


敬请关注 4月22日上午10点  
Keysight在线研讨会




global sources

热搜排行，点一下又不会怎样。 3 文章 搜索 高级搜索

热门搜索： 无线路由器拆解   Allegro16.3   EXCEL常用技巧   特权同学   2015IIC China

博客大赛2nd，模拟、电源、EDA  
约会IIC2015，分享收获赢好礼  
2014年度博主评选—谁是MVP？  
每天登陆1秒签到，赚E币换礼！

EDN电子技术论坛>博客>小墨同学博客>【博客大赛】零基础学FPGA（二十六）必会！从静态时序分析到SDRAM时序收敛（上篇）



小墨同学的博客——零基础学FPGA学习日记

博主：小墨同学

介绍：主要是写一些自己从每天的学习中收获的一些东西，不一定很有价值，但是我觉得随着我的成长，这些东西将成为我的一段美好的回忆...而且我断定，在大学还剩下的时间里，FPGA将陪我一起走过...

文章(27)   访问(72648)   评论(140)   投票(122)   订阅本博

博文列表查看方式：

摘要视图

原【博客大赛】零基础学FPGA（二十六）必会！从静态时序分析到SDRAM时序收敛（上篇）

发布时间：2015-04-01 10:46:31

技术类别：CPLD/FPGA   个人分类：FPGA/CPLD

推荐

好像小墨有这么一个习惯啊，就是每篇文章的开头总喜欢叨叨几句啊~既然如此，那我们今天也聊几句好了，总感觉直接就开学习专业知识有点让人看不进去~

今天我们就说说生活吧~时间很快啊，这都已经大三下学期了，如果有人问，我大学最留恋的是什么，我一定会说是我周围的人，是那群积极向上而又逗比的人才使我的大学生活充满欢乐。所以我一直想，如果再给我两年上大学的时间该多好~虽然从老师那里没有学到东西，因为小墨是几乎从来不听课的，所有的课程几乎都是自学，当然可能跟个人有关，上过大学的都知道那种几个班一起上课是种什么样的场景，所以逃课，几乎成了我的家常便饭。并不是希望大家学我哈，每个人都有自己的学习方式嘛，重要的不在于你听不听课，而是自己学会了什么技能嘛~



大学里学习固然重要，但也不能忽视了它美好的东西。在这里我也奉劝那些未来的工程师们，不要把大量的时间投到实验室里，整天十几个小时的对着电脑，每天很晚才睡，其实如果时间久了，你就会发现，这些人一般不怎么爱说话，精神恍惚，缺乏精气神，本来好好的一个20岁的小伙子，整天没精打采的也不像样子是不。因为我可是亲眼目睹身边人的样子的。有时候还是需要拿出点时间，打扮下自己，去外面的世界走走，看一本好书，去健身房练一身强健的体魄，多陪陪女朋友，练练唱歌，学一门能够让人一眼记住你的才艺等等。要知道，决定你未来的，智商只占很少一部分，多锻炼一下你的情商，更有助于我们的成长。有一天当我们离开这里的时候，你会回忆起来的，是这里的美好，而不仅仅是一个学习机器....

博主资料

我是一名大二的在读本科生，认识FPGA没多久，但是我觉得挺吸引我的，希望能和爱好这一行的同学一起努力，让我们的大学不再迷茫，也希望网站的各位前辈多多指点，希望能与你们共同进步！也欢迎各位同行加我 小墨同学 QQ号 584642877 微信号 LYZ1023569

个人文章分类

FPGA/CPLD(27)

2015年02月 (1)   2015年01月 (3)  
2014年12月 (1)   2014年11月 (3)  
展开

博客大赛2015：突破自我，整点大的 HOT!

最新评论

- 雪龙523 21:30 03-29  
好好学习一下，精辟
- xu疯子 10:37 03-26  
继续加油，跟你一起学习！
- wjfbblack 13:24 03-25  
看来小墨同学的FPGA工程已经调好了，...
- 45du\_\_ 16:19 03-23  
最近一直在搞SDRAM，翻看了你以前的...
- 小学生来了 17:21 03-09  
好，学习



当然说这些也是一点个人见解了~不喜勿喷。每个人都有自己的时间安排，只要能让自己的大学过的问心无愧，充实而又美好，那么我们这么多学费就没白交哈~

下面我们进入正题，今天我们讲时序

一、从静态时序分析说起

我理解的静态时序分析，就是我们在不加激励的情况下，通过对电路进行时序的延迟计算，预计电路的工作流程，对电路提出我们需要的一些约束条件，比如我们需要从A寄存器到B寄存器的延迟不能大于10ns，如果我们不添加时序约束，综合工具可能会有好几条路径，按照它自己的要求来布局布线，那么从A寄存器到B寄存器的时间就有可能是20ns或者15ns之类的路径，而我们需要的是不能大于10ns，因此，我们需要添加时序约束，再根据特定的时序模型，使我们的系统达到设计要求。

