

# BDTI研究认证以DSP为核心的 FPGA设计的高级综合流程

基准测试公司发现，高级综合工具的发展简化了DSP开发人员用FPGA实现设计的工作。



作者: Jeff Bier  
伯克利设计技术公司总裁  
bier@bdti.com

Jennifer Eyre White  
伯克利设计技术公司  
DSP 分析师  
eyre@bdti.com

近年来,高级综合工具已成为在设计方案中使用或希望使用 FPGA 的工程师的必杀技。这种工具以应用的高级表示法(比如用 C 语言或 MATLAB 的 M 语言编写的表示法)为输入,并生成面向 FPGA 的硬件实现的寄存器传输级 HDL 描述。

高级综合工具(HLST)对两种类型的潜在用户非常有用,一是正在实现高强度数字信号处理(DSP)应用的 FPGA 用户,二是正在实现高强度数字信号处理(DSP)应用的高性能 DSP 处理器用户。这是因为信号处理工作负载非常繁重,通常需要较高的数据速率和高级并行处理能力,这种需求往往适合采用 HLST 的 FPGA 来实现。

对目前的 FPGA 用户来说,HLST 工具有望简化并加速设计进程。而对目前的 DSP 处理器用户而言,HLST 则提供了一种独特而相当富有吸引力的作用,无需编写 RTL 代码就能移植到更强大的处理引擎 FPGA 上。因此,何乐而不为呢?

关键在于,过去的高级综合工具都不能生成高效的 RTL 代码(就资源使用率而言)。大多数工程师都不愿牺牲手编的 RTL 代码的性能和效率,因此这种工具未能赢得较大市场份额。不过,最近出现的一些新鲜事例显示,面向赛灵思 FPGA 的新型 HLST 工具效率和易用性都非常高。在这种自相矛盾的情况下,潜在用户如何评判高级综合工

具到底是否值得考虑一下呢?

为了回答这一问题,独立基准测试与分析公司 BDTI 于 2009 年制定了 BDTI 高级综合工具认证计划。我们的目标是针对 FPGA 的 HLST 提供客观可信的数据与分析,确保潜在用户能快速了解其在高强度信号处理应用中的功能及局限性。评估时,我们是站在没有 FPGA 开发经验但却具有丰富 DSP 软件开发经验的工程师角度,这也反映了大量潜在受益于 HLST 的处理器用户的实际情况。

最初进行评估的两个 HLST 程序是 Synfora 的 PICO 和 AutoESL 的 AutoPilot。2010 年初,我们发布了首次评估测试计划的结果,其中的一些结果让许多 FPGA 和 DSP 处理器用户都感到吃惊。

### 采用 HLST 实现应用

我们采用 HLST 实现测试应用的步骤先是对所需功能进行高级语言描述,高级综合工具由此生成 RTL 实现。随后赛灵思的 RTL 工具(集成综合环境(ISE)和嵌入式开发套件(EDK))将 RTL 实现转化为比特流形式的完整 FPGA 实现,用于对带有 I/O 和存储器的特定硬件平台上的特定赛灵思 FPGA 进行编程。在本案例中,我们使用的平台为赛灵思的 XtremeDSP™ 视频入门套件——Spartan®-3A DSP 版本,这是一款基于 Spartan-3A DSP FPGA 的目标设计平台。

我们本可以评估仅限于高级综合工具,忽略设计流程中的“RTL 到比特流”部分的评估。不过我们认为,潜在用户应该了解从高级应用描述转变到 FPGA 实现的整个流程情况,这项工作需要 RTL 工具以及高级综合工具。因此,我们对整个实现流程进行评估,不光是“C 到 RTL”部分,还包含赛灵思 RTL 工具链。

