

Fabric Inserter 用户手册

(Version 1.0)

深圳市紫光同创电子有限公司

版权所有 侵权必究

文档版本修订记录

版本号	发布日期	修订记录
V1.0	2022.07.12	初始版本

目录

文档版本修订记录	2
目录	3
图目录	5
名词术语解释	7
启动方式	9
综合工具路径设置说明:	9
应用实例	9
1. 在 PDS 中添加资源文件	9
2. 打开 Inserter 软件	10
3. JtagHub 配置界面	10
4. DebugCore 配置界面	11
5. 保存 FIC 文件并插核	14
6. 另存 FIC 文件	15
7. Fabric Debugger 中导入 FIC 文件	15
3.1 用户界面介绍	16
3.2 菜单基本操作说明	17
1. File 菜单	17
2. Edit 菜单	17
3. View 菜单	17
4. Help 菜单	18
3.3 工具栏说明	18
3.4 界面功能介绍	18
主界面功能介绍	18
JtagHub 配置界面	19
DebugCore 配置界面	20
Preferences Dialog 功能介绍	34
在 RTL 中标记信号	36
1. 模糊标记	36
2. 精确标记	37
通过 Inserter 配置调试核	38

Insertter TCL Command 简介	39
Insertter TCL Command 详细介绍	39
1. Help	39
2. Ins_new	39
3. Ins_open	40
4. Ins_save	40
5. Ins_set_file	40
6. Ins_add_core	40
7. Ins_remove_core	40
8. Ins_operate	40
9. Ins_set_core	40
10. Ins_set_trig	41
11. Ins_set_unit	42
12. Ins_set_condition	42
13. Ins_set_capture	43
14. Ins_list_nets	43
15. Ins_set_net	43
16. Ins_core_info	44
免责声明	45

图目录

图 2.1 PDS 中添加 FIC 文件	9
图 2.2 Inserter 软件主界面.....	10
图 2.3 JtagHub 配置界面	11
图 2.4 Parameters 配 Trigger 置界面	12
图 2.5 Net Connections 配置界面（连线完成前）	12
图 2.6 Select Net 配置界面	13
图 2.7 Net Connections 配置界面（连线完成后）	13
图 2.8 Trigger Port 调整界面	14
图 2.9 Power On Init Parameters 配置界面.....	14
图 3.1 软件主界面	16
图 3.2 File 菜单	17
图 3.3 Edit 菜单.....	17
图 3.4 View 菜单	17
图 3.5 Help 菜单.....	18
图 3.6 工具栏.....	18
图 3.7 软件主界面.....	19
图 3.8 JtagHub 配置界面	20
图 3.9 DebugCore 配置界面	20
图 3.10 Trigger Parameters 配置界面	21
图 3.11 Trigger Input and Trigger Unit Settings.....	22
图 3.12 Capture and Storage Settings.....	22
图 3.13 Trigger Condition Setting	23
图 3.14 Net Connections 配置界面（连线完成前）	23
图 3.15 Select Net 配置界面	23
图 3.16 Net Connections 配置界面（连线完成后）	27
图 3.17 PowerOn Init Parameters 配置界面.....	28
图 3.18 Match Funciton 配置	28
图 3.19 Match Counter 配置.....	29
图 3.20 Trigger Condition	30
图 3.21 配置 Trigger Condition 布尔表达式	30

图 3.22 配置 Trigger Condition 顺序触发表达式	31
图 3.23 Capture Settings	32
图 3.24 Storage Qualification Condition 选择 All Data	33
图 3.25 Storage Qualification Condition 选择布尔表达式	34
图 3.26 Edit Preferences 窗口	35

名词术语解释

Debug Core: 调试核。

Trigger Port: 触发端口，只有当连接至 Trigger Port 的信号的值满足用户设置的触发条件时才开始进行数据捕获。

Data Port: 数据捕获端口，连接至 Data Port 的信号是 Debug Core 捕获数据的来源，只有连接至 Data Port 的信号才可以在 Debugger 软件中进行观察。

Clock Port: 时钟端口，连接至该端口的信号将被作为 Debug Core 的工作时钟同时也是采样时钟。

Reset Port: 复位端口，连接至该端口的信号将被作为 Debug Core 的复位信号，Debug Core 在复位信号为低电平时复位。

Trigger Unit: 匹配单元，用于设置触发条件。

1 软件简介

Fabric Inserter 软件主要功能是把 DebugCore 自动插入用户的设计网表中生成新的设计网表，从而使用户不需要手工在 HDL 代码中例化。

它的主要功能包括：

最大支持 15 个 DebugCore。

每个 DebugCore 最大支持 16 个 TriggerPort，每个 TriggerPort 最大位宽为 256 并支持 16 个 TriggerUnit，且所有 TriggerPort 的 TriggerUnit 之和不能超过 16。

每个 DebugCore 最大支持抓取 4096 个数据信号。

支持对 DebugCore 触发信号和触发条件等设置。

支持对 DebugCore 存储深度、数据位宽、采样设置等参数的设置。

支持 DataPort、TriggerPort 连接设置，DataPort 和 TriggerPort 可以相同也可以分开。

支持上电初始化触发和捕获属性的设置。

支持工程属性文件的导入和导出功能。

2 软件使用入门

2.1 启动方式

Inserter 需要结合 PDS 软件使用，可以通过单击菜单栏或工具栏中的 Inserter 图标启动 Inserter 软件，或通过双击 Constraints 列表中的.fic 文件启动 Inserter，需要注意的是只有在 PDS 中添加了资源文件（.v 或.vm）后才能启动 Inserter。

2.2 综合工具路径设置说明：

Insertor 软件始终使用 PDS 中设置的综合工具路径。

2.3 应用实例

下面，用一个简单的例子说明软件功能和如何使用 Fabric Inserter 软件。

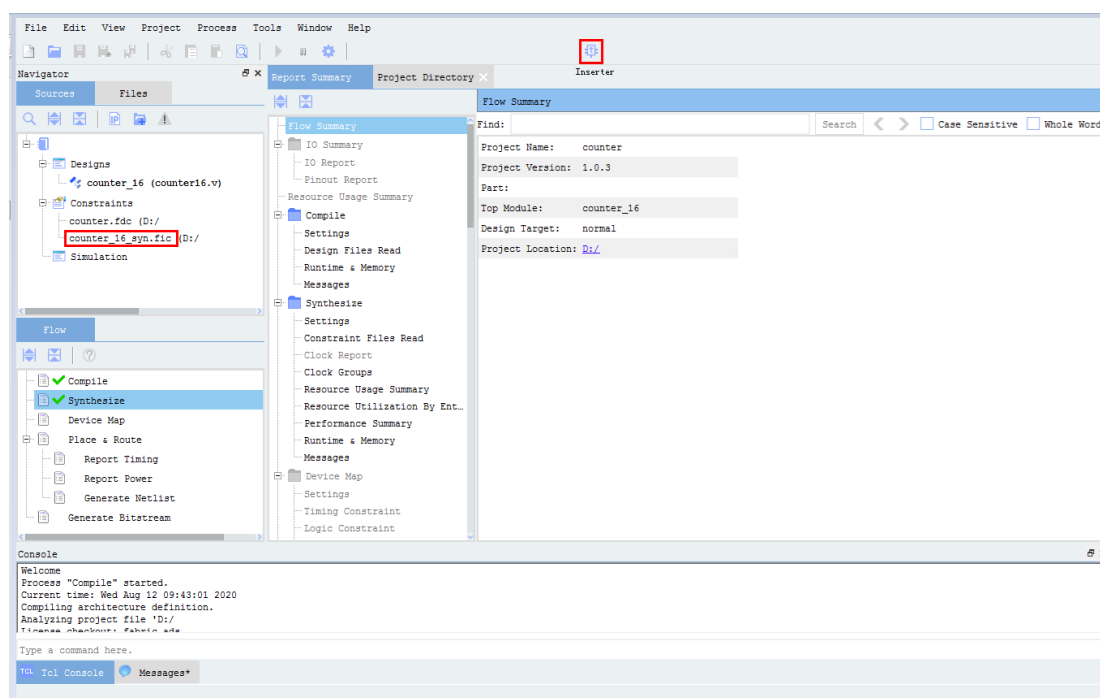


图 2.1 PDS 中添加 FIC 文件

2.3.1 在 PDS 中添加资源文件

Fabric Inserter 软件需要集成到 PDS 软件中一同使用，如图 2.1，在 PDS 资源文件中添加设计文件（.v 或.vm）及约束文件，用户可以手动添加 FIC 文件或使用 Inserter 软件自动生成的 FIC 文件。

2.3.2 打开 Inserter 软件

通过双击 PDS 约束文件列表中的 FIC 文件或单击工具栏中的 Inserter 图标可打开 Inserter 主界面，在 Inserter 中对 FIC 文件进行编辑。打开 Inserter 主界面后，Inserter 会自动加载并解析用户添加至 PDS 中的设计文件，并自动生成输出文件路径，软件主界面如图 2.2。

Inserter 软件启动后，用户仍然可以在 PDS 中进行工程属性修改、rerun 等操作。当 Inserter 检测到作为其输入网表的 ADF 文件被修改时，会弹出窗口询问用户是否重新载入文件，如果选择重新载入，则当前设置的 net 连接信息将被清空，其他核信息将被保留。

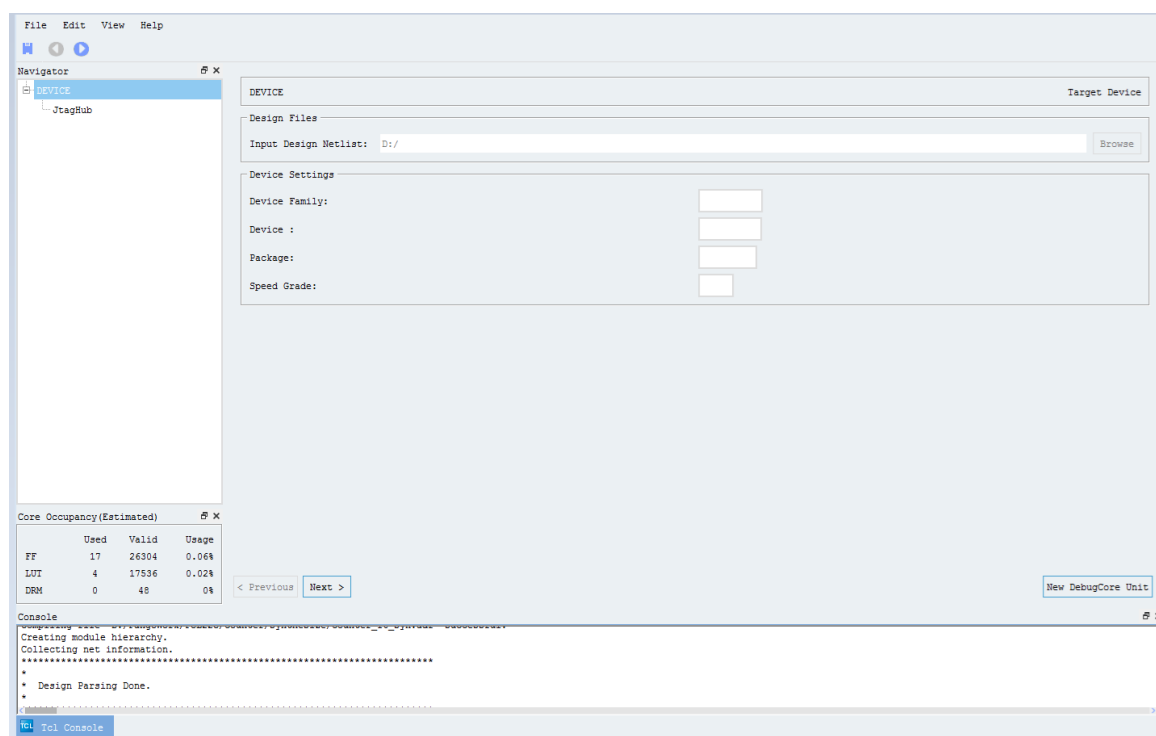


图 2.2 Inserter 软件主界面

2.3.4 JtagHub 配置界面

点击 Next 进入 JtagHub 配置界面，在这个界面中用户可做的选择为点击右下角 New DebugCore Unit 按钮添加 DebugCore，最多可添加 15 个。如不添加 DebugCore，再点击 Next 软件会默认添加一个 DebugCore。同时，用户可在该界面中指定要使用的边界扫描链，也可以指定要使用的 JTAG。

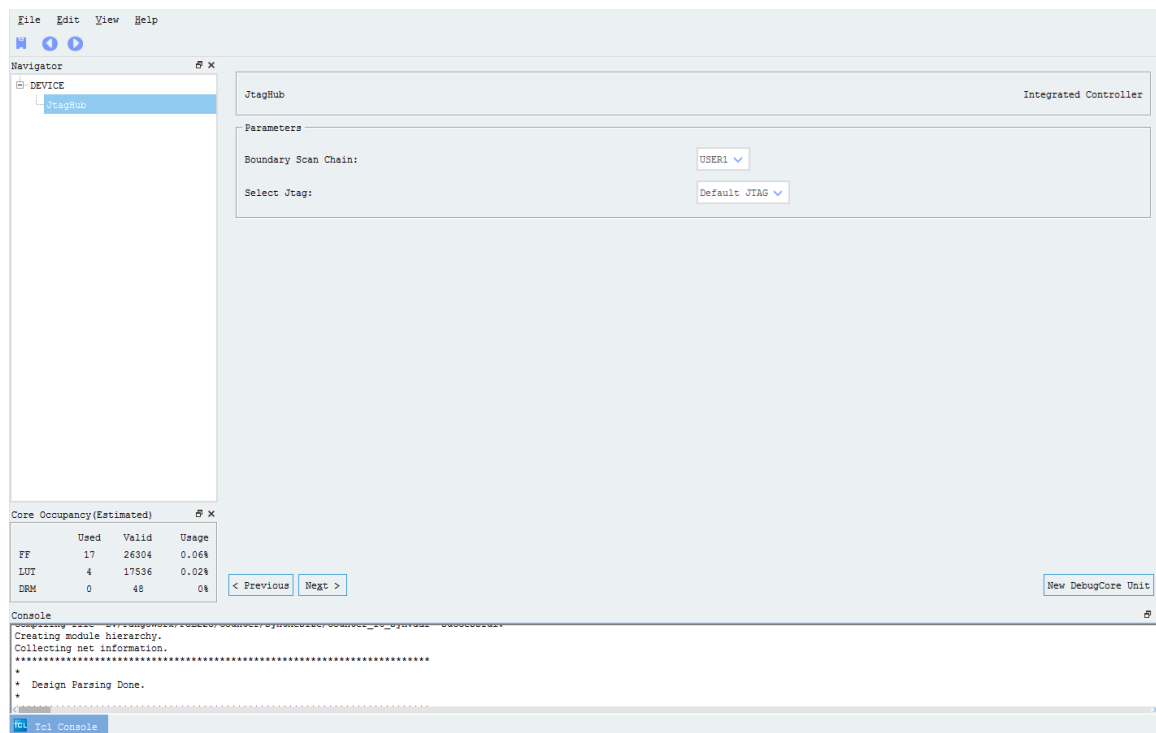


图 2.3 JtagHub 配置界面

2.3.5 DebugCore 配置界面

在配置完 JtagHub 后点击 Next 按钮进入 DebugCore 配置界面：

在这个界面里可以对 DebugCore 参数进行配置，这是 Inserter 软件功能的主要体现。一个 DebugCore 对应界面中包含以 Trigger Parameters、Net Connections 和 PowerOn Init Parameters 为标题的标签页面，分别对应 DebugCore 不同参数的配置。可以点击 Next 和 Previous 或者直接点击标签页标题进行标签页的切换。下面对各个标签进行简要说明。