这里的时间延迟可能包括我们的PCB板的走线延迟，也包括FPGA内部的走线延迟，时钟延迟等等，这里我们添加的时序约束，就相当于我们在没有下板之前，通过计算，来模拟仿真下板后的延迟情况，并通过时序报告查看有哪些违规路径，以及建立保持时间不平衡的情况，通过时序优化使其达到一个稳定状态的过程。

小墨在做SDRAM时序约束的时候深有体会，即使我们老老实实的按照时序计算公式将延时计算出来添加到工程中去，时序报告也显示时序收敛并且也几乎达到了建立保持时间的平衡状态，前后仿真也没什么问题，以为一切就绪了，但是下板后就是跑不起来，最后我还是改动了一下约束参数，有时候甚至改动幅度与计算幅度相比较较大，当然还是在保证时序收敛的情况下，下板之后竟然跑起来了，可能是我第一次接触时序这个高大上的概念，经验不足吧，有的时候真是碰运气....当然等会我将源码共享的时候，大家还需要自己进行时序约束，因为我们的FPGA芯片型号，PCB布局,SDRAM型号都不太一样，那么跑起来是件很困难的事情，所以大家还需要自己亲自学一下这部分的知识，自己去摸索。

我们直接来看这个图，

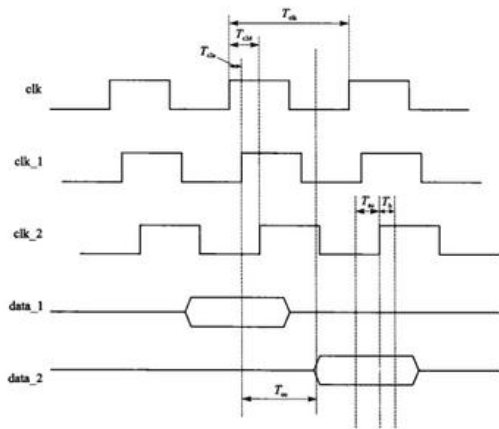


图 4.8 数据与时钟的关系

要看懂这个图，当然还需要有一个模型

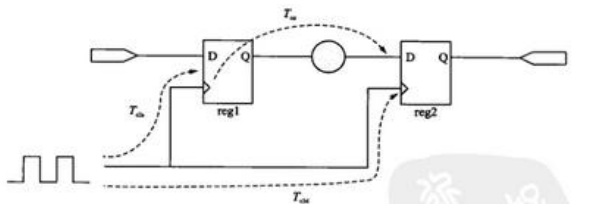


图 4.7 数据传输的寄存器模型

时序图上的第一个时钟，即我们的源时钟，或者说是我们的晶振时钟，从晶振到达源寄存器，这段路程肯定是有延迟的，我们把这段延迟叫做Tc2s，也即我们的clk1。

时序图上的第二个时钟，是我们的源时钟到达目的寄存器的时钟，即我们的锁存寄存器的时钟，它到目的寄存器的延迟，我们称为Tc2d，也即我们的clk2

clk1 到达源寄存器的时候，将输入的数据锁存到源寄存器，目的寄存器没有锁存值，等到第二个时钟周期，clk2 到达目的寄存器的时候，将源寄存器的输出数据锁存，数据必须在clk2 到达之前保持稳定，从数据到达目的寄存器到clk2到达这段时间，称为建立时间，当然，数据被锁存之后，还需要保持稳定一段时间，这段时间称为保持时间。



阅读排行

- 零基础学FPGA(一) 关于我和FPGA
- 零基础学FPGA(十四) 新的开始
- [博客大赛]零基础学FPGA(十五) 先来说V...
- 零基础学FPGA(二) 关于触发器
- 零基础学FPGA(三) Verilog语法基础基...

小墨同学的好友



小梅哥



EDNChina



coyoo



MamBa24



ilove314

[更多好友>>](#)

小墨同学的小组 加入小组

**EDN助学—FPGA/CPLD**

学习小组

《深入理解Altera FPGA应用设计》书友会

成员17369名

创建者: ilove314

成员279名

创建者: 胖螃蟹

精彩文章

推荐博客

- 人脸识别算法新进展：隐瞒年龄，不可能！
- 嵌入式开发：请不要屈服于质量，速度，成...
- LED小台灯终极升级
- 【博客大赛】零基础学FPGA（二十六）必...
- 【博客大赛】FPGA实战演练逻辑篇9：FPGA...
- 【博客大赛】板上系统逐渐过时，片上系统...
- 一个工程师说走就走的西藏之旅（五）
- 【博客大赛】FPGA实战演练逻辑篇4：Verilo...