在任何硬件目标上实现应用的第一步通常都是重建初始 C 代码。我们所说的“重建”是指重新编写初始的 C 代码(初始的代码主要是为了让代码变得更清晰,便于了解,而不是出于性能考虑),将其转变为更适合于目标处理引擎的格式。比方说,在 DSP 处理器上,我们或许应当重新安排应用的控制流程,确保中间数据始终适应高速缓存要求。就面向 FPGA 的高级综合工具而言,重建通常要提供应用表示法,确保工具能提取潜在的并行性,从而优化流水线实现。

一般而言,高级综合工具都不能自动进行重建,而是需要手工重建。事实上,设计人员可以不借助高级综合工具来进行重建。比方说,在我们的评估中,我们用 Microsoft Visual Studio 重建并验证 C 代码。相对于重建和语言翻译作为统一步骤出现的手编 RTL 代码而言,完全用 C 语言进行重建相对比较简单,出错的可能性也少,这是高级综合工具的一大优势。

重建高级代码后,用户让 HLST 综合生成 RTL HDL 代码形式指定功能的硬件实现。赛灵思的 RTL 工具(ISE 和 EDK)采用 HLST 生成的 RTL 代码,执行综合和布局布线任务,并报告实现的资源利用率,以及通知用户各种时序问题。

### BDTI 的工具认证计划

BDTI 制定高级综合工具认证计划的目标就是进行两大关键点对比,以满足两类潜在 HLST 用户需求。首先,我们希望比较基于 HLST 的 FPGA 应用实现和基于手编 RTL 代码实现的效率,这一信息对目前正在考虑是否要采用 HLST 加速开发时间的 FPGA 用户而言至关重要(就资源使用率而言)。其次,我们希望用面向 FPGA 的 HLST 与用 DSP 处理器及相关软件开发工具实行同一工

此前，用手写 RTL 代码在 FPGA 上实现的高要求应用通常能实现较高的质量，但生产力不高，而用 DSP 处理器实现的速度很快，但结果质量相对较差。

作负载时的性能与难度比较。这一对比使 DSP 处理器用户能估算出将技术移植到 FPGA 设计方案上的难度有多大。

我们用明确定义的样本应用（即“工作负载”）评估高级综合工具流程（包括相关的 RTL 工具）。上述应用（下一节做了简要说明）主要是指代表设计人员通常在 FPGA 上实现的高强度数字信号处理应用，数据速率和计算强度均要求高。其他类型的应用产生结果会与这里给出的结果可能会有所不同。

我们用不同方法实现这两种应用。首先，我们将高级综合工具与赛灵思 RTL 工具结合使用，在目标 FPGA 上实现给定的工作负载。随后，采用传统的 RTL 设计方法，或者用 DSP 处理器配合其相关开发工具（取决于具体的工作负载），在相同的 FPGA 上实现相同的工作负载。这样，我们就能比较出采用不同工具与芯片组合实现的应用的结果质量与生产力。

### 评估工作负载

我们用于评估目的的两个应用是 BDTI Optical Flow Workload 和 BDTI DQPSK Receiver Workload。

“光流法”这个术语是指一种分析场景中对象运动和对象特性（如边缘）的视频处理算法。BDTI Optical Flow Workload 操作 720p 分辨率（1,280 x 720 逐行扫描）的输入视频序列并生成一系列二维矩阵，确定序列中的水平及垂直运动。在设计工作负载时，我们集

成了动态数据依赖性决策和阵列索引，从而确保为工具提供严格的测试。

BDTI Optical Flow Workload 涉及两个操作点，每个操作点都使用相同的算法，但针对不同的标准优化。操作点 1 是固定工作负载，定义为每秒 60 帧的 720p 分辨率处理视频。操作点 1 的目的就是最大限度的减少实现指定分辨率所需的资源利用率（资源利用率指的是实现工作负载所需资源占可用处理引擎资源的比例）。