1) Trigger Parameters 选项卡

Trigger Parameters 选项卡中的配置选项分为三组，分别用于对触发端口(Trigger Port)，捕获存储(Capture & Storage)和触发条件(Trigger Condition)进行相关参数配置。触发端口相关设置包括触发端口的数量，每个触发端口下挂接的匹配单元的数量及其类型等。捕获存储相关配置包括数据的采样深度，采样时钟沿等。触发条件相关配置主要是对顺序触发的最大等级进行配置。

当有存在多个 Trigger Port 时，可以通过拖动相应的 Trigger Port 页签来进行排序，或者点击 Trigger Port 页签上的“关闭”按钮来删除某一个 Trigger Port。用户也可以在 Net Connections 选项卡中对 Trigger Port 的进行排序或者删除。当仅有一个 Trigger Port 时，将不能进行排序和删除操作。

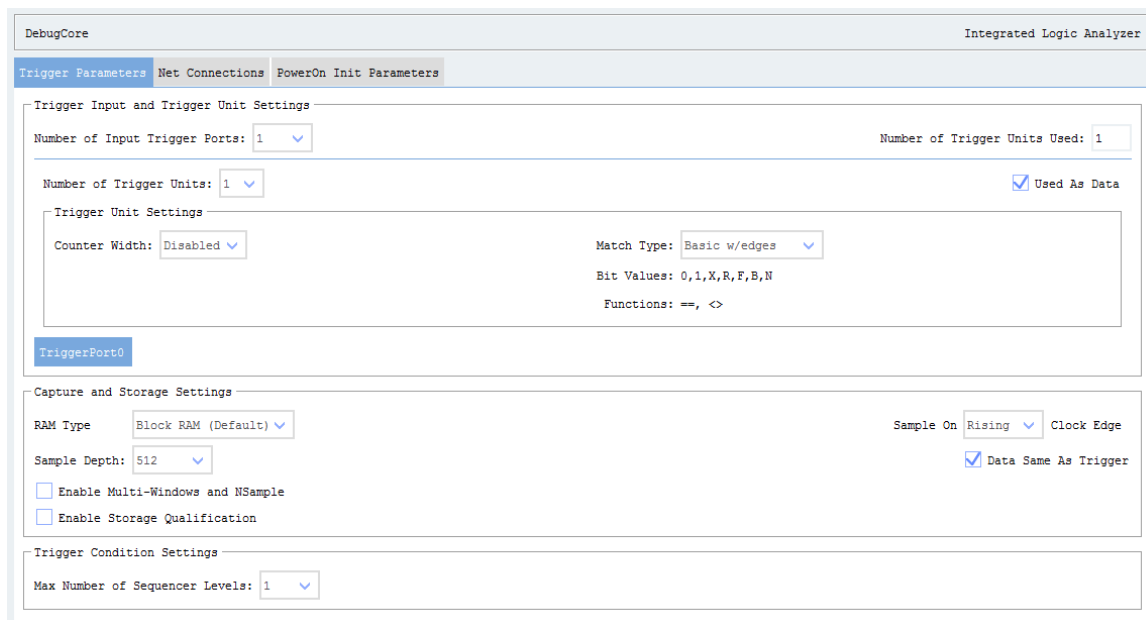


图 2.4 Trigger Parameters 配置界面

2) Net Connections 选项卡

Net Connections 选项卡用于将 DebugCore 的输入信号和用户 design 中的网线连接起来，尚未连接时，Net Connections 中的所有信号都为红色显示如

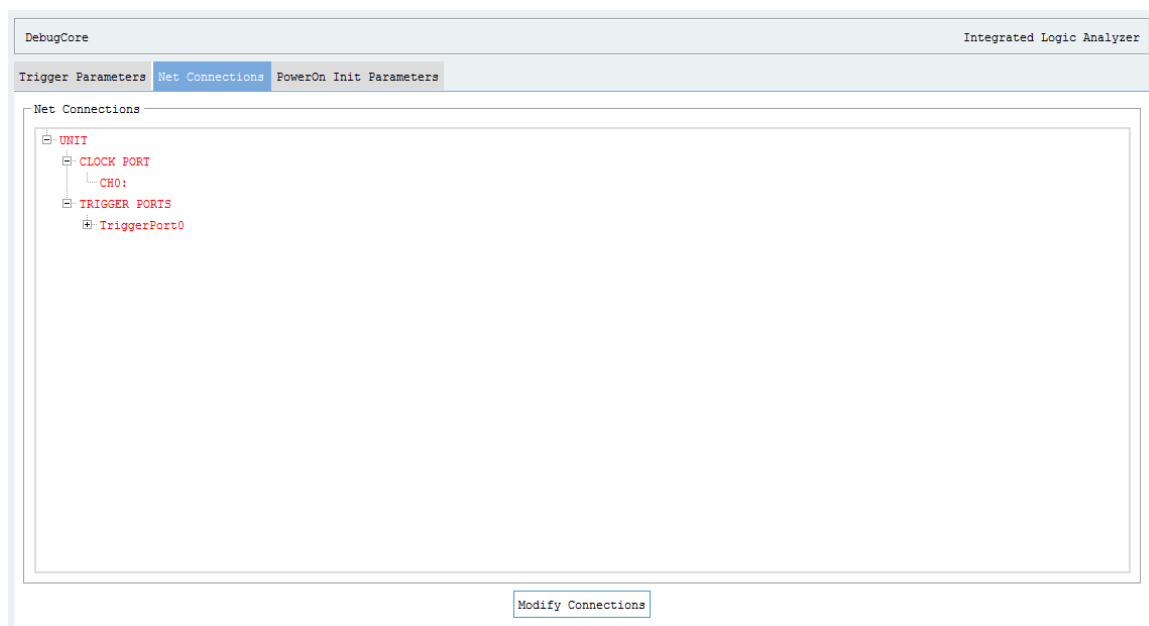


图 2.5。点击 Modify Connection，会弹出 Select Net 配置界面，如

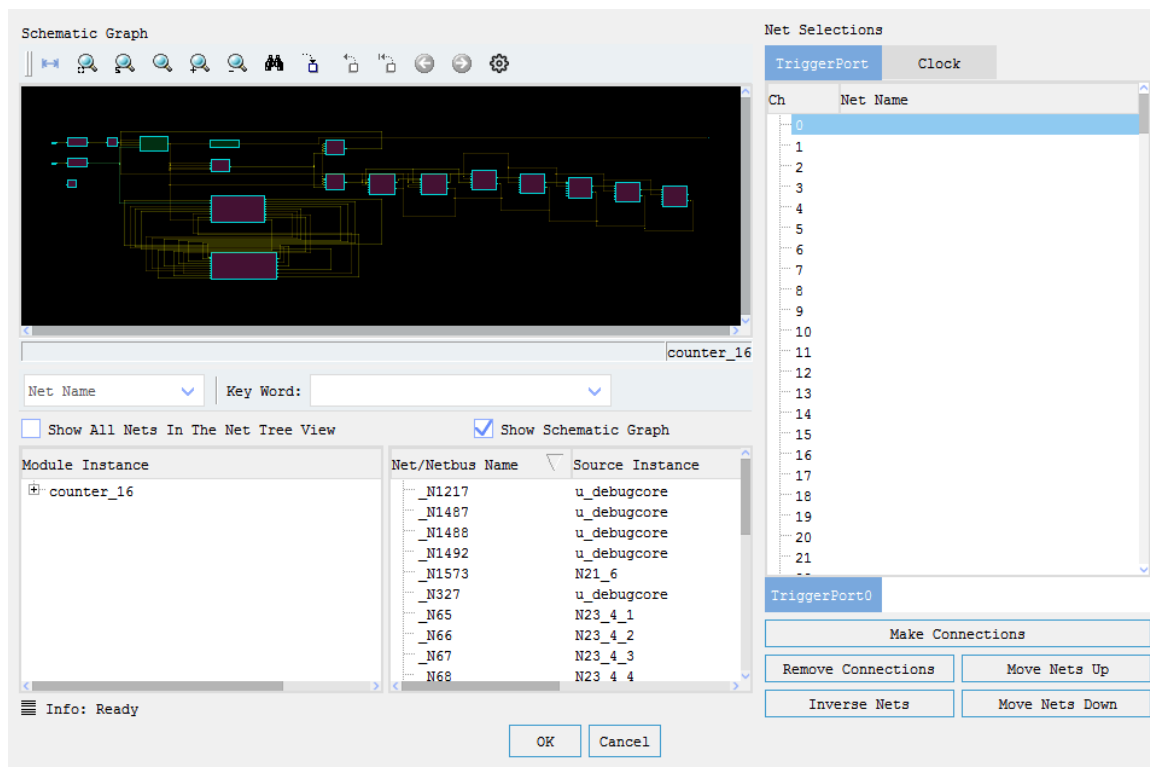


图 2.6。利用 Select Net 对话框，可以把 Debug Core 的工作时钟、触发信号、数据信号与设计中的网线连接起来，方法是右侧选中 Debug Core 的信号，在左侧下方选中需要连接的网线，单击 Make Connections 按钮，即可以完成一条或者多条信号的连接，所有信号都连接好之后，单击 OK 按钮。

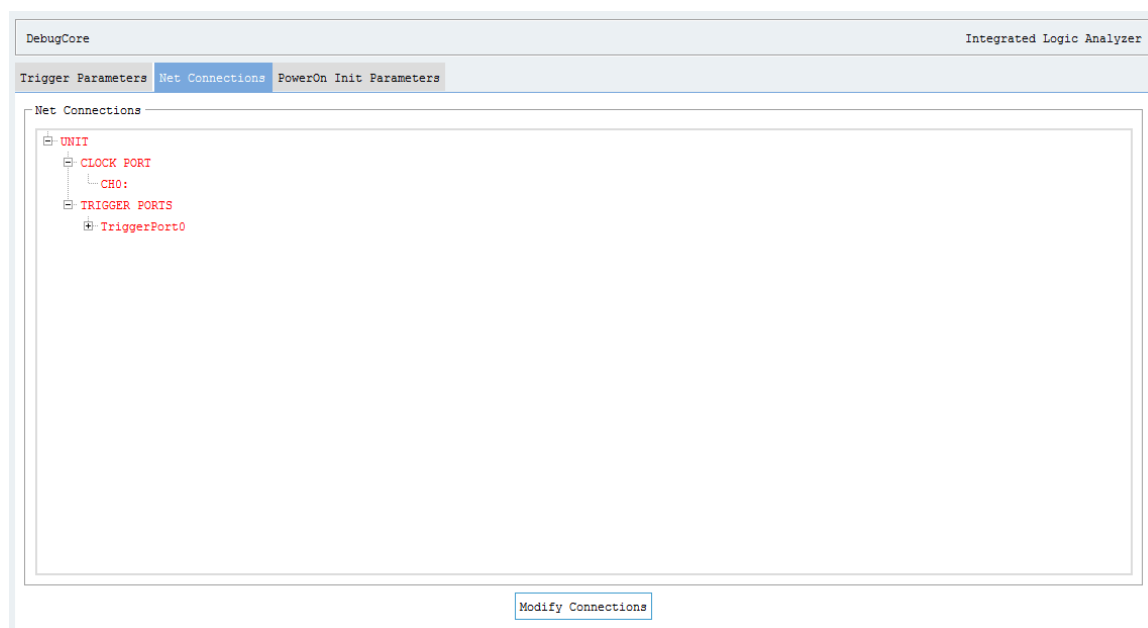
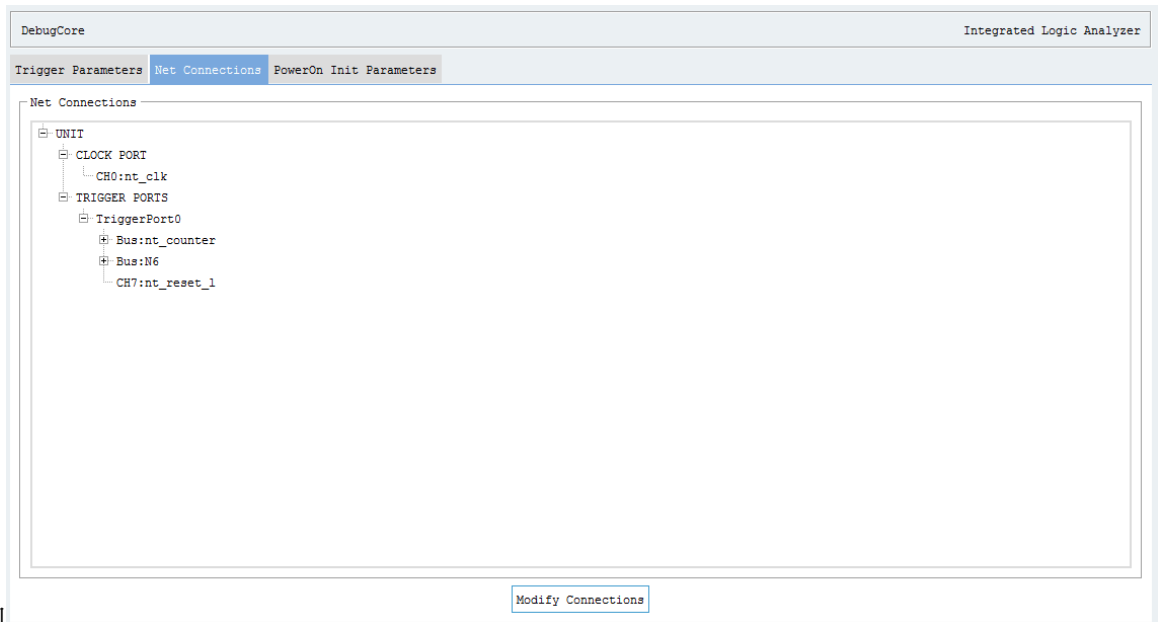


图 2.5 Net Connections 配置界面（连线完成前）

每个 DebugCore 都必须进行信号连接，其中 Trigger Port 和 Clk Port 是必须连接的信号，如果启用了（Trigger Parameters 选项卡 Capture and Storage Settings 中）Data Same As Trigger 选项则无需连接 Data Port，否则也必须对 Data Port 进行信号连接。Reset Port 为可选项（可在 Preference Edit 窗口中启用）。本例中启用 Data Same As Trigger 并且不使用 Reset Port，所以只需对 Clk Port 和 Trigger Port