最新网络博文

- Energia 发布 0101E0015 版
  - 苹果终于开始行动！
  - 电容测量注意事项
  - 【博客大赛】FPGA实战演练逻辑篇9：FP...
  - TimeQuest User Guide 翻译4
- [更多博文>>](#)

从数据被源寄存器锁存开始，到数据出现在目的寄存器的这段时间，我们称为数据的传输时间Tco

我们做这样一个考虑，如果Tco很长，也就是说，数据从被源寄存器捕获，到出现在目的寄存器的延时时间较长，甚至长到clk2到达目的寄存器的时候还没有到达，那么，我们就说这段路径，建立时间违规，数据不能被正确捕获，这时候，我们就可以给其添加约束，让数据在多少纳秒之内到达，保证数据被捕获，然后综合工具就会根据我们的要求，寻找最佳路径，从而保证建立时间足够长。

同理，假设Tco很短，短到clk2到来的时候，本来应该被捕获的数据，还没有保持一定的稳定时间，就被下一个要捕获的数据给覆盖了，或者说我们捕获到的是下一个要捕获的数据，从而造成保持时间违规

所以，Tco的时间既不能太长，也不能太短，必须在一个合理的范围内，尽量让目的寄存器的捕获时钟在数据的有效窗口内，所以，建立保持时间必须满足下列公式，公式的由来完全来自于时序图

建立时间必须满足下面的公式：

$$T_{clk} + (T_{dd} - T_{ds}) > T_w + T_{su} \text{ 即 } T_{co} < T_{clk} - T_w + (T_{dd} - T_{ds})$$

保持时间必须满足下面的公式：

$$T_h < T_{co}$$

其中，Tc2d - Tc2s叫做时钟偏斜，这两个公式，是我们下面分析SDRAM时序的两个重要公式

再来看一下这两个公式，既然我们要做时序分析，得要知道我们要分析的是哪一部分吧，Tclk我们无法改变，就是系统时钟嘛，时钟偏斜我们也改变不了，因为对于一个特定的FPGA器件，时钟偏斜相对固定，所以我们可以改变的就是Tco，怎么改变呢，就是添加我们的时序约束，让综合工具按照我们的要求寻找符合我们约束的路径，从而满足建立保持时间的要求。

二、输入输出模型以及IO约束

1、先来看输出模型

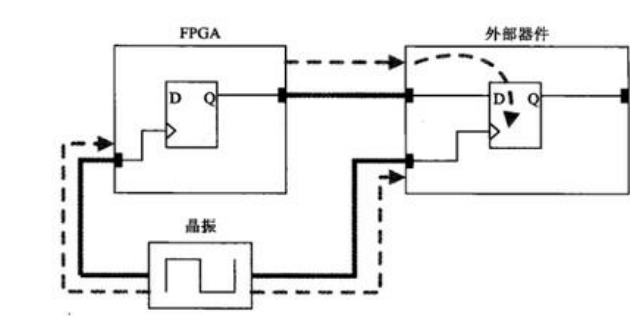


图 4.38 输出延时路径

当然这个输出模型并不是很符合我们的FPGA和SDRAM的输出模型，是我从网上找来的一张图片，不过没关系，做一下修改就好了。

我们的源时钟需要进入FPGA的PLL中，然后输出给我们的FPGA做内部时钟，在输出到外部做SDRAM的工作时钟，所以上图中，晶振到外部器件的时钟路径，应该是PLL的输出到SDRAM的输出路径

还有，我们之前做的静态时序分析，是基于在FPGA内部的，所以数据的延迟时间只有Tco一个参数，然而，我们的这个输出模型，是源寄存器在FPGA内部，而目的寄存器在我们的SDRAM内部，所以，我们的数据输出路径的延迟，还需要加上一部分，即我们的外部PCB走线延迟

这样，我们的输入延迟 = 数据在FPGA内部传输延迟 Tco + 外部PCB走线延迟

带入我们的静态时序分析的建立时间公式得

Tco < Tclk - Tsu - 外部PCB延迟 + 时钟偏斜

由于是输出模型，所以，Tco即我们的约束部分，Tsu即SDRAM的建立时间，又因为，altera官方提供的资料，不用时钟偏斜这一概念，用的是时钟网络延时，

即时钟网络延时 = - 时钟偏斜，即 Tc2s - Tc2d

所以，我们得出输出模型的一个最终公式

Tco < Tclk - Tsu - 外部PCB延迟 - 最大时钟网络延时

其中 Tsu + 外部PCB延迟 + 最大时钟网络延时 称为最大输出延时

也即 Tco < Tclk - 最大输出延时

我们到时候给时序分析工具添加的，就是这个最大输出延时，有人会问，为什么我们添加的不是 Tclk - 最大输出延时，这个问题我之前也想过，我理解的就是，最大输出延时是真实存在的，当我们下板之后，这些延时会直接影响系统的，而Tclk确实固定的，因此，我们只需要算出最大输出延时，其实Tco就已经确定了，综合工具会根据这个确定值寻找路径