与此同时，操作点 2 的目的则是用所有可用资源来最大限度地提高吞吐量（以每秒帧数为单位）。

第二个工作负载 BDTI DQPSK Receiver Workload 为无线通信接收器基带应用，其中包含许多无线收发器中的传统通信模块。这是一种固定工作负载，只有一个单一操作点来处理输入流，输入流为复数调制数据，速率为每秒 1875 万次采样，接收器链的时钟速率为 75 MHz。接收器可生成每秒 4.6875 M 比特的解调输出比特流。该工作负载的目的是最大限度地减少实现指定吞吐量所需的 FPGA 资源的使用。

对不同工作负载而言，存储器的使用与存储器的带宽要求差异较大。BDTI DQPSK Receiver Workload 只需使用最小的存储器（因此无需外部存储芯片），而 BDTI Optical Flow Workload 则需要存储四个视频帧（每帧 1,280 x 720 像素）的历史记录，因此需要外部存储芯片来配合 Spartan-3A DSP

FPGA。Optical Flow Workload 操作点 1 需要单一外部存储芯片与接口（带宽约为每秒 450 MB），而 Optical Flow Workload 操作点 2 通常需要两个外部存储芯片与接口，其带宽之和约为每秒 1.4GB。

就 BDTI Optical Flow Workload 而言，在典型的 FPGA 实现中，操作点 1 每个时钟循环处理一个像素，而操作点 2 每个时钟循环则处理两个像素。BDTI DQPSK Receiver Workload 实现每四个时钟循环处理一个输入样品。

### 度量指标和平台介绍

此前，虽然用手写 RTL 代码在 FPGA 上实现的较高要求应用通常能实现更优异的质量（即性能和效率较高），但工作效率不高；而用 DSP 处理器实现的应用工作效率虽高，但结果质量差强人意。面向 FPGA 的高级综合工具旨在提供二者相结合后的最佳水平，既提供较高质量，又实现较高效率。因此，我们在评估中考虑了两组度量指标：结果质量和可用性。

结果质量度量指标用于评估工作负载实现的性能和资源使用情况。BDTI Optical Flow Workload 提供 HLST-Xilinx 流程和 DSP 处理器流程的结果质量度量指标。BDTI DQPSK Receiver Workload 提供 HLST-Xilinx 流程以及赛灵思根据典型业界设计实践采用手写 RTL 设计的传统 FPGA 实现方案的结果流程，其中包括在适当的时候使用赛灵思 CORE Generator™ 知识产权模块。

可用性度量指标用于评估 HLSTXilinx 设计方案的工作效率和易用程度，并以我们实现 BDTI Optical Flow Workload 的情况为参考依据。这些度量指标将面向 FPGA 的 HLST 和赛灵思工具流相对于使用 DSP 处理器及其相关软件开发工具链的工作效率和易用性进行比较。我们根据工具使用的九个方面量化评估可用性度量指标，其中包括直接体验、易用性、工具功能完整性、整体设计方法的效率以及文档与支持质量等。

就本次评估而言，目标 FPGA 为赛灵思 Spartan-3A DSP 3400 (XC3SD3400A)；就 BDTI Optical Flow Workload 而言，赛灵思 XtremeDSP Video Starter Kit—Spartan-3A DSP Edition 为目标平台。我们使用了赛灵思 RTL 工具，包括 ISE 和 EDK 工具套件（版本 10.1.03, lin64）以及高级综合工具等。

本项目中的目标 DSP 处理器为德州仪器的 TMS320DM6437。这款面向视频的处理器采用 600-MHz TMS320C64x+ DSP 内核以及视频硬件加速器（硬件加速器不适用于 BDTI Optical Flow Workload，因此我们并未使用）。评估中采用了德州仪器的 DM6437 Digital Video Development Environment 作为目标平台，并采用了德州仪器的 Code Composer Studio 工具套件（版本为 V3.3.82.13, Code Generation Tools 版本为 6.1.9）。

### 实现与认证进程

我们将两个工作负载的实现工作分配给两个芯片，并根据使用的芯片和工具链来选择采用赛灵思和 BDTI 不同的高级综合工具厂商。HLST 厂商使用其自有工具以及赛灵思的工具实现两个工作负载，并将性能和资源使用结果提交给 BDTI 进行验证与认证。我们通过经