进行连接，如

图 2.7。

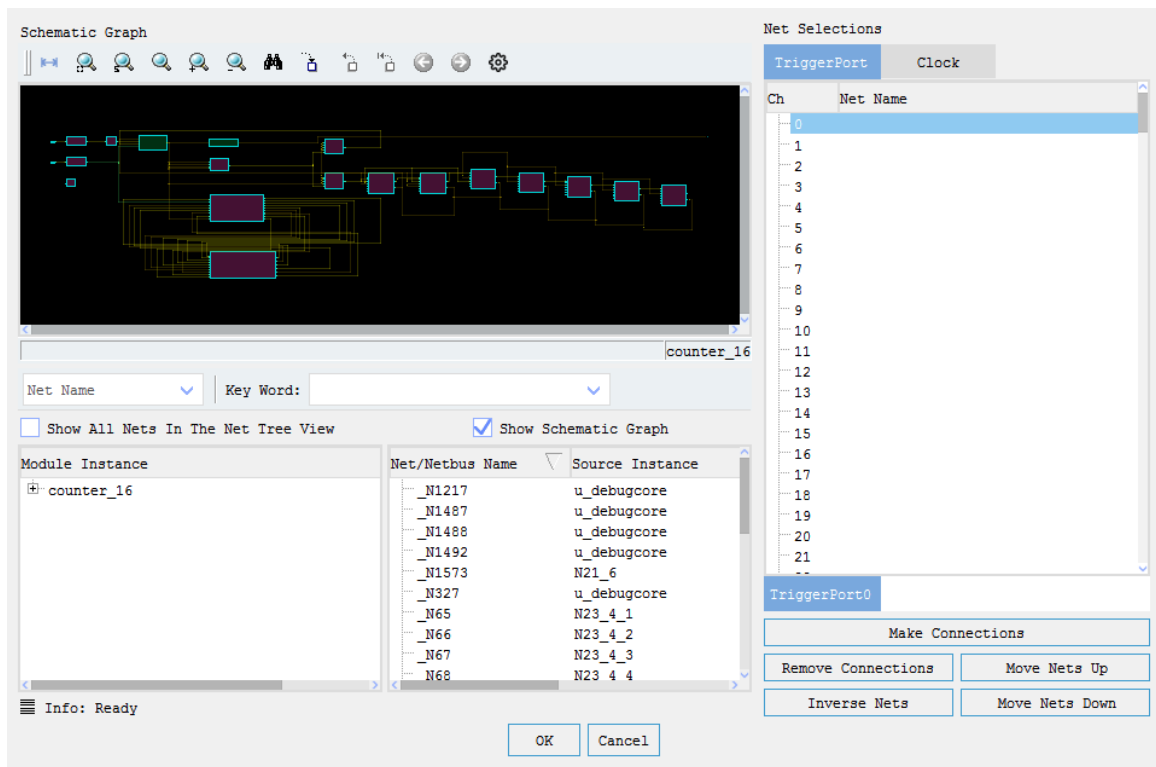


图 2.6 Select Net 配置界面

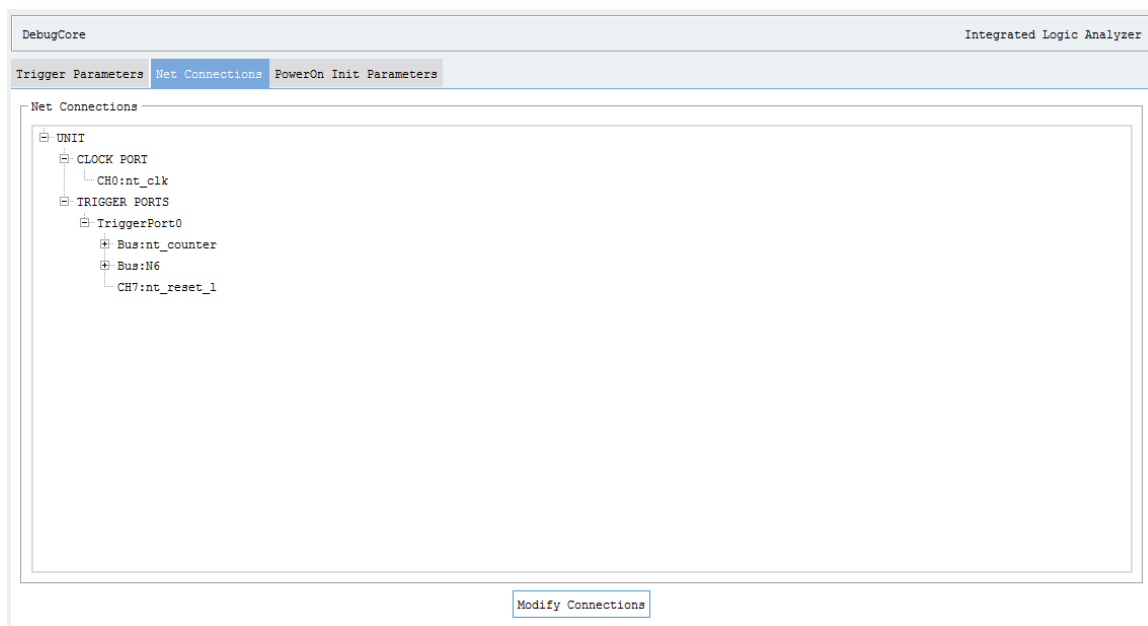


图 2.7 Net Connections 配置界面（连线完成后）

在 Net Connections 配置界面里，用户也可以进行 Trigger Port 排序和删除操作。如

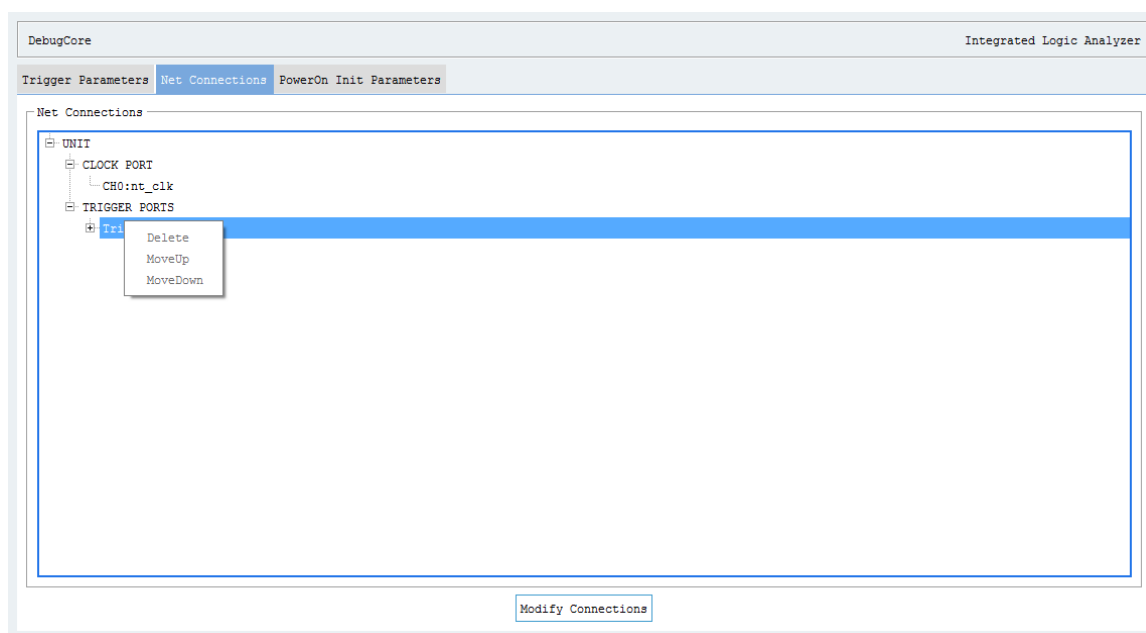


图 2.8 所示，用户在相应的 Trigger Port 上单击鼠标右键将会弹出一个菜单，用户可根据菜单中提供的操作对所选中的 Trigger Port 进行上移、下移或删除。当仅有一个 Trigger Port 时，将不能进行排序和删除操作。

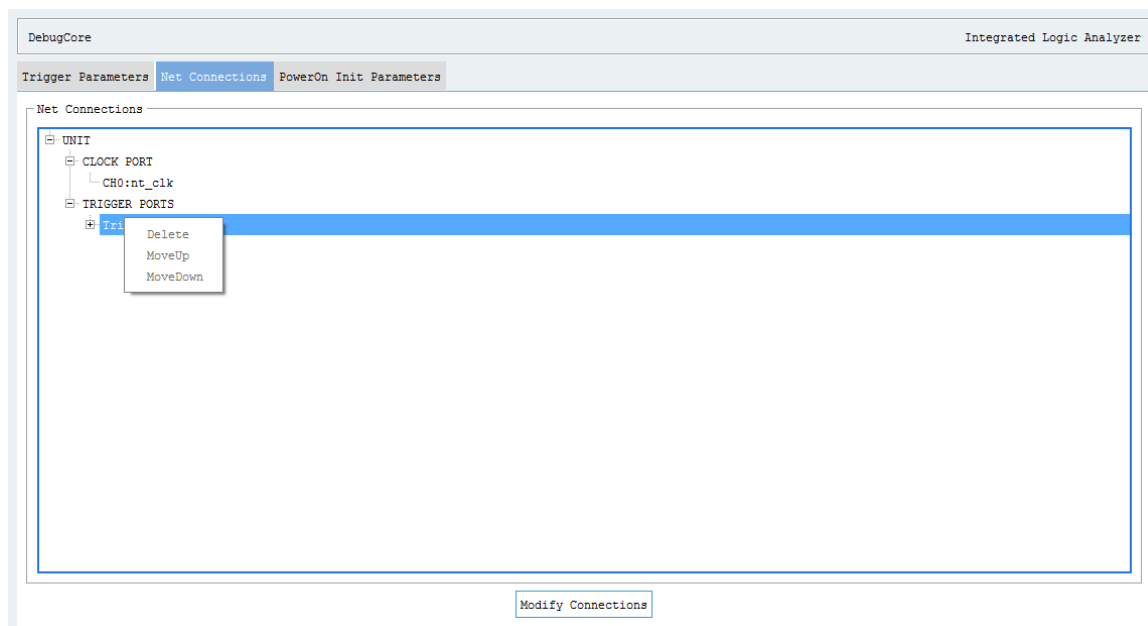


图 2.8 Trigger Port 调整界面

3) PowerOn Init Parameters 选项卡

PowerOn Init Parameters 选项卡用于设置在上电初始化瞬间信号的捕获条件，了解上电初始化瞬间信号的形态和稳定性。

默认情况下，不使能捕获上电初始化数据。如果要使能该功能，请点击“Enable Capturing Power On Initial Data”前的复选框，详见

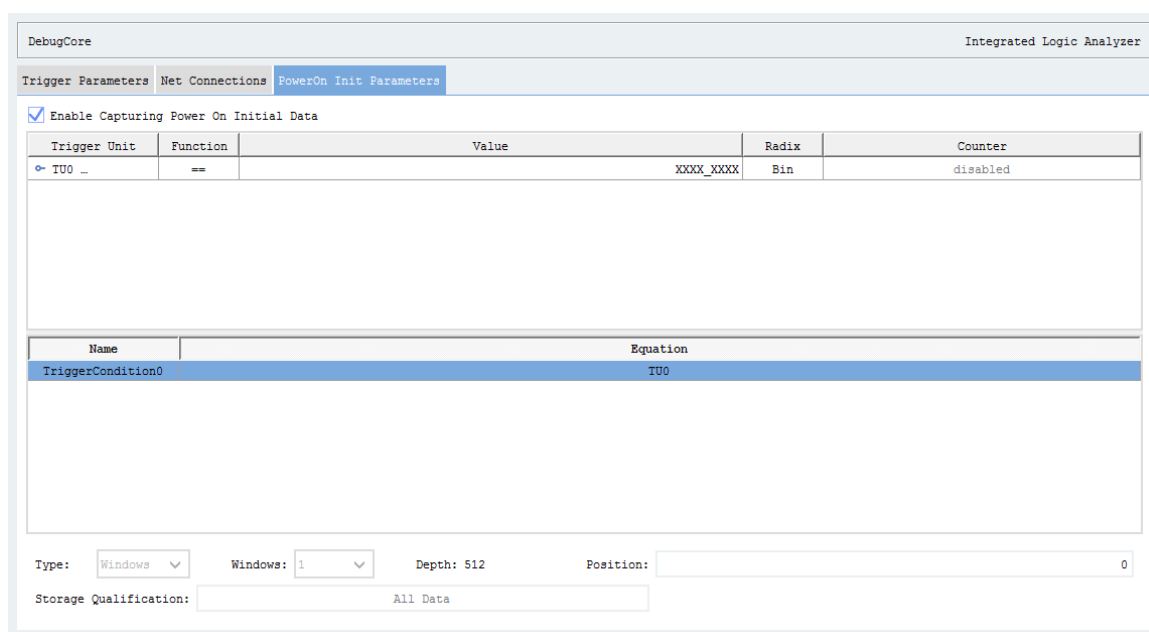


图 2.9。选中后可以点击界面中表格、下拉框等进行详细参数设置，具体配置项意义详见界面功能介绍。

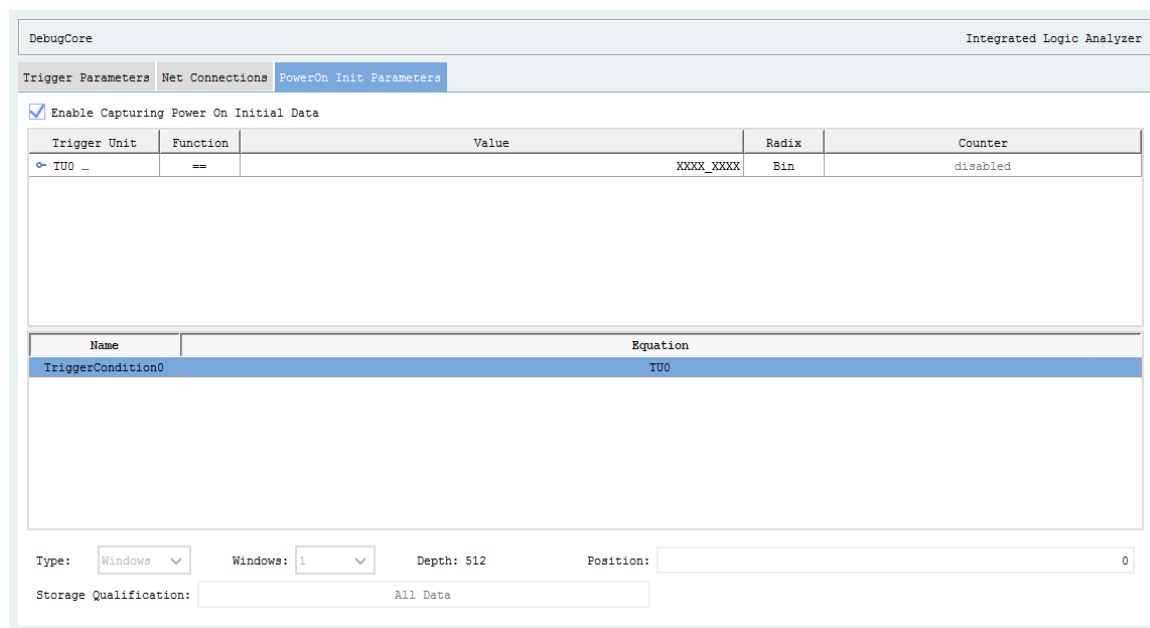


图 2.9 Power On Init Parameters 配置界面

2.3.6 保存 FIC 文件并插核

Debugger Core 各项参数配置完成并完成 Net 连接工作后，点击保存后退出 Inserter。返回 PDS 软件中运行 Flow 中的 Map，PDS 将自动调用插核流程。

2.3.7 另存 FIC 文件

如果需要将当前编辑的 FIC 文件另存至其他位置，可以通过 File 菜单中的 Save Project As 功能进行保存。

2.3.8 Fabric Debugger 中导入 FIC 文件

在使用 Fabric Debugger 进行调试时，可以通过导入位流文件对应的 fic 文件（即生成插核网表文件时使用的工程文件）的方法更新 Trigger Port 及 Data Port 的名称，并自动生成 NetBus。