同理，我们根据静态时序分析的公式算出最小输出延时

即Tco + 外部PCB走线延迟 + 时钟偏斜 > Th

即 Tco > Th - 外部PCB走线延迟 + 最大时钟网络延时

其中，最小输出延时 = Th - 外部PCB走线延迟 + 最大时钟网络延时

即 Tco > 最小输出延时

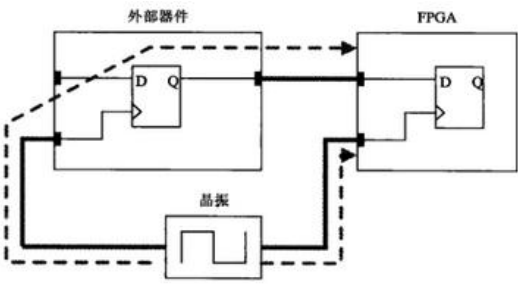


图 4.37 输入延时路径

输入模型无非就是跟输出模型相反罢了，原来FPGA作为输出，现在作为输入，还是要考虑我们之前静态时序分析的时候推出的那两个公式

即  $T_{clk} + \text{时钟偏斜} > T_{co} + \text{外部PCB走线延时} + T_{su}$

那么  $T_{co} + \text{外部PCB走线延时} + \text{时钟网络延时} < T_{clk} - T_{su}$

小于号左边就是我们的最大输入延时，这里的Tco就不再是FPGA内部的了，而是数据经过SDRAM的延时了。Tclk就是FPGA的内部时钟了，Tsu当然也是FPGA的建立时间。到了这个地方我就有点不懂了，既然是输入延时，为什么不把FPGA的建立时间考虑在内呢？要知道我们在推输出延时的时候也是把SDRAM的建立时间考虑在内了啊，为什么这里就不把FPGA的建立时间考虑在内呢？

我是这么理解的，会不会是因为我们只需要算数据到FPGA引脚的延时就可以了，剩下的路径，包括数据从FPGA引脚到数据被锁存这段时间，也就是已经包括了FPGA的建立时间在内了的缘故，这段时间是综合工具根据我们的输入延时来分配路线的。

这个问题，希望看到的大神给个意见吧

当然了，最小输入延时也是根据那个公式求得

即  $T_{co} + \text{外部PCB走线延时} + \text{时钟网络延时} > T_h$

即最小输入延时  $> T_h$

3、下面我们做一个总结

最大输出延时 = 外部器件的Tsu + 外部PCB最大延迟 + 最大时钟网络延时

最小输出延时 = 外部器件的Th - 外部最小PCB走线延时 + 最大时钟网络延时

最大输入延时 = 外部器件最大Tco + 外部PCB最大走线延时 + 最大时钟网络延时

最小输入延时 = 外部器件最小Tco + 外部PCB最小走线延时 + 最小时钟网络延时

下面再来解释一下最大最小问题，看到这大家肯有点看不懂了，刚才还觉得可以，加了一些最大最小就看不懂了，其实我们要做的输出，不仅仅是有一个引脚输出，拿我们的SDRAM来说，有12根地址线，还有16根数据线还有其他的控制信号线要输出，那么，我们应该选其中最长的，或者最短的来考虑，只要最长的或者最短的满足要求了，其他的自然也就满足要求了，即我们用最坏的打算来考虑我们的约束路径。

还有最大最小时钟网络延时，由于这里我们只有一跟时钟线，就无所谓最大最小了，这么说就是为了规范

最大最小外部器件的Tco，数据经过SDRAM的时候，并不是时间都是一样的，总有些快慢问题，这里我们也取极限，就是数据经过SDRAM的最大时间和最小时间，这个时间在datasheet里面是有参数的

细心的朋友可能会发现 最小输出延时计算的时候，为什么是减去 最小PCB走线延时呢，为什么不是最大？为什么不是加上最大时钟网络延时呢？为什么不是最小？其实，时序分析工具分析的时候是按照一定的公式计算数据到达时间和数据需要时间的，这个公式见下图

Clock Hold Slack Time=Data Arrival Time—Data Required Time

Data Arrival Time=Latch Edge+Clock Network Delay to Source Register+uT<sub>co</sub>+Register-to-Pin delay