认证的结果生成了本文的结果质量度量指标。

与此同时，我们的工程师接受了 HLST 厂商的培训，并分别用高级综合工具和赛灵思工具独立实现了 BDTI Optical Flow Workload 的某些部分。这一过程为 BDTI 提供了工具链易用性及其生成结果质量的第一手信息。我们还在 DSP 处理器上实现了 BDTI Optical Flow Workload，而赛灵思则实现手写 RTL FPGA 版的 BDTI DQPSK Receiver Workload（随后也经 BDTI 验证与认证）。

### 结果质量：性能和效率

如图 1 所示，用高级综合工具为 BDTI Optical Flow Workload 创建的 FPGA 实现方案的性能比 DSP 处理器实现方案高出约 40 倍。我们再分析一下芯片成本，图 2 所示的是相应的性价比优势，FPGA 实现方案大约有 30 倍

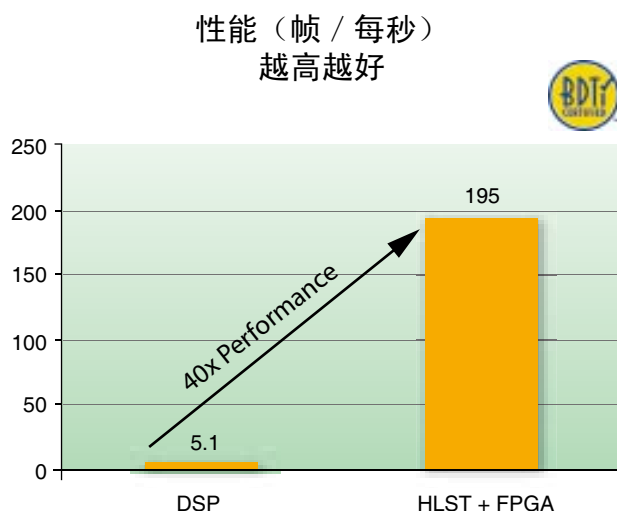


图 1——使用 HLST 的 Spartan-3A DSP FPGA 在 BDTI Optical Flow Workload 视频应用上以 720p 的分辨率获得了每秒 195 帧的成绩，而同样情况下 C64x+ DSP 处理器的成绩仅为 5.1 帧每秒。

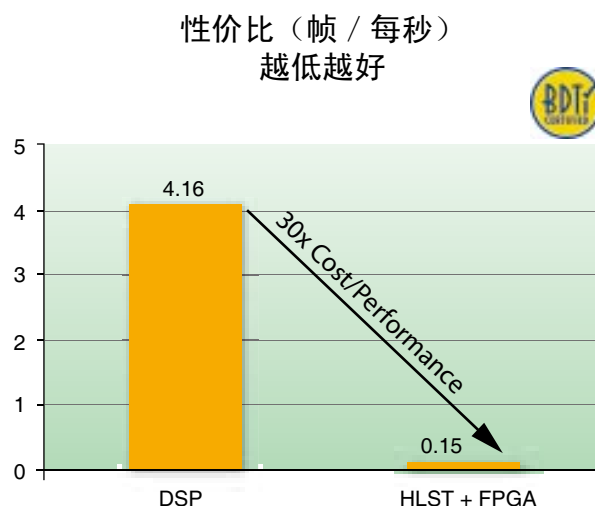


图 2——Spartan-3A DSP FPGA 上使用 HLST 的 BDTI Optical Flow Workload (720p) 性价比优于 600-MHz TI C64x+ 架构的 DSP。

的优势。显然，配合高级综合工具使用的 FPGA 能为某些类型的应用提供出色的性价比优势（更多详细结果请参见 [www.BDTI.com](http://www.BDTI.com)。）