3 Fabric Inserter 功能介绍

3.1 用户界面介绍

启动 Fabric Inserter 软件，出现如图 3.1：

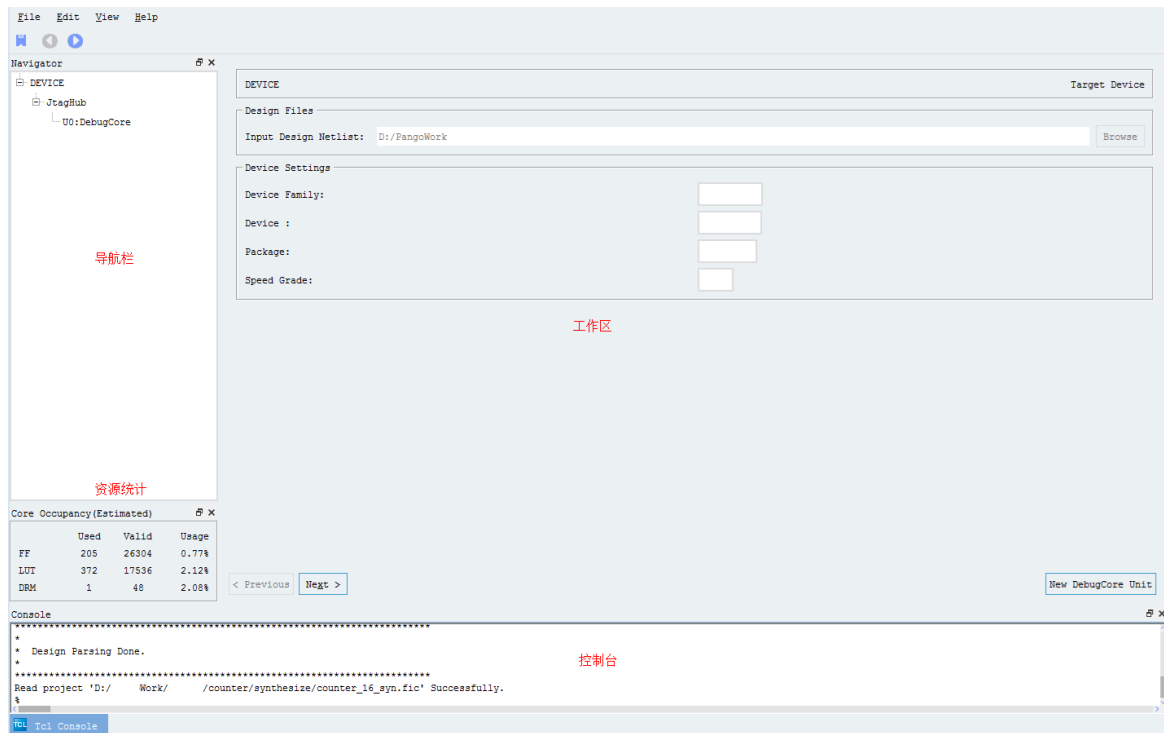


图 3.1 软件主界面

【导航栏】：列出了插入到设计中的所有 JtagHub 和 DebugCore 核，选中一个核后，就可以在参数设置中查看和修改其参数。

【资源统计】：报告逻辑资源的使用量及使用率。

【控制台】：软件工作状态信息都在该窗口显示。

【工作区】：软件参数查看和参数修改设置。

3.2 菜单基本操作说明

3.2.1 File 菜单

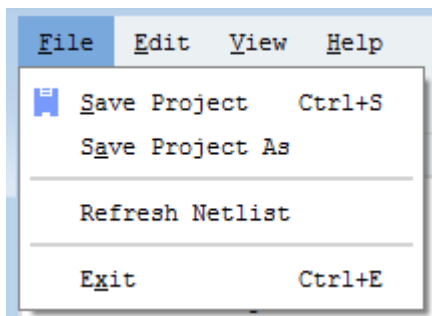


图 3.2 File 菜单

【Save Project】: 用于保存当前工程。

【Save Project As】: 当前工程另存为。

【Refresh Netlist】: 用于重新解析网表。

【Exit】: 用于退出 Inserter 软件。

3.2.2 Edit 菜单

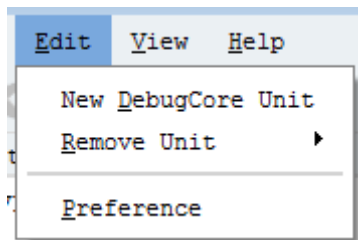


图 3.3 Edit 菜单

【New DebugCore Unit】: 新建 DebugCore Unit 单元。

【Remove Unit】: 删除指定 DebugCore Unit 单元。

【Preference】: DebugCore Unit 参数配置对话框。

3.2.3 View 菜单

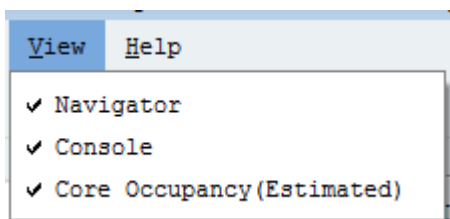


图 3.4 View 菜单

【Navigator】: 用于显示或关闭 Navigator（导航栏）窗口。

【Console】: 用于显示或关闭 Console（控制台）窗口。

【Core Occupancy(Estimated)】: 用于显示或关闭 Core Occupancy(Estimated)（资源统计）窗口。

3.2.4 Help 菜单

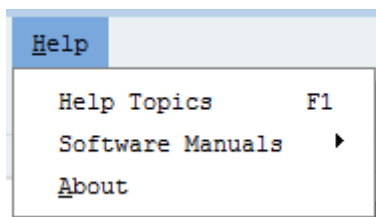


图 3.5 Help 菜单

【Help Topics】: 打开 “help


【Software Manual 是】: 打开 Inserter 用户手册（PDF）

【About】: 用于显示关于该软件的相关信息，如版本信息。


3.3 工具栏说明



图 3.6 工具栏

【Save Project 】: 对应菜单栏 File ->Save Project。

【Previous Screen 】: 用于显示当前视图的前一视图。

【Next Screen 】: 用于显示当前视图的下一视图。

3.4 界面功能介绍

3.4.1 主界面功能介绍

主界面中显示输入及输出文件路径，器件系列及型号，由于上述参数均由 PDS 软件传递，故无法修改。

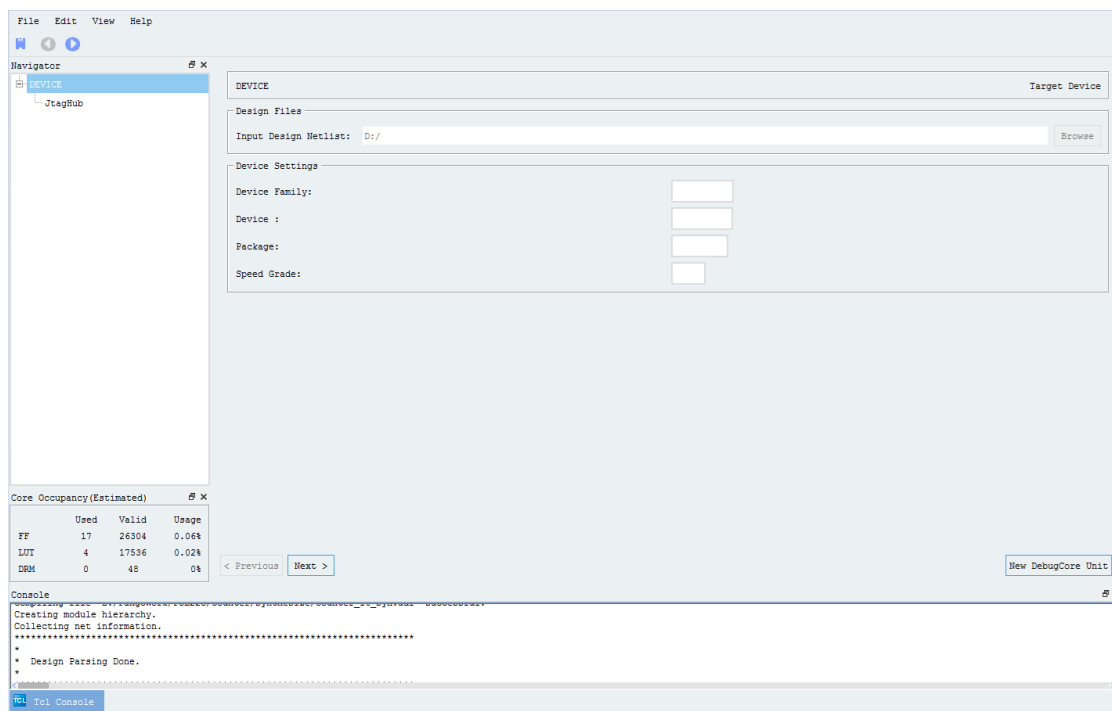


图 3.7 软件主界面

3.4.2 JtagHub 配置界面

在这个界面中用户可做的选择为点击右下角 New DebugCore Unit 按钮添加 DebugCore，最多可添加 15 个。如不添加 DebugCore，再点击 Next 软件会默认添加一个 DebugCore。同时，在该界面中通过 Boundary Scan Chain 的下拉列表可以指定使用的 Scan Chain。通过 Select Jtag 下拉列表可以指定使用的 JTAG，如果需要通过 GTP_JTAGIF 使用 DebugCore，请选择“USER JTAG”，否则请使用“Default JTAG”。

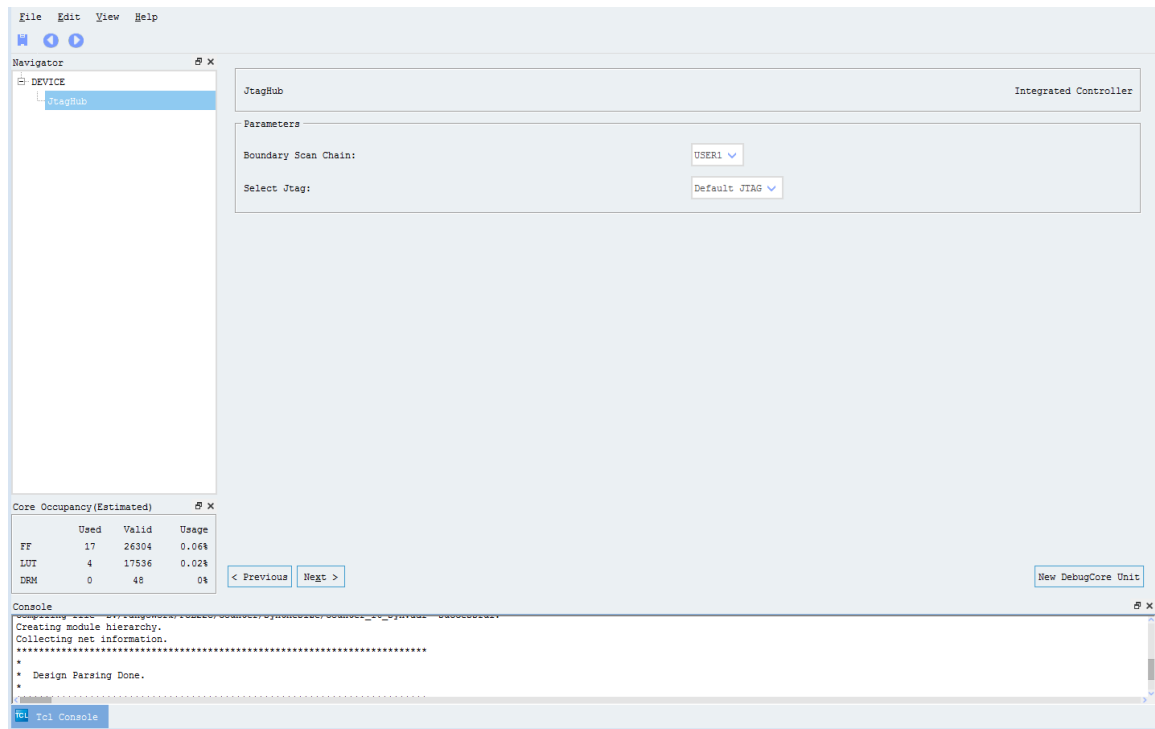


图 3.8 JtagHub 配置界面

3.4.3 DebugCore 配置界面

在这个界面里可以对 DebugCore 参数进行配置。一个 DebugCore 对应的界面中包含 Trigger Parameters、Net Connections 和 PowerOn Init Parameters 为标题的标签页面，分别对应 DebugCore 不同参数的配置。

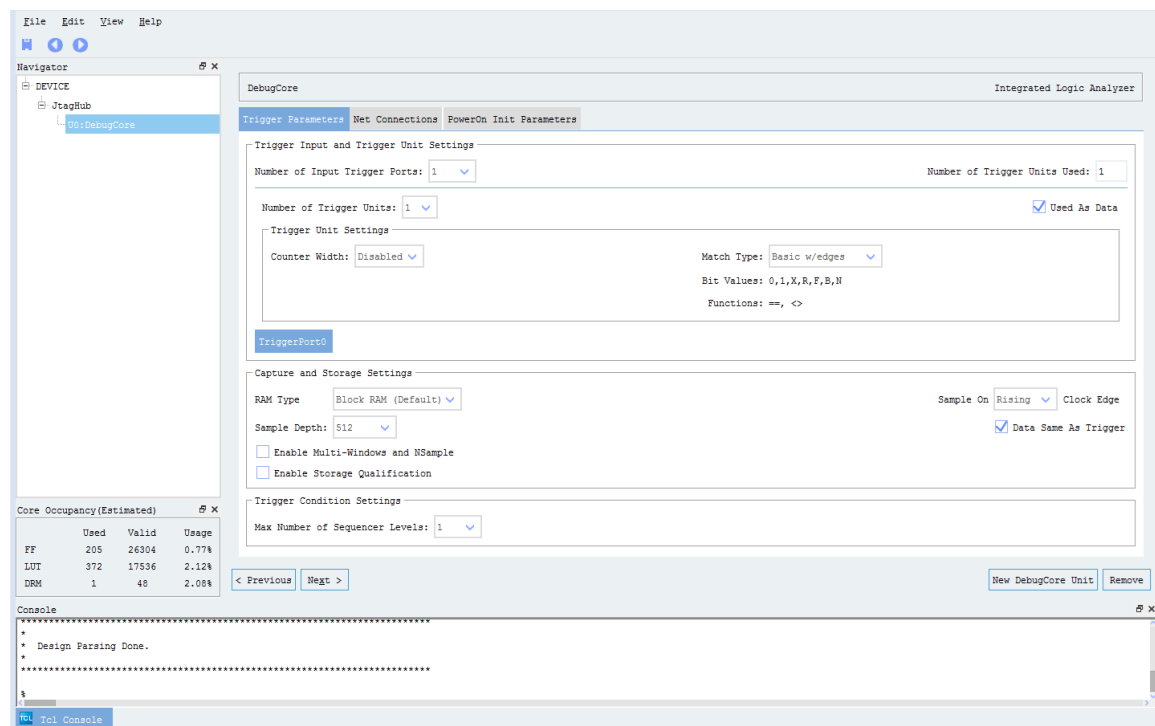


图 3.9 DebugCore 配置界面

1. Trigger Parameters 配置界面

Trigger Parameters 选项卡中的配置选项分为三组，分别用于对触发端口(Trigger Port)，捕获存储(Capture & Storage)和触发条件(Trigger Condition)进行相关参数配置。触发端口相关设置包括触发端口的数量，每个触发端口下挂接的匹配单元的数量及其类型等。捕获存储相关配置包括数据的采样深度，采样时钟沿等。触发条件相关配置主要是对顺序触发的最大等级进行配置。