Data Required Time=Latch Edge+Clock Network Delay—Output Minimum Delay of Pin

在计算保持时间余量，数据需要时间的时候，公式是减去 output minimum delay of pin 为什么是减去呢，按理说应该是加上我们的输出最小延时才对，小墨当时也不理解，所以就当是记住吧，也就是说

这里的 output minimum delay of pin = - 输出最小延时

这样理解的话，那么输出最小延时越大，那么output minimum delay of pin 就越小了，也就是为什么是减去最小PCB延时的原因吧

后面我们到时序分析工具里面看时序报告图的时候就会发现，如果我们约束的最小输出延时是负数的话，里面显示的却是正数，所以，我们就暂时记住好了，因为真的是不好理解。

三、参数计算



根据我们刚才推断出来的那几个公式，我们就可以进行参数计算了，先来看看我们需要什么参数

输出模型的时候，我们需要的是SDRAM的建立时间，保持时间，外部PCB走线延时，输入模型的时候，我们需要的是外部器件的最大最小Tco

SDRAM的建立保持时间可以从datasheet里面找到

Data-out hold time	DOH	2.8	-	2.8	-	2.7	-	2.7	-	2.7	-	2.7	-	2.7	-	2.7	-	ns	1
Data-input setup time	DIS	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	2	-	2	-	ns	1
Data-input hold time	DIH	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	1	-	1	-	ns	1
Address setup time	AS	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	2	-	2	-	ns	1
Address hold time	AH	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	1	-	1	-	ns	1
CKE setup time	CKS	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	2	-	2	-	ns	1
CKE hold time	CKH	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	1	-	1	-	ns	1
Command setup time	CS	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	1.5	-	2	-	2	-	ns	1
Command hold time	CH	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	0.8	-	1	-	1	-	ns	1

根据器件型号，我们选择建立时间为 1.5 ns ,保持时间为 0.8ns，外部PCB走线延时这个不怎么好算，只能估计吧，因为板子不是我画的，长度也不知道，根据经验，我们知道，信号在PCB上的延时大概为0.18ns /1000mil，那我就估计为最大0.3ns，最小0.1好了

然后是时钟网络延时，也就是源时钟到达源寄存器 - 到达目的寄存器的值，由于我们的时钟是总PLL发出的，时钟从FPGA管脚输出后再送给SDRAM，因此这个值一定是负值，而且也是时钟在PCB上的走线延时，这个值我们估算为-0.1ns好了，因为真的没法算我觉得。

然后是数据经过外部器件的最大最小延时，这个值我们在SDRAM的理论篇里面说了，就是Tac参数和Toh参数，不懂得朋友可以到前面去补补课，这个值我们在datasheet里面也找得到

HY57V641620HG(L)T-7							
	CAS Latency	t RCD	t RAS	t RC	t RP	t AC	tDH
143MHz (7ns)	3CLKs	3CLKs	6CLKs	9CLKs	3CLKs	5.4ns	2.7ns
133MHz (7.5ns)	3CLKs	3CLKs	6CLKs	9CLKs	3CLKs	5.4ns	2.7ns
100MHz (10ns)	2CLKs	2CLKs	5CLKs	7CLKs	2CLKs	6ns	3ns

所以我们取最大Tco 为5.4ns 最小Tco为2.7ns

经过计算得我们的约束参数

最大输出延时 =外部器件的Tsu+外部PCB最大延迟+最大时钟网络延时

=1.5+0.3-0.1=1.7ns

最小输出延时 = 外部器件的Th - 外部最小PCB走线延时 + 最大时钟网络延时

= - (0.8 - 0.1 - 0.1 ) = - 0.6ns

最大输入延时 = 外部器件最大Tco + 外部PCB最大走线延时 + 最大时钟网络延时

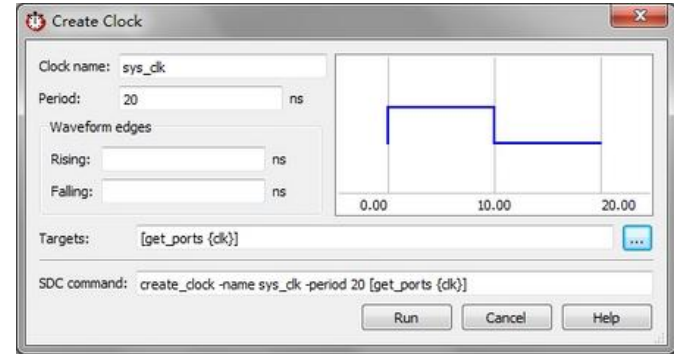
= 5.4 + 0.3 - 0.1=5.6ns

最小输入延时 = 外部器件最小Tco + 外部PCB最小走线延时 + 最小时钟网络延时

= 2.7 + 0.1 - 0.1 = 2.7ns

四、全局时钟约束

由于我们的时钟是由晶振源传到PLL，然后再由PLL输出给SDRAM和FPGA内部寄存器的，因此我们需要对输入的时钟进行约束，约束其为50M的时钟，并对应好我们的端口时钟clk，即约束我们的输入时钟为50M，并给这个时钟取一个在时序分析时钟的名字，即sys\_clk，关于timequest的使用，大家还需自行学习，这里仅仅只是说约束方法



