Signal Tap II 的使用方法:

Signal Tap II——实时信号分析调试工具(在线逻辑分析仪)

SignalTap II 全称为 SignalTap II Logic Analyzer,从名称上可以看出它本质上就是一个逻辑分析仪,不过它是免费的:硬件部分用 FPGA 内部资源实现,软件集成在了 Quartus II 中。它可以对 FPGA 的输入输出管脚及任意内部信号进行逻辑分析,能迅速分析电路某些信号的运行状态和锁定设计问题,方便 FPGA 开发者对设计进行调试。

SignalTap II 占用的是 FPGA 的内部资源(以 RAM 资源为主),其工作原理为:将采样的信号存储到内部 RAM 中,通过 JTAG 传到 PC 机上,在软件上显示供开发者观察分析。其采样深度会受 FPGA 内部 RAM 资源容量的限制,但对采样深度要求不是很高的信号来说,已经足够使用。

逻辑分析仪与在线逻辑分析仪(signal Tap II)的比较:逻辑分析仪存在的问题:

1、价格昂贵

即使是最便宜的性能很差的逻辑分析仪,价格也要几百上千。而稍微好点的如 Agilent 等品牌的逻辑分析仪,更要上万甚至十几万,对于一个 FPGA 开发者,这简直就是天价。

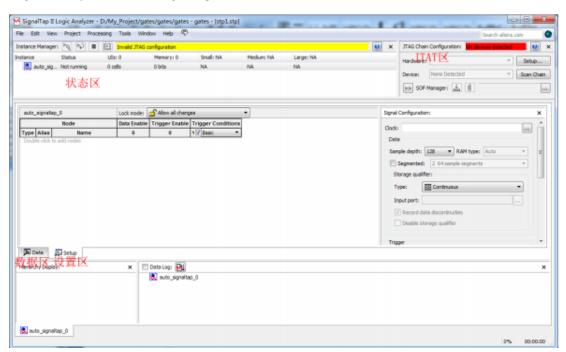
2、信号引出不方便

如果要使用逻辑分析仪进行信号分析,我们需要将待分析的信号通过外部引脚引出,这个过程比较不方便,如果需要引出的信号数量较多,比如说引出一个 32bit 总线,那更是很麻烦的一项工作——而且你的板子上还不一定存在那么多外部引脚。

对于 FPGA 开发者而言, SignalTap II 的出现解决了 FPGA 设计中信号分析的难题,

以下开始 signaltapII 的操作:

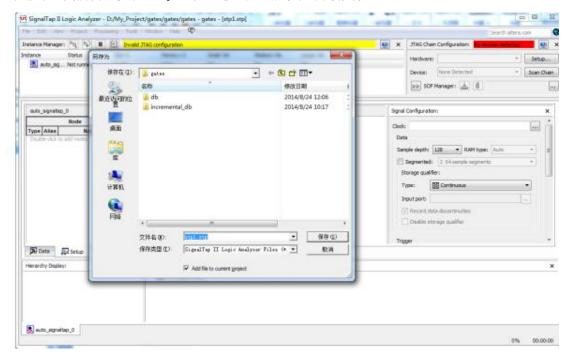
1) 创建 SignalTap II 文件。在 Quartus II 主界面选择菜单栏的 Tools->SignalTap II Logic Analyzer,打开 SigalTap II 软件



其中数据区与设置区是重叠的两个栏,上图中显示的是设置区,点击下边栏的 Data,我们把显示切换为数据区,如下图所示,可以看到数据区还没有任何数据,因为我们还没有采样。



在 SignalTap II 主界面下选择 File->Save (或直接 Ctrl+S) 保存文件,点击后得到下图,名称就用其默认名称 stpl,点击保存,完成 SignalTap 文件的保存。注意观察对话框的下部有一个选择 Add file to current project,即是否把文件添加到当前工程中,默认是勾选的,我们自然要把 stpl 添加进工程,所以无需改变设置。



这时会弹出下图所示的对话框,提示数据和触发输入是空的,因为我们没有添加,自然是空的,点击 OK 即可



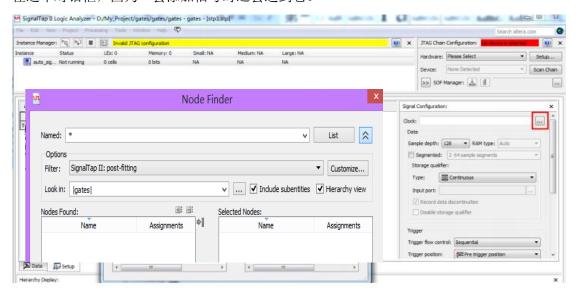
这时又会弹出下图所示的对话框,提示是否为当前工程使能 stpl, 按照前面的讲解,选择 Yes, stpl 就被当前的工程使能了。至此 SignalTap II 文件的创建和添加就完成了。

【Tips】一个 Quartus II 工程中可以有多个 SignalTap II 文件,但只能使能一个。我们在完成 SignalTap II 的创建与设置后,要重新编译工程,也只有被使能的 SignalTap II

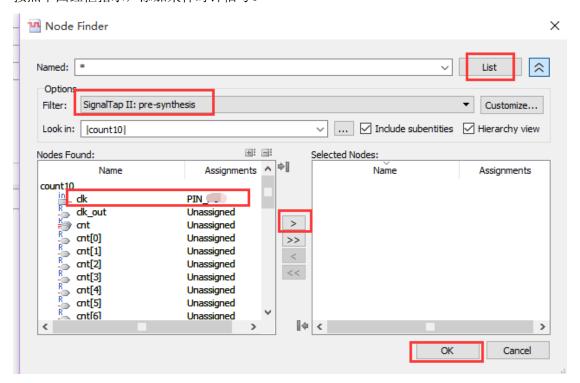
文件会被编译进工程中,占用 FPGA 的资源。

- 2) 设置 SignalTap II 文件。一般来讲,一个 SignalTap II 文件在使用前需要进行的设置有:添加采样时钟、添加信号、设置采样方式、设置触发方式、设置触发条件。
- a) 添加采样时钟