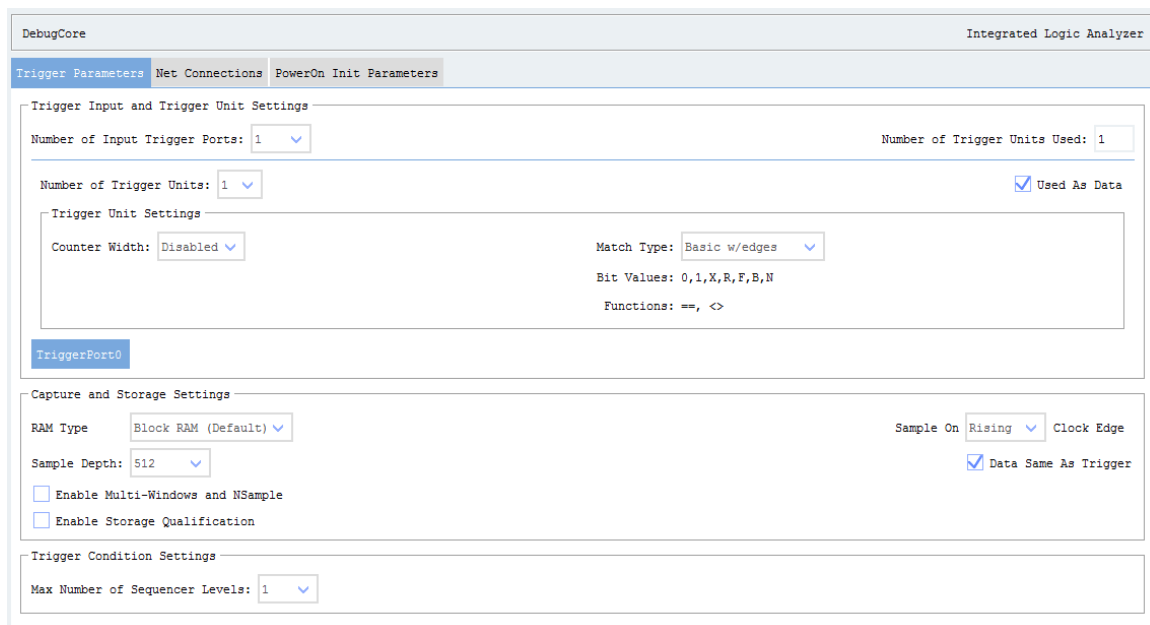


图 3.10 Trigger Parameters 配置界面

1) 触发端口及匹配单元设置 (Trigger Input and Trigger Unit Settings)

【Number of Input Trigger Ports】: 下拉列表，DebugCore 输入触发端口数目，最大值为 16。

【Number Of Trigger Units】: 挂接在当前触发端口下的匹配单元的数量，这些匹配单元共享这个触发端口的 channel。

【Used As Data】: 复选框，设置当前触发端口的输入信号是否同时用作数据捕获端口的输入信号（仅在使能 Data Same As Trigger 时有效）。

【Match Type】: 下拉列表，可选项有 Basic 、Basic w/edges、Extended 、Extended w/edges、Range 、Range w/edges，该选项与 Bit Values 和 Functions 关联。

【Counter Width】: 匹配单元计数器的宽度，可用于对匹配单元的匹配条件被满足的次数进行计数。

【Number of Trigger Units Used】: 当前 Debug Core 中匹配单元总数，即所有触发端口下挂接的匹配单元数量的总和，该数值不能大于 16。对 Number of Input Trigger Ports 和 Number of Trigger Units 下拉列表进行设置使该参数超过 16，软件会报警，并将自动将参数修改为合适的值。

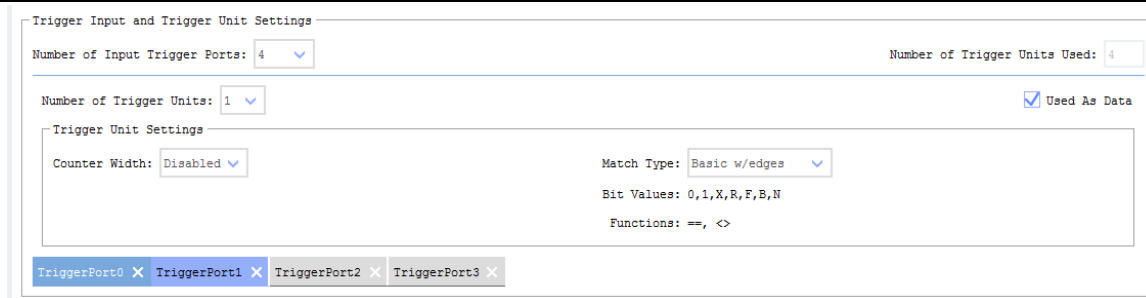


图 3.11 Trigger Input and Trigger Unit Settings

2) 数据捕获及存储设置(Capture and Storage Settings)

【RAM Type】: 设置 DebugCore 存储资源的实现方式，用户可通过选择“Block RAM (Default)”或“Distribute RAM”选项来指定 DebugCore 存储资源的实现方式。

【Sample Depth】: 设置采样深度，范围 64~131072。

【Sample On】: 设置采样时钟沿，Rising 为时钟上升沿，Falling 为时钟下降沿。

【Enable Multi-windows and NSample】: 复选框，设置是否使能多窗口触发及 Nsample 功能，默认为不使能。使能该功能会对 Debug Core 的时序性能造成一定的影响，如果需要 Debug Core 工作在较高频率时，不建议使能该功能。

【Enable Storage Qualification】: 复选框，设置是否使能存储条件，使能该功能后，可以设置一定的条件对待捕获的数据进行过滤，只存储符合条件的数据并在 Debugger 中展示。

【Data Same As Trigger】: 复选框，设置是否将连接至触发端口的信号同时连接至数据捕获端口，如果使能该项，则所有连接至勾选了 Used As Data 项的触发端口的信号也会被连接至数据捕获端口，并且使能该项后，用户无法再对数据捕获端口进行额外的连接。若所有触发端口的 Used As Data 复选框取消勾选，则该复选框自动取消勾选；若该复选框勾选，软件会自动确保至少有一个触发端口的 Used As Data 复选框勾选。

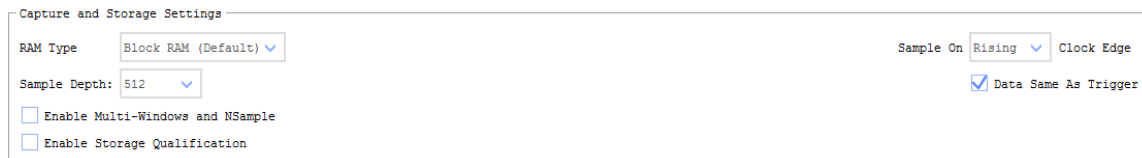


图 3.12 Capture and Storage Settings

3) 触发条件设置(Trigger Condition Settings)

【Max Number of Sequence Levels】: 顺序触发最大级数设置,下拉列表用来选择最大触发条件的级数 n，最大不超过 16 级。



图 3.13 Trigger Condition Setting

2. Net Connections 配置界面

Net Connections 选项卡用于将 DebugCore 的输入信号和用户 design 中的网线连接起来，尚未连接时，Net Connections 中的所有信号都为红色显示如

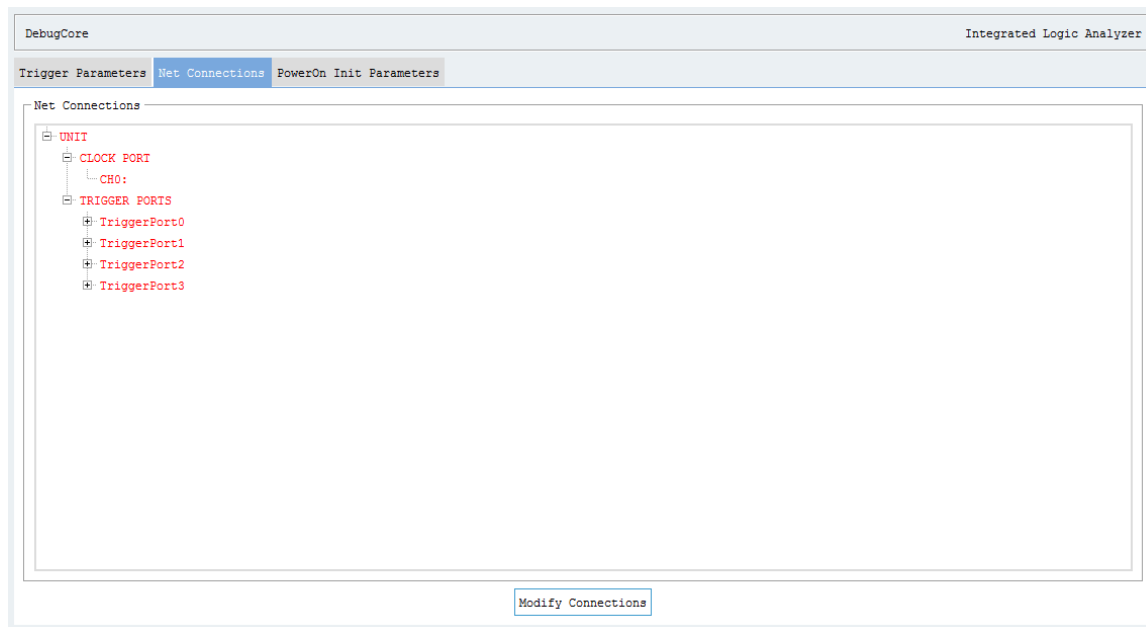


图 3.14。

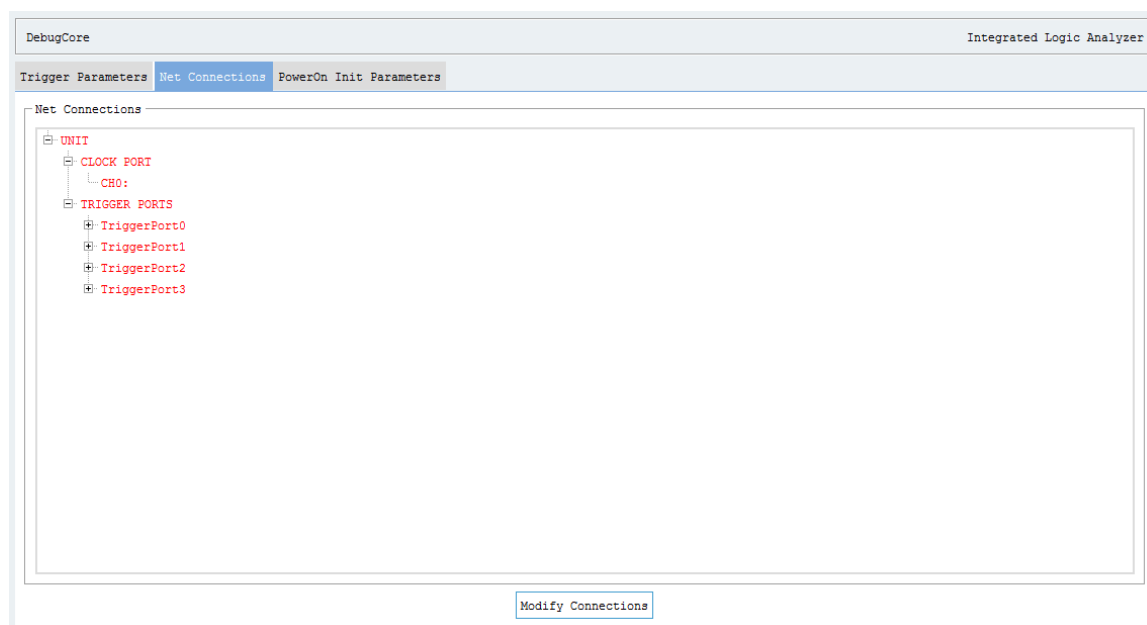


图 3.14 Net Connections 配置界面（连线完成前）

点击 Modify Connection，会弹出 Select Net 配置界面，如图 3.15。

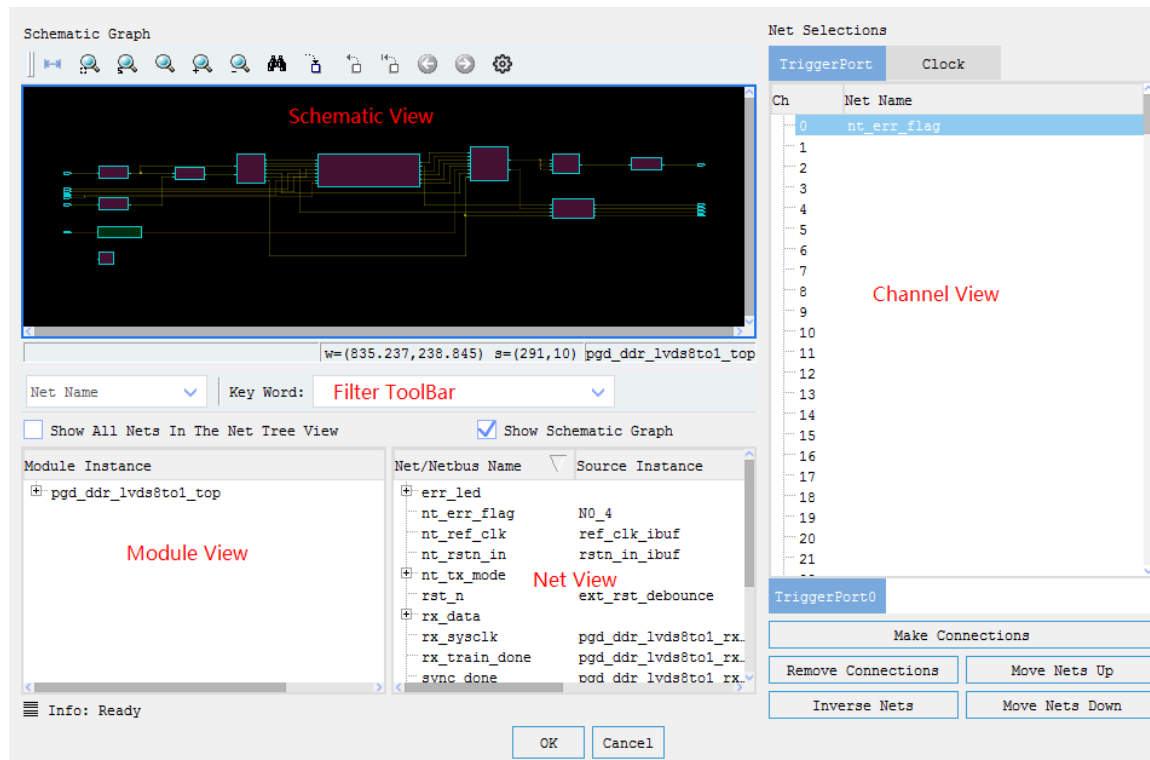


图 3.15 Select Net 配置界面

1) Schematic View

Select Net 对话框左上的窗口称为 Schematic View，该窗口将用户设计文件中所有的模块以及信号连接关系以图形和线段的方式呈现。Schematic View 支持基本的图像放缩功能以及信号连接功能。双击 Schematic View 中的某根信号会将目标信号添加到 debug core 当前 port 的 Channel View 上，与用户设计端口相连的信号无法与 debug core 相连进行观测。

2) Channel View

Select Net 对话框右侧的窗口称为 Channel View，该窗口中显示的是当前 Debug Core 中所有可以用于连接的 channel。该窗口中有多个页签，每个页签代表一个端口（Port），每个端口可以容纳的最大 channel 数不同，例如：触发端口（Trigger Port）有 256 个 channel，即最多可以连接 256 个网线（Net），数据捕获端口（Data Port，使能 Data Same As Trigger 时将不会在该窗口中显示）有 4096 个 channel，即最多可以连接 4096 个网线，而时钟端口（Clock Port）和复位端口（Reset Port）仅有一个 channel，且必须连接至对应的时钟信号和复位信号上，如果时钟端口和复位端口连接不正确将导致 debug Core 无法正常工作。此外，触发端口与数据捕获端口至少连接 1 个网线。Channel View 窗口还支持通过拖动进行信号的排序，只需选中目标信号拖动到相应位置即可。