我们还就 BDTI DQPSK Receiver Workload 对基于 HLST 的 FPGA 实现方案与采用手写 RTL 实现相同工作负载的效率进行了评估。我们再次发现，HLST 的表现非常出色。如表 1 所示，AutoPilot 和 PICO 都能生成效率（即资源使用情况）相当于手写 RTL 代码的效率结果并非偶然；我们在评估工作开始时就为 AutoESL 和 Synfora 提供了手写 RTL 实现方案的资源使用示意图，这两家公司可能用该图作为其优化实现方案的目标（不过我们也应当注意，这种信息并非高效使用高级综合工具所必需的，而且 HLST 厂商并未获得 RTL 设计方案）。

我们还采访了使用 AutoESL 和 Synfora 高级综合工具的设计人员，他们确认了资源使用情况的真实性。事实

| 平台   | 芯片资源使用<br>(越低越好) |
|--|------------------|
| 面向赛灵思 XC3SD3400A FPGA 的 HLST<br>加赛灵思 RTL 工具      | 5.6% - 6.4%      |
| 采用赛灵思 RTL 工具、面向赛灵思<br>XC3SD3400A FPGA 的手写 RTL 代码 | 5.9%             |



表1——BDTI DQPSK Receiver Workload 的资源使用情况，输入数据速度为每秒 18.75 M 次采样，时钟速度为 75-MHz。

上，他们表示该工具所产生的结果优于通过手写 RTL 代码所能实现的结果，而且还减少了大量设计和验证工作——这是一个巨大的成就。

### 易用性度量指标

我们的易用性度量指标主要用于评估高级综合工具流程相对于 DSP 处理器工具链的易用性。对每个易用性度量指标而言，我们都相应给出优秀、很好、一般和差等评分。在打分时，我们

要考虑完整项目的整体设计方法——先是 C 语言应用规范，最后是目标芯片（FPGA 或 DSP 处理器）上的实时实现。易用性度量指标如表 2 所示。

通常，PICO 和 AutoPilot 都便于安装和使用，甚至无需 FPGA 设计经验就能进行。与此形成对比的是，我们在安装和使用赛灵思的 RTL 工具时会遇到了很大困难，最终还是决定邀请经验丰富的 FPGA 工程师协助设置运行

|                       |      |     |       |         | 设计方法效率           |                   |                   |          |             |
|-----------------------|------|-----|-------|---------|------------------|-------------------|-------------------|----------|-------------|
|                       | 直接体验 | 易用性 | 功能完整性 | 文档和支持质量 | 学习使用工具           | 设计和实现<br>(首个编译版本) | 设计和实现<br>(最终优化版本) | 平台基础设施开发 | 参考代码所需的修改幅度 |
| HLST + 赛灵思 RTL 工具综合评级 | 普通   | 好   | 好     | 好       | 很好               | 很好                | 好                 | 好        | 好           |
| 德州仪器软件开发工具评级          | 好    | 很好  | 很好    | 很好      | 未知<br>(假设早已为人熟知) | 优秀                | 好                 | 好        | 普通          |



表 2——HLST 和 FPGA 工具与 DSP 开发软件的易用性度量指标对比

## 在权衡可编程 DSP 处理器和 FPGA 的使用问题时，开发时间对许多系统设计人员而言都是一大障碍。我们的评估显示，采用高级综合工具的新方法在很大程度上消除了这一障碍。

在 FPGA 上的设计方案。例如，我们需要让 FPGA 工程师解读赛灵思 RTL 工具发出的错误消息，并将 HLST 生成的 RTL 模块与 I/O 和存储器模块连接在一起，以生成可在 FPGA 上运行的完整设计方案。总体而言，我们发现在高级综合工具之外发生的设计问题相当难于解决。如果 HLST 用户没有 RTL 设计和工具技能（我们就不具备这方面的技能），在设计流程的这一阶段就需要具有相关经验的工程师予以帮助。