而对于PLL输出时钟的约束，我们仅需在.sdc文件相应的位置，手动输入约束即可，仅需要一条语句 ,这条命令输入进去以后，时序分析工具会自动的为我们约束好PLL的输出时钟

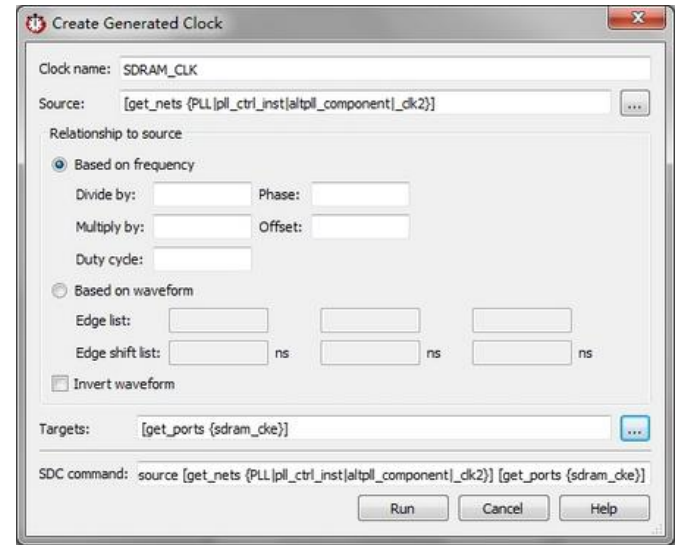
```
# Create Generated Clock
*****
derive_pll_clocks |use_tan_name
*****
```

我们可以通过查看时序报告，可以看到，我们的时钟都跑到了我们约束的条件，如果我们不加这条语句的的话，那么PLL的输出时钟是达不到这个要求的，那样的话，我们的时钟都不稳定，更别提系统稳定性了

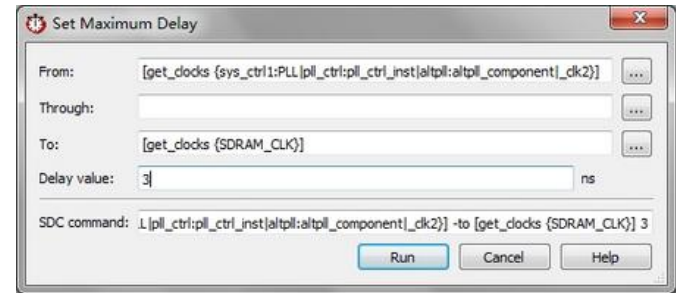
Clock Summary										
Clock Name	Type	Period	Frequency	Rise	Fall	Duty Cycle	Divide by	Multiply by	Phase	
1. SDRAM_CLK	Generated	10.000	100.0 MHz	7.291	12.291		1	1		
2. sys_clk	Base	20.000	50.0 MHz	0.000	10.000					
3. sys_clk1.PLL[pll_ctrl_inst[altpll_component]_clk0]	Generated	20.000	50.0 MHz	0.000	10.000	50.00	1	1		
4. sys_clk1.PLL[pll_ctrl_inst[altpll_component]_clk1]	Generated	10.000	100.0 MHz	0.000	5.000	50.00	1	2		
5. sys_clk1.PLL[pll_ctrl_inst[altpll_component]_clk2]	Generated	10.000	100.0 MHz	7.291	12.291	50.00	1	2	262.5	

五、关于虚拟时钟

上图中，细心的朋友可能会看到，第一行有一个SDRAM\_CLK，这个时钟是怎么来的呢？这个时钟，其实就是我们给外部的芯片模拟了一个时钟，因为外部器件的时钟是我们PLL生成，送到外部管脚去的，并不是我们FPGA的内部时钟，既然不是内部时钟，那么我们就无法对其进行约束，我们只能定义这么一个虚拟的时钟，来尽可能模拟SDRAM的工作时钟。例如，输入的时候，源寄存器在SDRAM中，目的寄存器在FPGA中，数据的传输是在SDRAM工作时钟下进行的，但是时序分析工具不知道这个时钟，所以我们定义一个虚拟时钟，连接好它的路径，即从哪输出，输到哪，这里的话就是从PLL的clk2输出，到SDRAM的时钟输入引脚了。