3) Module View

Select Net 对话框下方左侧的窗口称为 **Module View**，该窗口将用户设计文件中的所有模块按照一定的层次结构进行显示，单击该窗口中的项，可以将对应模块中的所有网线显示在左下方的 **Net View** 中。此外，用户可以使用鼠标左键或右键拖动该窗口中的项，并放置在 **Channel View** 的 **channel** 项上，此拖动操作可一次性将该模块中所有的网线全部连接至目标 **Channel** 及后续 **channel** 中（**channel** 数量不足时，没有 **channel** 可以连接的网线将被忽略）。

4) Net View

Select Net 对话框下方中间的窗口称为 **Net View**，该窗口中主要显示当前模块（**Module View** 中选中的模块）中可以用于连接的总线（**Bus**）及网线。用户可以直接通过选取总线将总线中所有的网线一次性连接，也可展开总线，选取其中个别网线进行逐个连接。连接至 **Channel View** 的总线依然会以总线的形式显示，并且在 **Debugger** 中导入带有总线信息的.fic 文件时，**Debugger** 会自动生成相应的总线。此外，**Net View** 中按住单击左键+shift 可以进行连续多选信号，单击左键+Ctrl 可以进行点选信号，选中项目后可以使用鼠标左键拖动并在 **Channel View** 中的 **channel** 项上放下，此拖放动作可以将选中的总线或网线连接至当前及后续 **channel** 中。

5) Filter ToolBar

Select Net 对话框中左侧中间位置的工具栏称为 **Filter ToolBar**，主要用于对 **Module View** 和 **Net View** 中的项进行筛选。左侧下拉框中可对筛选模式进行选择，主要分为以下 7 种模式：

【Module Instance】: 用于对 **Module View** 中的 **module** 进行筛选，筛选时只关注 **Module Instance** 列（第一列）中的信息是否符合条件。

【Module Name】: 用于对 **Module View** 中的 **module** 进行筛选，筛选时只关注 **Module Name** 列（第二列）中的信息是否符合条件。

【Net - All】: 用于对 **Net View** 中的网线及总线进行筛选，筛选时关注全部 4 列信息，只需网线中有一种信息符合条件，那么当前网线即为符合条件的网线。

【Net Name】: 用于对 **Net View** 中的网线及总线进行筛选，筛选时只关注 **Net/Netbus Name** 列（第一列）中的信息是否符合条件。

【Source Instance】: 用于对 **Net View** 中的网线及总线进行筛选，筛选时只关注 **Source Instance** 列（第二列）中的信息是否符合条件。

【Source Component】: 用于对 **Net View** 中的网线及总线进行筛选，筛选时只关注 **Source**

Component 列（第三列）中的信息是否符合条件。

【Base Type】: 用于对 Net View 中的网线及总线进行筛选，筛选时只关注 Base Type 列（第四列）中的信息是否符合条件。

【Key Word】: Filter ToolBar 中间为 Key Word 编辑框，用于输入筛选关键字，输入完成后输入 Enter 键可以开始筛选，如果勾选了“Show Schematic Graph”选项，则筛选时 Schematic View 会同时跳转到当前筛选框中出现的第一个信号对应处。筛选时将使用筛选关键字对目标信息（通过上述 7 种模式进行设置）进行匹配，匹配方式为模糊匹配且不区分大小写，即只需目标信息中含有筛选关键字就认为该项为符合条件的项。

筛选结束后对于 Module View 和 Net View 中筛选结果的显示方式有所不同：

【Module View 筛选结果显示方式】: 使用 Filter ToolBar 对 module 进行筛选时，该工具栏最右侧会出现两个额外的按钮[Previous]和[Next]，对 module 的筛选完成，窗口焦点会自动跳转至第一个符合条件的项上，然后[Previous]和[Next]按钮将被使能，可以通过点击[Previous]和[Next]按钮移动窗口焦点至前一个或后一个符合条件的项上，焦点移动时，Net View 中显示的网线也会随之变化。

【Net View 筛选结果显示方式】: 使用 Filter ToolBar 对网线进行筛选结束后，Net View 中之后显示符合条件的网线及总线，不符合条件的网线及总线将被隐藏，如果需要再次显示所有网线和总线，只需清空筛选关键字后在进行依次筛选即可。

6) Show All Nets

Select Net 对话框中 Filter ToolBar 左下方复选框 **【Show All Net In The Net Tree View】** 可实现显示所有网线及总线的功能。在不勾选该项时，Net View 中只显示 Module View 中选中的模块中的网线及总线，便于按模块查找网线及总线。勾选该项后，Module View 窗口将被冻结，此时 Net View 窗口中会将当前设计文件中所有的网线及总线全部显示出来，方便查找无法按模块定位的网线及总线。

7) Show Schematic Graph

Select Net 对话框中 Filter ToolBar 右下方复选框 **【Show Schematic Graph】** 可实现展示或隐藏 Schematic View 的效果。在不勾选该项时，Schematic View 将会隐藏。在勾选该项时，Schematic View 则会出现，方便用户以图形界面的方式进行信号连接。默认情况下该复选框为勾选状态

8) Buttons

【Make Connections】: 连接按钮，将 Net View 中选中的网线或总线连接至 Channel View 中选中的 channel 中，快捷键为 Ctrl+M。

【**Remove Connections**】: 移除当前 Channel View 中选中的一个或多个 channel 的连接关系，快捷键为 Ctrl+R。

【**Inverse Nets**】: 将当前 Channel View 中选中的多个 channel 的连接顺序反序并重新连接，快捷键为 Ctrl+I。

【**Move Nets Up**】: 将当前 Channel View 中选中的一个或多个 channel 上移，快捷键为 Ctrl+Up。

【**Move Nets Down**】: 将当前 Channel View 中选中的一个或多个 channel 下移，快捷键为 Ctrl+Down。

【**OK**】: 保存本次连接并退出窗口，快捷键为 Ctrl+S（仅保存当前连接）、ESC（退出窗口）。

【**Cancel**】: 不保存修改直接退出窗口，快捷键 ESC。

9) 拖放与 Make Connections 按钮的不同

Select Net 窗口中有两种方法可以将网线与 channel 进行连接，一是通过选中网线和 channel 然后点击 Make Connections 按钮的方式进行连接，二是通过拖放的方式进行连接。这两种方式在进行连接时有一点不同：当当前 channel 已经存在连接关系时，使用 Make Connections 按钮连接会直接覆盖当前 channel 中的连接关系，而使用拖放连接则会执行插入操作，当前 channel 的连接关系不会被覆盖，而是被移动至后续的 channel 中。

10) 操作 Bus

在 Net Select Dialog 中,用户可以直接在 Net View 中选择或拖动一个总线将其连接至 channel view 中。连接后, channel view 中会保持原有的总线结构。

此外,用户可以通过 channel view 的右键菜单来创建新总线,或拆分现有总线,以及对总线进行改名。

通过在 channel view 中选择多个连续的且已连接的 channel, 点击鼠标右键, 选择弹出菜单中的“Make Bus”项, 即可创建总线。新建的总线自动被命名为“CustomBus”, 用户可选择该总线, 点击鼠标右键, 选择弹出菜单中的“Rename Bus”项, 或鼠标左键双击该总线, 即可对其进行改名。此外, 如果需要拆分一个总线, 可以选择需要拆分的总线, 点击鼠标右键, 选择弹出菜单中的“Split Bus”即可将总线拆分为独立的网线。

注: “Rename Bus”、“Split Bus” 每次只能操作一个 Bus 项; “Make Bus” 时选择的 channel 必须是连续且已连接了 net 的, 同时不能包含总线或总线中的网线。

11) 完成连接

当所有端口都完成了必要的连接后，单击 OK 按钮确认本次连接，单击 Cancel 按钮放弃本次连接。连接完成后的 Net Connections 界面如

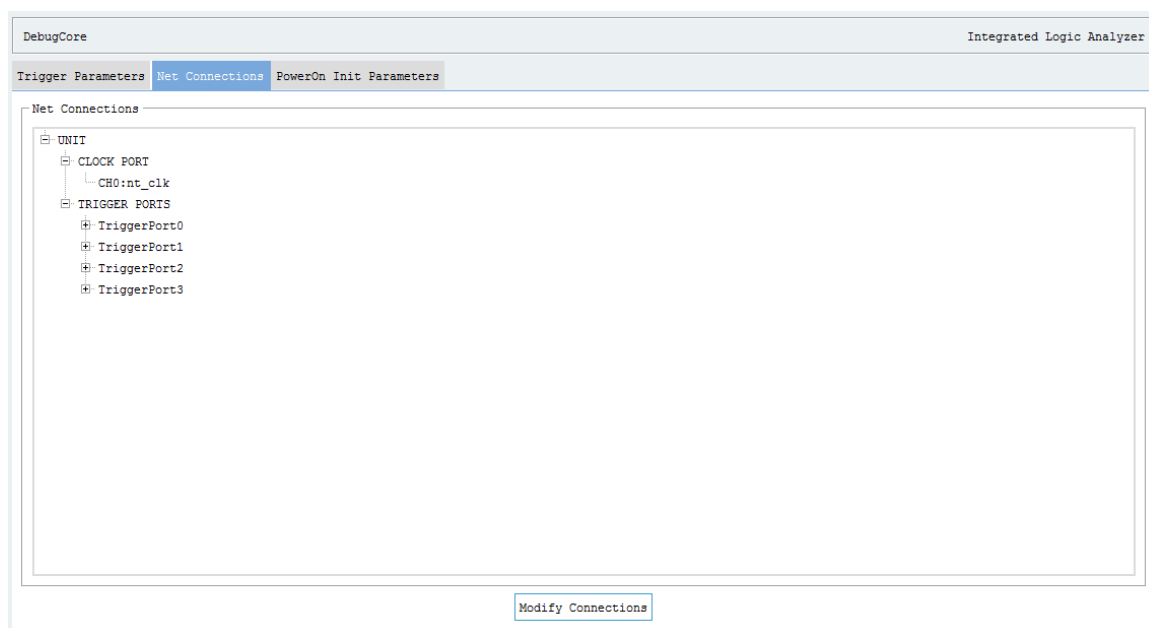


图 3.16 所示。

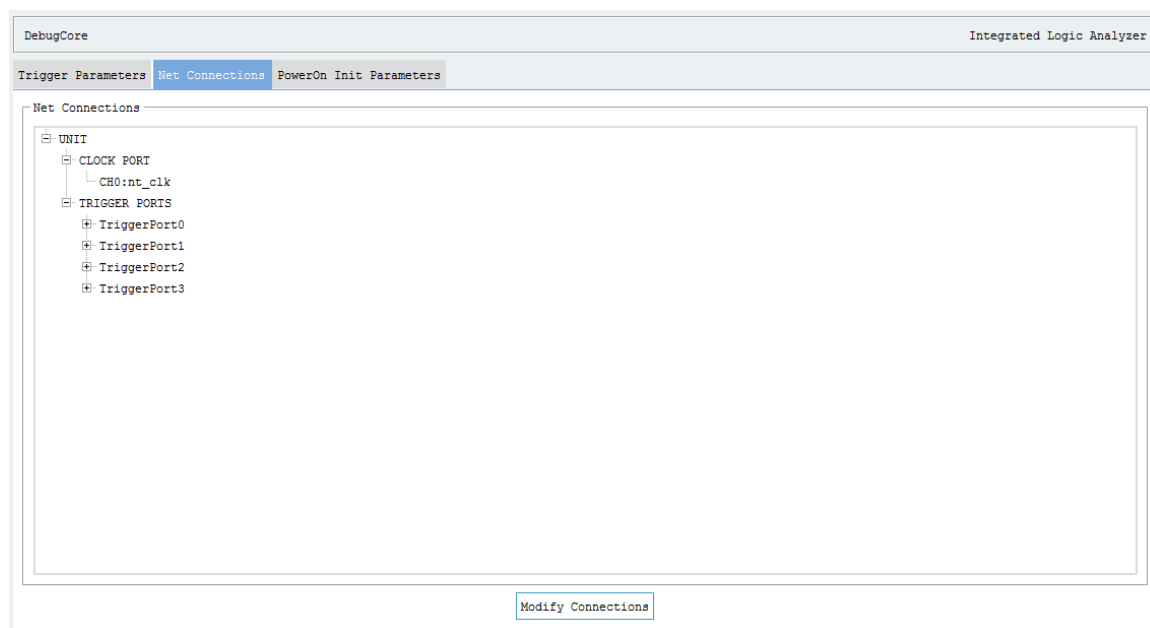


图 3.16 Net Connections 配置界面（连线完成后）

3. PowerOn Init Parameters 配置界面

这个选项卡用于设置在上电初始化瞬间时信号的捕获条件，了解上电初始化瞬间信号的形态和稳定性。默认情况下，不使能捕获上电初始化数据。如果要使能该功能，请点击“Enable Capturing Power On Initial Data”前的复选框。

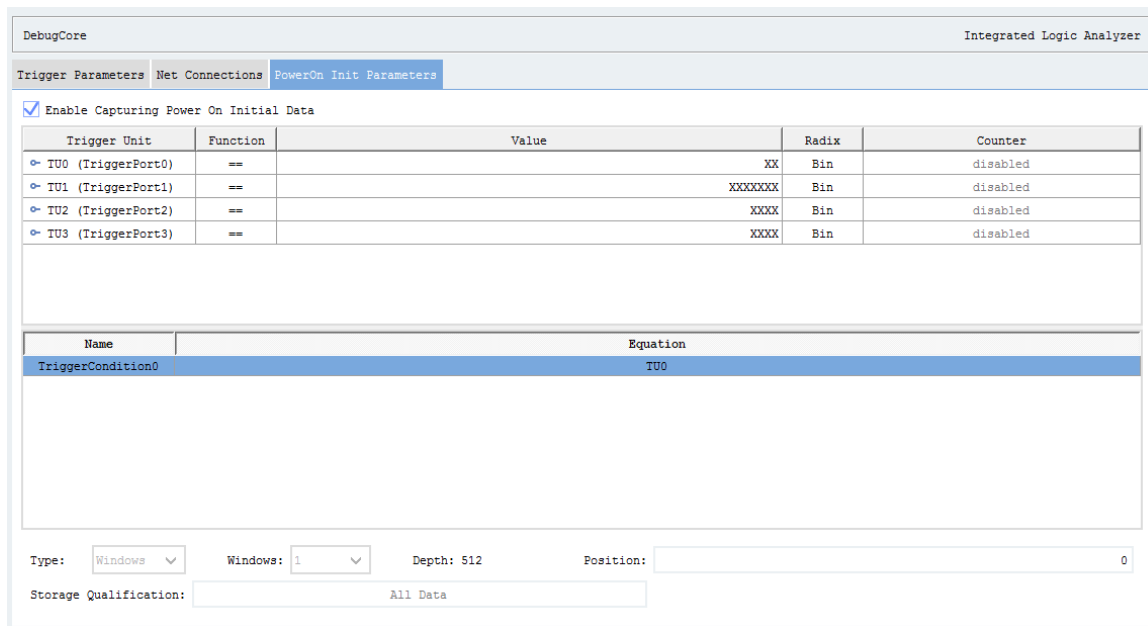


图 3.17 PowerOn Init Parameters 配置界面

1) Match Functions

Match Function 定义了每个 Trigger Unit 在触发条件中的值，如

Trigger Unit	Function	Value	Radix	Counter
TU1 (TriggerPort1)	==	XXXXXXX	Bin	disabled
nt_counter[0]		X		
nt_counter[1]		X		
nt_counter[2]		X		
nt_counter[3]		X		
N6[1]		X		
N6[2]		X		
N6[3]		X		
TU2 (TriggerPort2)	==	XXXX	Bin	disabled