不过，即便考虑到与流程中 RTL-比特流部分相关的挑战，我们从表 2 中仍能看出，HLST-Xilinx 工具链仍能够实现出色的易用性和生产力业绩，且并不比 DSP 处理器流程所产生的业绩差。总之，如果有经验丰富的 FPGA 工程师帮助我们开展流程中的某些工作，则我们发现在 TI DSP 处理器上实现 BDTI Optical Flow Workload 的工作量与采用两种 HLST 在赛灵思 FPGA 上加以实现的工作量基本相同。

上述结论意义重大，可能会让很多 DSP 软件工程师感到惊讶。在比较对可编程 DSP 处理器和 FPGA 的使用时，开发时间对许多系统设计人员来说都是一大障碍。我们的评估显示这种新方法很大程度上为 BDTI Optical Flow Workload 等应用消除了时间障碍。

### HLST：规则改变者？

我们早先对 FPGA 和 DSP 处理器进行的基准测试（发表于 2007 年的“用 FPGA 完成 DSP 工作任务”报告

中）显示，用传统 RTL 设计方法实现 FPGA 时，FPGA 在某些应用中具有较大的性能和性价比优势。新的分析结果确认了上述性能优势（例如，BDTI Optical Flow Workload 上的速度提升了 40 倍、性价比提升了 30 倍），反映出 FPGA 配合使用高级综合工具也可实现类似的性价比优势。此外，我们发现，我们目前所评估的两种高级综合工具——Synfora 的 PICO 和 AutoESL 的 AutoPilot——所实现的资源使用效率相当于通过手写 RTL 代码所实现的效率。尽管我们没有直接评估 HLST 相对于手写 RTL 代码的时间节约优势，但根据我们对目前 HLST 用户的采访结果，我们相信时间方面的优势也会相当显著。

用传统手写 RTL 编码创建的 FPGA 设计方案通常比 DSP 处理器上用软件实现的同类应用要花费更加多的工作量——这较好地解释了很多 DSP 处理器用户不愿意改变工作方法的原因。因此，本项目最令人惊讶的结果或许就在于，在 FPGA 上（用 AutoPilot 或 PICO 配合赛灵思工具）实现评估工作负载与使用 DSP 处理器所需的工作量基本相当。

对 FPGA 用户而言，我们的研究充分说明 HLST 可在无显著负面影响的情况下提升生产力。而对于 DSP 处理器用户，FPGA 显而易见确实值得考虑——HLST 正快速成为改变格局的重大技术。🌟

## 赛灵思 28nm 全新 7 系列 FPGA 品牌发布了！

**VIRTEX<sup>7</sup>**

业界领先的 Virtex 品牌代表了 FPGA 的最高水平和最强功能。Virtex-7 系列可提供业界最高的系统性能和最强大的功能。全新 Virtex-7 系列是目前业界容量最大的器件（包括 200 万个逻辑单元），与前代产品相比，系统性能翻了一番。Virtex-7 旨在满足有线基础设施、高性能计算系统以及航空与国防等应用的高性能需求。同时，该系列还包括 EasyPath-7 成本降低解决方案，有助于实现大批量生产。

**KINTEX<sup>7</sup>**

Kintex 的名称取自“kinetic”和“kin”，分别代表着“运动、活力”和“相关性”。Kintex 实现了业界最佳性价比。全新 Kintex-7 能以 Virtex-6 一半的成本实现与其相当的性能，相对于 Virtex-6 和 Spartan-6 而言，显著提高了性价比，不仅能满足严格的功耗和成本要求，还能满足新一代广播系统和无线网络等应用对带宽的巨大需求。

**ARTIX<sup>7</sup>**

Artix 取自“Arctic”，即北极，意味着“冷”与“低功耗”。Artix 是业界功耗和成本最低的 FPGA 系列产品。Artix-7 系列产品比 Spartan-6 系列的功耗减少一半，成本降低 35%，从而使其成为业界领先的产品系列，理想适用于此前 ASSP 和 ASIC 所针对的低成本、大批量市场。