钟输入。这就是图7的1处和图9中部分设置。

其余很多设置都是默认的，熟悉原理即可，MIG用户读写数据位宽与图6中2的设置和数据位宽的设置有关，最终要达到数据传输速率一致，具体问题具体分析。

本文完成了对MIG IP的参数分析和设置，生成了IP，下文将讲解生成IP的各个信号的含义，以及对该IP的仿真及上板验证。

如果对文章内容理解有疑惑或者对代码不理解，可以在评论区或者后台留言，看到后均会回复！

如果本文对您有帮助，还请多多点赞👍、评论💬和收藏★！您的支持是我更新的最大动力！将持续更新工程！