图 3.18。

Trigger Unit	Function	Value	Radix	Counter
TU1 (TriggerPort1)	==	XXXXXXX	Bin	disabled
nt_counter[0]		X		
nt_counter[1]		X		
nt_counter[2]		X		
nt_counter[3]		X		
N6[1]		X		
N6[2]		X		
N6[3]		X		
TU2 (TriggerPort2)	==	XXXX	Bin	disabled

图 3.18 Match Function 配置

【Trigger Unit】: Trigger Unit 列出了当前 DebugCore 中所有的 Trigger Unit，默认名称为 TU0。

点击 Trigger Unit 列中的项，可以展开 Trigger Unit 中的 channel，展开后如

Trigger Unit	Function	Value	Radix	Counter
TU1 (TriggerPort1)	==	XXXXXXX	Bin	disabled
nt_counter[0]		X		
nt_counter[1]		X		
nt_counter[2]		X		
nt_counter[3]		X		
N6[1]		X		
N6[2]		X		
N6[3]		X		
TU2 (TriggerPort2)	==	XXXX	Bin	disabled

图 3.18，用户可以根据信号名称对单一信号的值直接进行编辑。

【Function】: Function 列用来选择 Trigger Unit 的 function。

【Value】: Value 列被单击后，会变为可编辑状态，用来输入 Trigger Unit 的值，值的显示与 Radix 列的选择有关。当输入光标位于待编辑字符前时，该字符可被编辑，当输入了一个可被接受的字符后，该位置原来的字符被替换。不同进制下可接受的字符如下：

【Hex】: X, ?, 0-9, 以及 A-F。X 指其所代表的 4bit 值不被关心，即可以取任意值。当出现“?”，指其所代表的 4bit 值包含 0-9, A-F, X、R、F、B、N 多个值的组合，此符号不可输入。

【Octal】: X、?、0-7；当出现“?”，指其所代表的 3bit 值包含 0-7, X、R、F、B、N 多个值的组合，此符号不可输入。

【Binary】: X（不关心值为多少）、0、1、R（上升沿）、F（下降沿）、B（任意翻转）、N（不翻转），当 Trigger Unit 可以检测沿（Basic w/edges、Extended w/edges、Range w/edges）时 R、F、B、N 可输入。

【Unsigned】: 0-9（0 至 2 的 n 次幂减一，n 为该 Trigger Unit 对应 Trigger Port 中的信号数）

【Radix】: Radix 列被单击后，变为下拉列表，用来选择 Trigger Unit 值以何种进制显示。可以选择 Hex、Octal、Bin 以及 Unsigned。

【Counter】: Counter 用来选择该 Trigger Unit 的 Function 被满足多少次后，该 Trigger Unit 的表达式才算满足。如果该 Trigger Unit 中的 Counter 可用，则该列项为黑色字，如果不可用，则该列项为灰色。Counter 列被单击后，弹出 Match Counter 配置对话框，如图 3.19：

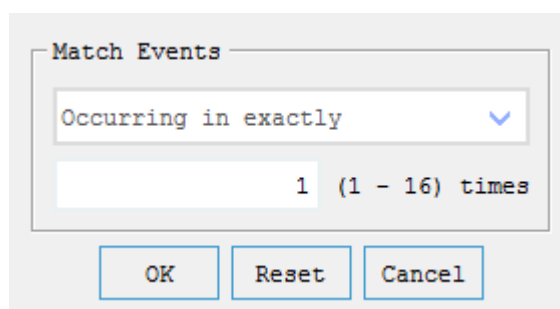


图 3.19 Match Counter 配置

【Match Events】

Match Events 下拉菜单中共有 4 个选项：

【counter disable】: 不使用计数器。

【Occurring in exactly】: 精确匹配 N 次 Trigger Unit 设置的条件（N 为文本框中输入的数值）。

【Occurring in at least】: 至少匹配 N 次 Trigger Unit 设置的条件（N 为文本框中输入的数值）。

【Lasting for at least】: 至少连续匹配 N 次 Trigger Unit 设置的条件（N 为文本框中输入的数值）。

2) Trigger Condition

Trigger Condition 是一个或多个 Trigger Unit 组成的布尔表达式或者序列，用来指导 DebugCore 数据的捕捉。

Name	Equation
TriggerCondition0	TU0

图 3. 20 Trigger Condition

【Trigger Condition Name】: Trigger Condition Name 列用来编辑 Trigger Condition 的名字，默认为“TriggerCondition0”。

【Trigger Condition Equation】: Trigger Condition Equation 用来显示 Trigger Unit 组成的布尔表达式或者序列，该序列可以通过点击列项弹出 Trigger Condition 配置对话框来更改。

Trigger Condition 配置对话框的 Boolean 标签页是一个包含了所有 Trigger Unit 的表，每个 Trigger Unit 占用表的一行。“Enable”列用来选择是否将该 Trigger Unit 作为表达式的一部分；“Negate”列用来决定是否将该 Trigger Unit 取逻辑非。所有使能的 Trigger Unit 可以通过逻辑与、或进行组合，可以通过点击单选按钮“And Equation”、“Or Equation”来选择。“Negate Whole Equation”用来将表达式整体取非。被选择的表达式显示在对话框的最底部，如图 3. 21。

The screenshot shows the 'Sequencer' tab of the 'Trigger Condition' configuration dialog. At the top, there are radio buttons for 'And Equation' (selected) and 'Or Equation', and a checkbox for 'Negate Whole Equation'. Below this is a table with three columns: 'Trigger Unit', 'Enable', and 'Negate'. The table contains four rows for TU0 through TU3. TU0, TU1, and TU2 have their 'Enable' checkboxes checked, while TU3 is unchecked. TU2 also has its 'Negate' checkbox checked. Below the table is a text area labeled 'Trigger Condition Equation' containing the expression 'TU0 && TU1 && !TU2'. At the bottom are 'OK' and 'Cancel' buttons.

Trigger Unit	Enable	Negate
TU0 (TriggerPort0)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TU1 (TriggerPort1)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TU2 (TriggerPort2)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
TU3 (TriggerPort3)	<input type="checkbox"/>	<input type="checkbox"/>

Trigger Condition Equation

TU0 && TU1 && !TU2

OK Cancel

图 3.21 配置 Trigger Condition 布尔表达式

Trigger Condition 配置对话框的 Sequencer 标签页包含了一个“Number of levels”下拉列表用来选择触发条件的级数 n ，该级数是当前 DebugCore 的参数之一，最大不超过 16 级；以及一个包含了 n 行的表，用来选择每级触发的 Trigger Unit，该表的“Trigger Unit”列的每个列项被点击后变成一个下拉列表，列表列出了当前 DebugCore 中的所有 Trigger Unit，可以选择任一个，“Negate”列用来将其对应的 Trigger Unit 取非；被选择的 Trigger Unit 按照 Level 顺序等待触发，顺序触发从 Level1 开始，直到 Level1 被满足后，Level2 才能被等待触发，依次类推。所有的触发可以是连续的，也可以是不连续的，通过“Use Continuous Match Event Only”选择。触发条件表达式显示于对话框底部，如

图 3.22。

The screenshot shows the 'Sequencer' tab in a configuration window. At the top, there are two tabs: 'Boolean' and 'Sequencer'. Below the tabs, there is a 'Number of Levels' dropdown set to '4' and a checkbox for 'Use Continuous Match Event Only' which is unchecked. A table lists four levels with their trigger units and negate checkboxes. Level 3 is selected with a checked negate box. Below the table is a 'Trigger Condition Equation' text area containing the expression 'TU0->TU1->!TU3->TU2'. At the bottom are 'OK' and 'Cancel' buttons.

Level	Trigger Unit	Negate
1	TU0 (TriggerPort0)	<input type="checkbox"/>
2	TU1 (TriggerPort1)	<input type="checkbox"/>
3	TU3 (TriggerPort3)	<input checked="" type="checkbox"/>
4	TU2 (TriggerPort2)	<input type="checkbox"/>

Trigger Condition Equation

TU0->TU1->!TU3->TU2

OK Cancel

图 3.22 配置 Trigger Condition 顺序触发表达式

3) Capture Settings

Capture Settings 主要分为两个部分：Multi-Windows & Nsample 和 Storage Qualification，需要分别使能 Trigger Parameters 页签中的 Enable Multi-Windows and Nsample 和 Enable Storage Qualification 项后才能使用。

Capture Settings 如下图所示，它定义了窗口数、每个窗口的数据深度及触发位置。每个窗口显示一个连续的采样序列，有且仅有一个触发点。如果 Capture Settings 设置有误，则相应区域会变为红色，用以警告用户。

The screenshot shows the 'Capture Settings' window. It has two sections. The top section is for 'Windows' and includes a 'Type' dropdown set to 'Windows', a 'Windows' dropdown set to '1', a 'Depth' field set to '512', and a 'Position' field set to '0'. The bottom section is for 'N Samples' and includes a 'Type' dropdown set to 'N Samples', a 'Samples Per Trigger' field set to '512', and a 'Storage Qualification' dropdown set to 'All Data'.

Type: Windows Windows: 1 Depth: 512 Position: 0

Storage Qualification: All Data

Type: N Samples Samples Per Trigger: 512

Storage Qualification: All Data

图 3.23 Capture Settings

【Type】: Type 是个下拉列表，有两个可选项。如果选择“Windows”，则每个窗口中捕捉的采样点数必须为 2 的幂，触发点可以是窗口中的任何位置；如果选择“N Samples”，则每个窗口的采样点数可以任意制定，采样窗口数由总采样深度和每个窗口的采样深度共同决定，但触发点一定是每个窗

口的第一个采样点。

【Windows】: Windows 是个输入框，仅当 Type 选择“Windows”时可用。Windows 用来定义每次数据捕捉的窗口数目，可以为正整数 N ($1 \leq N \leq$ 当前 DebugCore 的最大采样深度)。

【Depth】: 仅当 Type 选择 “Windows” 时可用，显示当前每个窗口的采样深度。

【Position】: Position 是个输入框，仅当 Type 选择“Windows”时可用。Position 定义了每个窗口触发点的位置，可取整数 N ($0 \leq N \leq \text{Depth} - 1$)。

【Samples Per Trigger】: Samples Per Trigger 是个输入框，仅当 Type 选择“N Samples”时可用。Samples Per Trigger 定义了每个窗口的采样点数，可取整数 N ($1 \leq N \leq$ 当前 DebugCore 的最大采样深度)。

【Storage Qualification Condition】: Storage Qualification Condition 是个文字显示框，点击后弹出“Storage Condition”对话框。Storage Qualification Condition 定义了一个布尔表达式，它由一个或多个 Trigger Unit 进行逻辑与、或、非组成。这个表达式用来指导触发条件满足后数据的存储，即触发后，满足表达式的数据才被存储。

Storage Condition 对话框有一个包含了所有 Trigger Unit 的表，每个 Trigger Unit 占用表的一行。“Enable”列用来选择是否将该 Trigger Unit 作为表达式的一部分；“Negate”列用来决定是否将该 Trigger Unit 取逻辑非。

Storage Qualification Condition 表达式可以选择捕捉所有的数据，如图 3. 24，或者捕捉满足布尔表达式的数据，如图 3. 25，布尔表达式可以通过选择“Negate Whole Equation”整体取非。表达式被显示于对话框的下方。

☒ All Data ☐ And Equation ☐ Or Equation ☐ Negate Whole Equation

Trigger Unit	Enable	Negate
TU0 (TriggerPort0)	<input type="checkbox"/>	<input type="checkbox"/>
TU1 (TriggerPort1)	<input type="checkbox"/>	<input type="checkbox"/>
TU2 (TriggerPort2)	<input type="checkbox"/>	<input type="checkbox"/>
TU3 (TriggerPort3)	<input type="checkbox"/>	<input type="checkbox"/>

Storage Condition Equation

All Data

OK Cancel

图 3. 24 Storage Qualification Condition 选择 All Data

☐ All Data
 ☒ And Equation
 ☐ Or Equation
 ☐ Negate Whole Equation

Trigger Unit	Enable	Negate
TU0 (TriggerPort0)	<input type="checkbox"/>	<input type="checkbox"/>
TU1 (TriggerPort1)	<input checked="" type="checkbox"/>	<input type="checkbox"/>
TU2 (TriggerPort2)	<input type="checkbox"/>	<input type="checkbox"/>
TU3 (TriggerPort3)	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Storage Condition Equation

TU1 && TU3

图 3. 25 Storage Qualification Condition 选择布尔表达式

3.4.4 Preferences Dialog 功能介绍

通过 Edit 菜单中的 Preference 选项可以打开 Preferences Dialog 窗口，窗口如图 3. 26 所示：

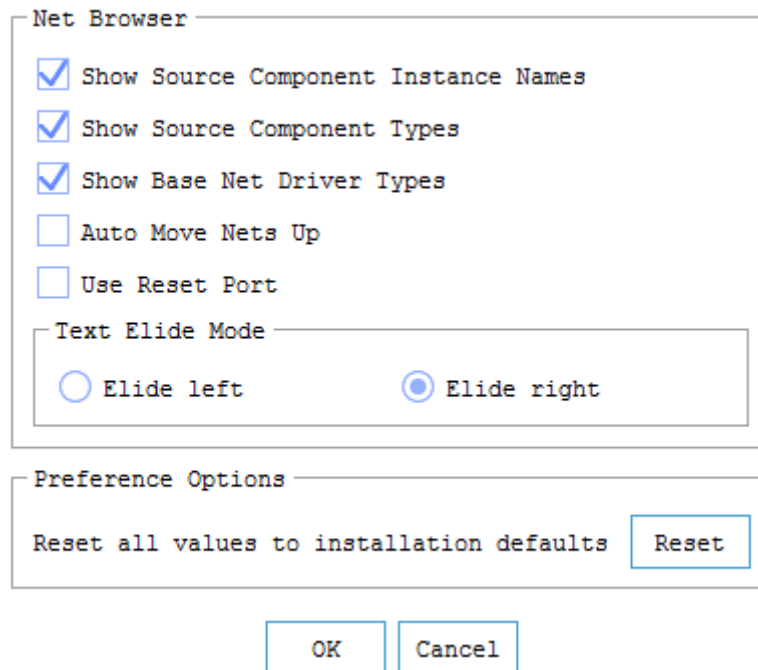


图 3. 26 Edit Preferences 窗口

Edit Preferences 窗口主要功能:

【Show Soure Component Instance Names】: 用于设置 Select Net 窗口中是否显示 Source Instance 列，默认显示。

【Show Source Component Types】: 用于设置 Select Net 窗口中是否显示 Source Component 列，默认显示。

【Show Base Net Driver Types】: 用于设置 Select Net 窗口中是否显示 Base Type 列，默认显示。

【Auto Move Nets Up】: 用于设置 Select Net 窗口中是否在移除一个 Net 的连接后自动将后续 Net 自动上移，默认不启用。

【Use Reset Port】: 用于配置是否启用 Debug Core 的 Reset 功能，不启用 Reset 功能时，Net Connections 窗口和 Select Net 窗口中将不会显示 Reset Port 相关信息，默认不启用。

【Text Elide Mode】: 用于设置 Select Net 对话框中 Net Selections 窗口中的文字省略的方向。

【Reset Button】: 用于将 Net Browser 中的所有设置重置为默认值。

4 RTL 级插核

配合 ADS 综合工具，Inserter 可以实现 RTL 级插核功能。通过在 RTL 代码中添加特殊的标记，综合工具会将需要观察的信号保留下来，同时在 Inserter 中显示时，Inserter 会以其在 RTL 中的名称进行显示。