既然是模拟，那我们也应该考虑到这个时钟从PLL输出，到FPGA引脚的这段时间的延时，我们对其进行最大，最小延时约束



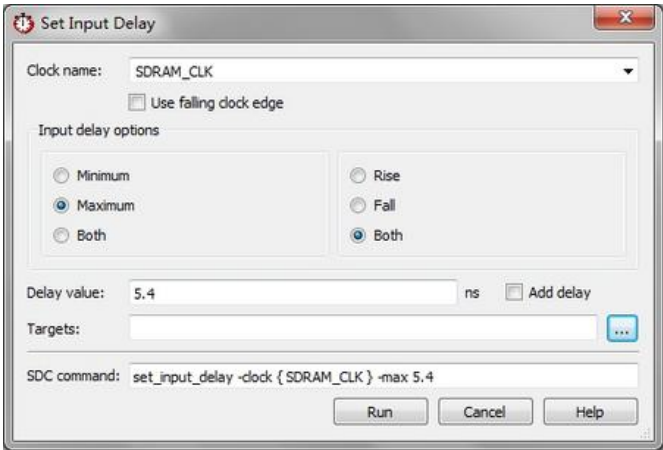
同理，我们约束最小值为0ns，这里就不做演示了，方法是一样的

六、I/O时序约束

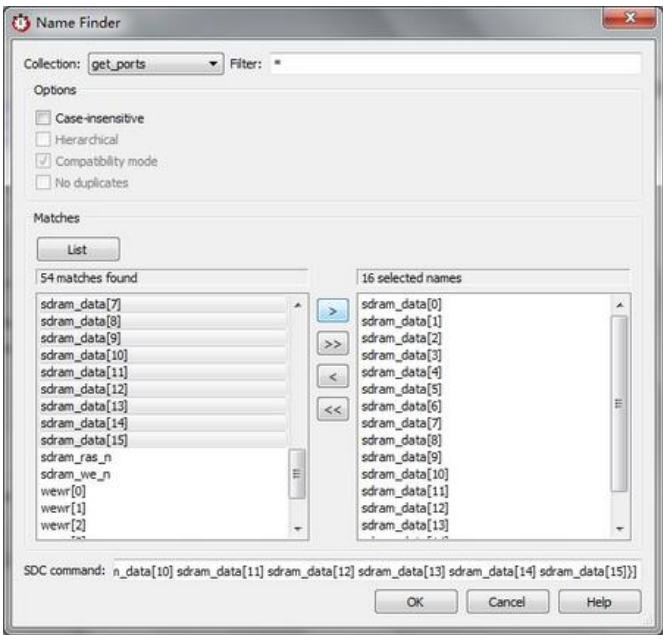
由于篇幅所限，这里就不能一点点的做了，只举个例子吧，做最大输入延时的约束，我们点击 constraints ——> set input delay

时钟名字我们选的就是我们生成的虚拟时钟，为什么呢？因为我们做的是IO约束，即FPGA和SDRAM互相交换数据，既然交换数据肯定是建立在时钟的基础上的，这个时钟，就是虚拟时钟。这个地方比较难懂了，我理解的也不是太好。毕竟我们是要看得是时序报告，时序报告就是建立在虚拟时钟的基础上的一个模拟过程。先理解到这里吧，也不知道该怎么解释了，还请看到的大神给指点一二吧~