将系统时钟作为采样时钟,开发板上的时钟为 16.384MHz。方法是在 SignalTapII 软件的主界面,点击设置区下图红框所示的位置,就会弹出 Node Finder 对话框,我们要记住这个对话框,因为一会添加信号时还会遇到它。



按照下图红框指示,添加采样时钟信号。



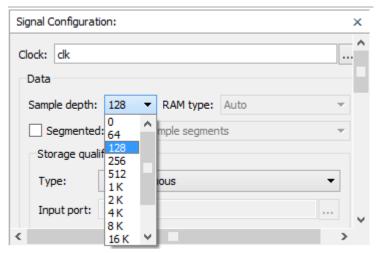
b) 添加观测信号

在设置区的左边空白部分右键单击(或者双击空白处),选择 Add Nodes,采用 Filter 的 pre synthesis 方式筛选出信号,并将除 clk 和 rst 外的信号添加到 Selected Nodes 中。

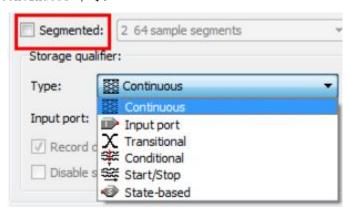
添加方式和添加采样时钟类似。

c) 设置采样方式

第一步是设置采样深度。采样深度的设置位于设置区右边的 Sample depth 选项处,深度从 0 到 128k 可选,单位为样点数。比如设置为 128,代表采样深度为 128,即可以采 128 个样点的信号,因为工程较简单,我们把深度设为 128 就足够了。



第二步是设置采样模式。采样模式分为分段采样和非分段采样(也叫循环采样),如果没有勾选上下图红框所示的 segment(分段),就是非分段采样,在信号触发后就连续采样至采样深度,即填满整个 SignalTap II 的存储。我们还要选择 Type (类型),一般设置为 Continuous 即可。



如果勾选上了 segment,就是分段采样,将采样深度分为 N 段,信号每触发一次就采样一段长度的数据,需要连续触发 N 次来填满整个存储。我们需要选择采样的段数及每段长度,如 8 16 sample segments 代表分成 8 段,每段 16 个采样点

d) 设置触发方式

设置触发流程控制为 Sequential, 触发位置为前端触发, 触发条件为 1 个 触发流程: Sequential (顺序的)-用于一般信号分析; State-Based (基于状态的), 用于

较复杂的触发控制。

触发位置:前端触发;中间触发;后端触发;决定了信号触发点在整个采样数据的位置。触发条件;最多可以设置 10 个级别的触发条件。对于非分段采样,直到满足触发条件后,才 开始正式捕捉信号;对于分段采样,只需要满足最后一级触发条件即可触发。

e) 设置触发条件

触发条件的类型分为 Basic 和 Advanced, 一般选择 Basic 即可

Node			Data Enable	Trigger Enable	Trigger Conditions	
Туре	Alias	Name	6	6	1√	Basic ▼
•		and_out	V	V		Basic
						Advanced

【注】: Don't Care 代表任意条件都触发,

Low 表示信号低电平时触发,

Falling Edge 表示信号下降沿时触发,

Rising Edge 表示信号上升沿时触发,

High 表示信号高电平时触发,

Either Edge 表示任意沿触发。

需要注意的是, 同一列中的 Trigger Conditons 属于同一级别,可以给该列中多个信号同时设置触发条件(注意这里与上面设置触发条件级别的区别!),比如给 in1、in2 都设为 Low,那么当 in1、in2 信号都为低电平时,才会触发采样。

f) 每个信号设置为 Don't care

trigger: 2015/10/10 19:31:42 #0			Lock mode: all changes							
Node			Data Enable	Trigger Enable	Trigger Co					
Туре	Alias	Name	4	4	1 ✓ Basic AND ▼					
*		result[3]	$\overline{\checkmark}$	\checkmark	■ ■					
*		result[2]	$\overline{\checkmark}$	\checkmark						
*		result[1]	$\overline{\checkmark}$	\checkmark	■ ■					
*		result[0]	\checkmark	\checkmark	- 3					

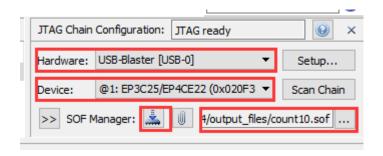
g) 保存文件。

3) 捕获信号

[1]. SignalTap 设置完成后,重新编译 Quartus II 工程

任务:观察编译前和编译后逻辑资源的应用情况,以验证 signaltap 的添加是否占用了 FPGA 的资源。

[2]. 编译完成后,在 signaltapII 操作界面中,将 SOF 文件通过 JTAG 区下载到开发板中



[3]. 点击运行



- 1 键是运行键,点击一次捕获一次信号;
- 2 键是连续运行键,点击一次捕获连续进行;
- 3 键是停止键,可以中止当前的信号捕获。

Status 栏显示着捕获状态,分为Not running (未运行)、Waiting for trigger (等待触发)和 offloading acquired data (导出捕获到的数据),此时显示 Not running,表示捕获未运行。

点击连续运行键, 可观测变化情况