注：该功能只能配合 ADS 综合工具使用。

4.1 在 RTL 中标记信号

使用 synthesis PAP_MARK_DEBUG 可以对 RTL 代码中的信号进行标记，例如：

```
wire [7:0] debug_this_bus /* synthesis PAP_MARK_DEBUG="true" */;
```

或

```
wire [7:0] debug_this_bus; // synthesis PAP_MARK_DEBUG="1"
```

同时，也可以直接对 RTL 代码中的 Port 进行标记，例如：

```
module my_module (  
    input      clk /* synthesis PAP_MARK_DEBUG="true" */,  
    input      res,  
    output     result  
);
```

需要注意的是：对 port 进行标记时，标记必须写在“,”前，否则标记会被标记到下一个 port 上。

PAP_MARK_DEBUG 的值的类型分为两种：模糊和精确。

4.1.1 模糊标记

模糊标记的值可以是字符串“ture”或以双引号包裹的数字，例如：

```
wire [7:0] debug_this_bus /* synthesis PAP_MARK_DEBUG="true" */;
```

或

```
wire [7:0] debug_this_bus /* synthesis PAP_MARK_DEBUG="1" */;
```


注：PAP_MARK_DEBUG 的值都必须以英文双引号 (") 包裹。

字符串"true"是最简单的标记方式，只用于标记信号，没有其他附加信息。

使用数字进行标记时，除标记信号外，还可以用于对标记的信号进行排序（排序的作用将在后续说明）。

4.1.2 精确标记

精确标记是通过字符串指定信号需要连接的 channel。整个字符串分为三个部分，分别指定调试核序号、port 类型及序号，channel 序号，各个部分用 "/" 分割，整个字符串使用 "<" 和 ">" 包裹，例如：

```
wire [7:0] debug_this_bus /* synthesis PAP_MARK_DEBUG="<0/t1/12>" */;
```

<0/t1/12>含义如下：

0：调试核序号为 0

T1：指定 port 类型为 trigger port（其他类型的 port 表示法：Data port 为 d，Clock port 为 c，Reset port 为 r），port 序号为 1

12：channel 序号为 12

注意：对于只有一个 channel 的 Clock port 和 Reset port，在标记时 port 序号以及 channel 序号均不可省略，将二者均标记为 0 即可，例如：

```
module my_module(
    input  clk /* synthesis PAP_MARK_DEBUG="<0/c0/0>" */,
    input  rst /* synthesis PAP_MARK_DEBUG="<1/r0/0>" */,
    output result
);
```

当一个信号需要进行多次连接时，可以使用 “,” 分隔多个标记值，例如：

```
wire [7:0] debug_this_bus; // synthesis PAP_MARK_DEBUG="<0/t1/12>,<1/d0/5>"
```

注：序号的作用将在后续说明

4.2 通过 Inserter 配置调试核

在 RTL 代码中对需要调试的信号进行标记后,即可点击 PDS 界面中的 Inserter 图标,打开 Inserter 对调试核进行配置(打开 Inserter 前,PDS 会自动执行 Synthesize 等操作)。

打开 Inserter 界面后,所有的调试核配置与第二章及第三章描述相同,在此不再赘述。

特殊的,通过 PDS 打开 Inserter 时,如果 PDS 工程中没有添加 fic 文件,或添加了空的 fic 文件,那么 Inserter 打开后,会根据 RTL 中标记的信息自动进行信号连接。

连接时,以模糊标记的方式进行标记的信号默认会连接到第一个核的第一个 trigger Port 上,以字符串“true”标记的信号先进行连接,以数字标记的信号按数字从小到大排序后依次连接。如果 Trigger Port 的所有 channel 均连接完成,则在当前调试核中创建一个新的 trigger port 继续连接。

当存在以精确标记方式标记的信号时,按照调试核序号、port 类型、port 序号、channel 序号进行连接。连接后,对调试核进行检测,没有任何连接的 trigger port 将被移除,没有任何连接的调试核将被移除。因此最终的连接结果可能不是跟 RTL 中的标记的序号完全匹配,只保证连接顺序与 RTL 中标记的序号一致。

自动连接完成后,用户可根据需求再次手动对连接关系进行调整。

当所有配置及连接关系确定后,保存 fic 即可。

5 TCL 命令

5.1 Inserter TCL Command 简介

现阶段 Inserter 支持的 Tcl 命令如下：

Help

ins_new

ins_open

ins_save

ins_set_file

ins_add_core

ins_remove_core

ins_operate

ins_set _core

ins_set _trig

ins_set _unit

ins_set_condition

ins_set_capture

ins_list_nets

ins_set _net

ins_core_info

5.2 Inserter TCL Command 详细介绍

1. Help

Help 命令用于打印 Inserter 支持的 TCL 命令列表，该命令无参数。

2. Ins_new

该命令用于新建工程文件，可接受一个文件名参数，输入的文件名必须包含绝对路径，也可不输入任何参数，不输入参数时工程名为 Untitled。

例如：ins_new d:/project.fic。

3. Ins_open

该命令用于打开已存在的工程文件，必须输入包含绝对路径的文件名。

例如：ins_open d:/project.fic。

4. Ins_save

该命令用于保存当前的工程，可接受一个文件名参数，输入的文件名必须包含绝对路径，并建议以.fic 结尾，如果输入的文件名不以.fic 结尾，命令会自动加上.fic 后缀。如果不输入文件名参数，命令将检查当前工程是否有可用的文件名，如果没有则命令执行不成功。

例如：ins_save d:/project.fic 可将当前工程保存至 D:/project.fic 文件中。

5. Ins_set_file

该命令用于设置输入的设计文件，该命令支持 1 个 option: -input，用于设置输入的设计文件的文件名，文件名必须包含绝对路径。

例如：ins_set_file -input d:/design.adf 将会设置输入设计文件名为 d:/design.adf。

6. Ins_add_core

该命令用于添加 debug core，可接受一个设置添加 debug core 数量的参数。

例如：ins_add_core 5 将添加 5 个 debug core。

7. Ins_remove_core

该命令用于移除 debug core，可接受一个设置移除 debug core 数量的参数，移除时总是从 debug core 列表的尾部依次移除。

例如：ins_remove_core 3 将移除 debug core 列表最后的 3 个 debug core。

8. Ins_operate

该命令用于设置缺省的操作对象，该命令有 3 个 option: -core、-trig、-unit，分别用来设置 debug core、trigger port、match unit 的缺省操作对象。每个 option 对应一个参数，输入的参数是 debug core、trigger port 或 match unit 的序号，允许同时设置多个用空格分隔的序号，但需要使用 “ ” 标记。

例如：ins_operate -core “1 2 3 4” -trig “1 2 3 4” -unit “1 2 3 4” 将分别设置序号为 1 2 3 4 的 debug core、trigger port 和 match unit 作为缺省的操作对象。

9. Ins_set_core

该命令用来设置 debug core 中的相关参数，共 8 个 option:

-core : 对应一个参数，用于声明操作的 debug core 的序号，如果不输入该参数，将使用 ins_operate

设置的缺省对象，支持同时操作多个 debug core。

-enable_sq：对应一个参数 0 或 1，用于声明是否使能该 debug core 的 storage Qualification，1 表示使能、0 不使能。

-seq_level：对应一个参数，用于设置顺序触发条件的最大级数。

-data_same：对应一个参数 0 或 1，用于声明是否使能 data same as trigger。

-data_depth：对应一个参数，用于设置 data port 的最大存储深度。

-clock_edge：对应一个参数 0 或 1，用于设置触发时钟沿，1 为 Rising，0 为 Falling。

-trig_num：对应一个参数，用于设置 trigger port 的数量。

-power：对应一个参数 0 或 1，用于声明是否使能 power on init data。

-multi_windows：对应一个参数 0 或 1，用于声明是否使能多窗口及 Nsample 功能

各个 option 之间没有冲突可以同时使用。

10. Ins_set_trig

该命令用于设置 trigger port 中的相关参数，共 6 个 option：

-core：对应一个参数，用于声明操作的 trigger port 所属的 debug core 的序号，如果不声明则使用 ins_operate 命令设置的缺省对象，不支持同时操作多个 debug core，如果声明多个 debug core，则使用第一个。

-trig：对应一个参数，用于声明操作的 trigger port 的序号，如果不声明，则使用 ins_operate 命令设置的缺省对象，支持同时操作多个 trigger port。

-counter：对应一个参数，用于设置该 trigger port 的计数器的位数。

-unit_num：对应一个参数，用于设置 match unit 的数量。

-type：对应一个参数用于设置 match unit 的 match type，接受的参数为 match type 对应的序号：

0:Basic

1:Basic W/Edges

2:Extened

3:Extened W/Edges

4:Range

5:Range W/Edges

-as_data：对应一个参数 0 或 1，声明该 trigger port 是否在使能 data same as trigger 时用作 data port。

11. Ins_set_unit

该命令用于设置 Match Unit 中的相关参数，共 6 个 option:

-core : 对应一个参数，用于声明操作的 Match Unit 所属的 Debug Core 的序号，如果不声明，则使用 ins_operate 命令声明的缺省对象。不支持同时操作多个 debug core，如果声明多个 debug core，则使用第一个。

-unit : 对应一个参数，用于声明操作的 Match Unit 的序号，如果不声明，则使用 ins_operate 命令声明的缺省对象。支持同时操作多个 Match Unit。

-func : 对应一个参数，用于声明该 Match Unit 的 Function 类型，可接受的参数有: ==、<>、>=、>、<、<=、In Range、Out of Range，不区分大小写。

-value : 对应一个参数，用于设置该 Match Unit 的 value 值，只接受二进制格式的参数，且输入的参数的位数必须等于该 Match Unit 所属的 Trigger Port 的宽度。如果 Function 类型为 In Range 或 Out of Range 则需要输入 2 个二进制的值，用空格分隔且使用 “ ” 标记。

-cnt_type : 对应一个参数，用于设置该 Match Unit 计数器的类型，可接受的参数为对应类型的序号:

0: Counter Disable

1: Occurring in exactly

2: Occurring in at least

3: Lasting for at least

-cnt_num : 对应一个参数，用于设置该 Match Unit 计数器计数的次数。

12. Ins_set_condition

该命令用于设置 Trigger Condition 或 Storage Qualification, 共有 3 个 option 和一个必须输入的参数:

-core : 对应一个参数，用于声明操作的 debug core 的序号，如果不声明则使用 ins_operate 声明的缺省操作对象。

-trigger : 该 option 无参数，如果声明该 option，命令将会对 trigger Condition 进行配置。

-storage : 该 option 无参数，如果声明该 option，命令将会对 storage condition 进行配置。

必需参数 : 必需参数为设置的 condition 的表达式（表达式写法参考本手册关于 PowerOn Init Parameters 配置界面 Trigger Condition 的介绍），例如：“! TU1&&TU2”，命令只会对输入的表达式进行不完全的检查，所以请确保输入的表达式的正确性。

该命令允许同时操作多个 debug core。

13. Ins_set_capture

该命令用于配置 Storage Style 的相关信息，共有 6 个 option：

-core：对应一个参数，用于声明操作的 debug core 的序号,如果不声明，则使用 ins_operate 命令声明的缺省对象。

-type：对应一个参数 windows 或 Nsamples，不区分大小写，用于声明存储类型。

-windows：对应一个参数，用于声明在 存储类型为 Windows 的情况下的窗口数目。修改该参数，会对应修改每个窗口的存储深度。

-depth：对应一个参数用于声明存储类型为 Windows 的情况下的每个窗口的存储深度。

-position：对应一个参数，用于声明存储类型为 Windows 的情况下，触发点在窗口中的位置。

-samples：对应一个参数，用于声明存储类型为 Nsamples 的情况下的，每个触发周期中点的数量，即 GUI 中的 Samples per Trigger。

该命令允许同时操作多个 debug core。

14. Ins_list_nets

该命令用于将用户设计中可以连接至 debug core 的 net 打印至 console 窗口中，该命令没有参数。由于用于配置连线的 TCL 命令 ins_set_net 在连接 net 时，不接受 net 名为参数（由于某些 net 的名称中含有特殊字符，TCL 进行解析时会出现混乱，并且 net 名称过长不便输入），而是接收 net 在 net list 中的序号为参数，所以如果想准确的进行连接，就需要先使用 list_nets 命令将 net list 打印。

15. Ins_set_net

该命令用于配置 debug core 的连线，共有 10 个 option 和一个参数：

-core：对应一个参数，用于声明操作的 debug core 的序号，如过不声明，则使用 ins_operate 声明的缺省对象，一次仅允许操作一个 debug core。

-trig：对应一个参数，用于声明操作的 trigger port 的序号，允许同时操作多个 trigger port，不可与-all 同时声明。

-all：无参数，声明该 option，命令会将 debug core 中的所有 trigger port 作为操作对象，同时处理。

-data：无参数，声明该 option，命令会将 debug core 中的 data port 作为操作对象。

-clock：无参数，声明该 option，命令会将 debug core 的 clock port 作为操作对象。

-reset：无参数，声明该 option，命令会将 debug core 的 resetport 作为操作对象。

-channel：对应一个参数，当操作对象是 trigger port 或 data port 时，可以使用-channel 来声明操作的具体 channel 的序号，例如：-data -channel 4 表示将 data port 的 channel4 作为操作对象。允许

同时操作多个 channel。

-connect : 无参数, 声明该 option, 命令将会执行连接 net 的操作。连接 net 时, 命令会自动将声明的 net 依次连接至声明的 port 中, 而不会连接至-channel 声明的 channel 中, 以此保证 port 中连接 net 的 channel 是连续的。

-replace : 无参数, 声明该 option, 命令将会执行替换操作, 该操作需要使用-channel 声明需要进行替换操作的具体 channel 的序号, 并且, 只有已经连接了 net 的 channel 才可以进行替换操作, 将以连接的 net 替换为本次声明的 net, 以此保证 port 中连接 net 的 channel 是连续的。

-disconnect : 无参数, 声明该 option, 命令将会执行断开连接的操作, 该操作需要使用-channel 命令声明需要进行断开连接的 channel 的序号, 断开连接后, 后续 channel 会自动上移, 以保证 port 中连接 net 的 channel 是连续的。

参数 : 命令可接受一个可选参数, 该参数用来声明需要连接至 port 的 net 的序号, 仅有-disconnect 操作不需要该参数支持。

例如: `-ins_set_net -core 0 -data -clock -rest "1 2 3 4 5 6"` 执行该命令会将序号为 1、2、3、4、5、6 的 net 依次连接至 debug core0 的 DataPort 中。将序号为 1 的 net 连接至 debug core0 的 clock port 和 reset port 中。

16. Ins_core_info

该命令用于输出显示 Debug Core 的信息, 共 3 个 option:

-core : 对应一个参数, 用于声明需要显示信息的 Debug Core 的序号, 如果不声明该项则显示所有 Debug Core 的基本信息。

-trig : 对应一个参数, 用于声明需要显示信息的 Trigger port 的序号, 如果不声明, 则不会显示任何 Trigger Port 的信息, 支持同时显示多个 trigger port 的信息。

-unit : 对应一个参数, 用于声明需要显示信息的 Match Unit 的序号, 若果不声明, 则不会显示任何 Match Unit 的信息, 支持同时显示多个 Match Unit 的信息。

6 免责声明

版权声明

本文档版权归深圳市紫光同创电子有限公司所有，并保留一切权利。未经书面许可，任何公司和个人不得将此文档中的任何部分公开、转载或以其他方式披露、散发给第三方。否则，公司必将追究其法律责任。

免责声明

本文档仅提供阶段性信息，所含内容可根据产品的实际情况随时更新，恕不另行通知。如因本文档使用不当造成的直接或间接损失，本公司不承担任何法律责任。

本文档按现状提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

公司保留任何时候在不事先声明的情况下对公司系列产品相关文档的修改权利。