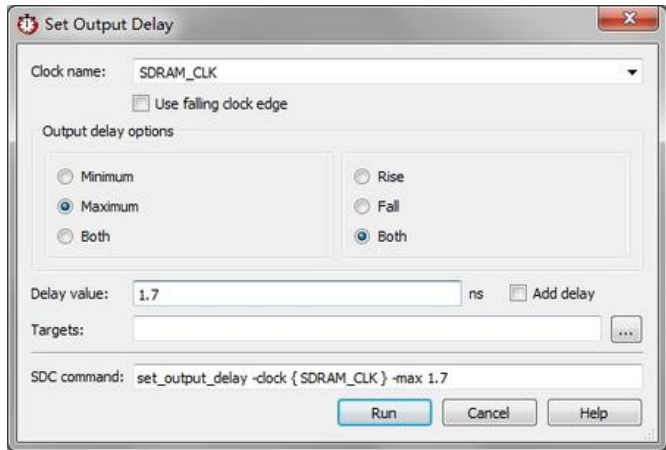
下面我们做最大输入延时的约束,设置我们最大的输入延时为5.4ns

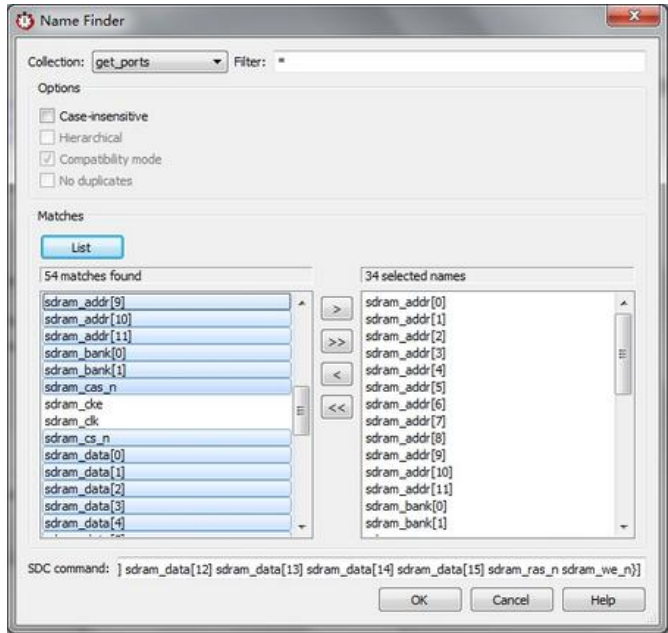


设置输入路径，因为只有SDRAM\_data路径是双向的，做输入的时候仅考虑这个路径即可



然后我们再做一个输出最大延时的约束，输出最大延时为1.7ns，输出路径包括除了时钟和时钟有效信号sdram\_cke信号外的所有的SDRAM引脚，当然也包括sdram\_data信号，为什么不包括cke信号呢？因为cke信号自始至终没有多少次的变化，也就是说对时序要求不严格，对于这种信号，类似还有片选信号，数码管片选，位选等，我们只需对其进行false路径约束即可，即对时序没有严格要求的路径





剩下的路径我就不再做了，都是一样的，将所有路径约束好之后，我们需要开始调sdram工作时钟的相位偏移了

由于篇幅所限，剩下的内容我放到下篇中讲，谢谢大家~

以后小墨同学的文章将在EDNchina和北京至芯科技官方技术论坛同时更新，至芯科技的官方技术论坛也有很多的资料供大家下载，里面也有小墨的专题板块，希望广大网友和爱好者的大力支持~

<http://www.fpgaw.com/>

小墨同学《零基础学FPGA》学习专版 一个大三学生学习之路的真实记录

<http://www.fpgaw.com/thread-78527-1-1.html>

下面是至芯科技官方网站，夏宇闻教授和那里的老师都超好的~有意愿接受FPGA培训的可以找工作人员咨询

<http://www.zxopen.com/>

以后小墨同学的教程会跟这款开发板配套，想跟小墨同学一起学习，一起进步的可以考虑一下购买一块开发板，毕竟学习要舍得投资嘛，下面是开发板淘宝链接

<http://item.taobao.com/item.htm? ... ;abbucket=13#detail>

标签： 博客大赛



分享到：



我喜欢

有[1]名读者喜欢此文

阅读(178)

评论(0)

收藏


打印

推荐到小组

转发到我的博客 (0)

下一篇：【博客大赛】零基础学FPGA（二十七）必会！从静态时序分析到SD...

最近到访博主

  
KPUHD

  
szm3

  
EDNChina

有[0]名读者发表评论 - 【博客大赛】零基础学FPGA（二十六）必会！从静态时序分析到SDRAM时序收敛（上篇）


我来评论 - 【博客大赛】零基础学FPGA（二十六）必会！从静态时序分析到SDRAM时序收敛（上篇）

昵称： 大灰灰

评论：



你还可以输入1000字

分享：

[返回博客首页](#)

模拟设计 | 电源技术 | 嵌入式系统 | 微处理器与DSP | 可编程器件 | 测试与测量 | EDA工具与服务 | 通信 | 医疗电子 | 消费电子设计

工业电子 | 汽车电子 | 新闻 | 热点专题 | 论坛 | 小组 | 博客 | 微博 | 下载 | 视频 | 在线研讨会 | 杂志订阅 | 历史文章回顾

About EDN China - 编辑部（编辑计划） - 广告部 - 发行部 - 读者往来 - 隐私政策 - 使用条款 - 网站导航 - 电子网站 - 帮助 - EDNChina- edn-china - 手机版 - 手机版介绍

2015 EDN China -- Use of this website is subject to its terms of use.

京 ICP证090436号 | 京ICP备14018902号-1 | 京公网安备110105019137 | 经营性网站执照信息

EDN.com | EDN Asia | EDN Japan | EDN Europe | EDN Australia | EDN Taiwan | 电子工程专辑 | 国际电子商情 | DatasheetsChina

机器人网 | Global Sources Direct



