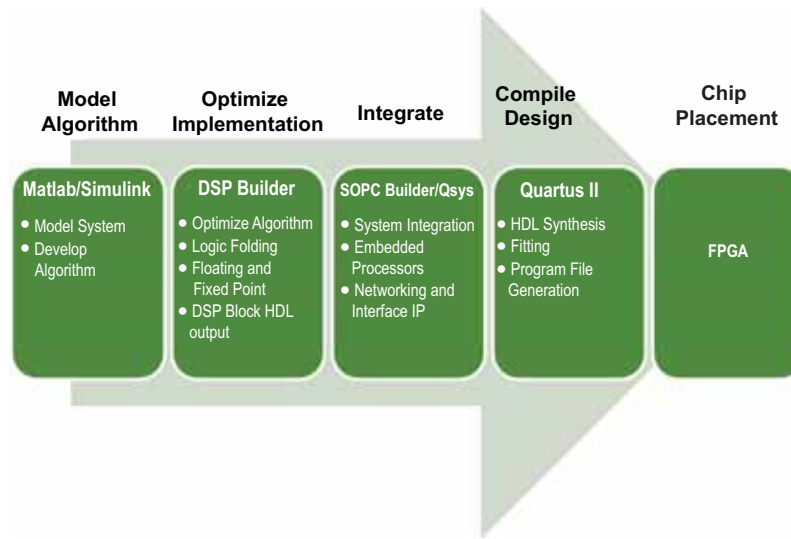


本文档推荐一种设计流程，它在电机控制设计中利用了 Altera FPGA 强大的适应能力、精度可调 DSP 以及集成系统设计工具。工业电机驱动设备的设计人员可以充分发挥这一设计流程的性能、集成和效率优势。

引言

在工业能耗中，三分之二以上的能耗来自工业电机驱动设备，因此，在工厂费用开支中，高效的电气特性是非常重要的因素。在电机驱动系统中采用速率可变驱动器 (VSD) 来替代传统的驱动器能够显著提高效率，能耗节省了近 40%。Altera 的 FPGA 体系结构具有图 1 所示的灵活性、性能、集成以及设计流程优势，是 VSD 系统的高效平台：

图 1. 电机控制 FPGA 优化设计流程



- 调整性能——通过并行处理以及灵活的功能进一步提高各种类型电机的性能和效率。设计集成——在一片器件中集成嵌入式处理器、编码器接口、DSP 运动控制算法以及工业网络。
- 灵活的设计——重新使用 IP，采用精度可调 DSP 模块。在控制通路的任意部分都可以实现定点或者浮点精度。
- 确定性延时——在硬件中实现电机算法以及确定性操作。

- 功能强大而又流畅的工具——使用 Simulink 等建模工具，结合 Altera 的 DSP Builder 以及 Qsys 或者 SOPC Builder 通用集成工具，优化低成本 FPGA 中的全电机系统。虽然通常使用商用微控制器 (MCU) 或者数字信号处理器 (DSP) 来实现处理和控制环，监控负载并调整位置、速度和其他驱动功能，但是，微控制器不够灵活，在性能上也有限制。在算法非常复杂的系统中，需要较高的 MIPS 处理能力，这些缺点尤其明显。而且，在软件中编写代码很难实现硬件最优系统。

同样的，虽然高端 DSP 一般能够处理电机控制计算功能，但是，高端 DSP 并不适用于需要同时处理高精度时间运算以及面向任务运算的系统，例如，存储器接口、信号接口和滤波，或者支持工业以太网协议标准等。

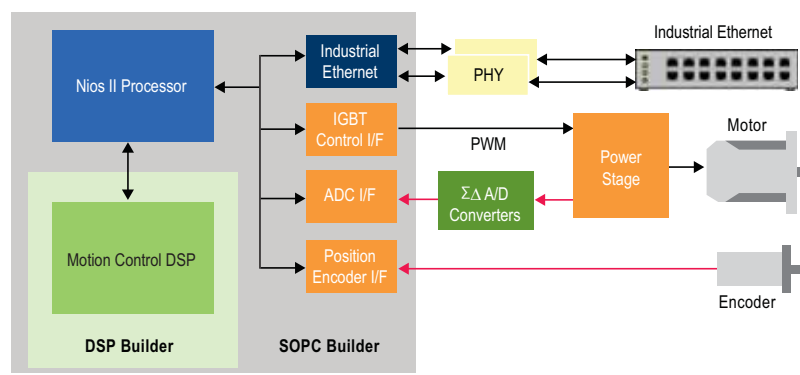
性能调整和集成优势

很多商用 MCU 或者 DSP 具有通用驱动工作基本单元。但是，这些器件存储器有限，模拟范围较窄，PWM 通道数量有限，对多轴系统的支持不足。下一代驱动功能对性能的要求更高，需要提高电机效率，所采用的平台应具有性能调整功能，以满足处理和 DSP 需求，同时能够灵活的集成并优化系统。

FPGA 很容易根据应用需求来调整性能。设计人员可以在 FPGA 中嵌入多个处理器或者使用灵活的 DSP 功能，然后，采用其他的逻辑、定制指令或者所支持的多种工业网络协议。设计人员利用 Altera FPGA 可以实现多种嵌入式处理器，分别控制每一个子系统。Altera FPGA 的并行特性支持电机控制系统构建模块的集成。例如，Altera Nios II 嵌入式处理器 (32 位 RISC 软核处理器) 可以控制各种接口以及传感器和编码器。设计人员可以使用精度可调浮点 DSP 模块完成现场定位控制 (FOC) 或者其他需要大量数学计算的算法。

图 2 所示为能够集成到 FPGA 中的各种单元，用于建立一个“单芯片驱动”系统。集成 IP 功能能够并行运行，保证了在顺序操作或者延时操作中不会出现瓶颈。

图 2. FOC 模型包括复杂数学算法



这一设计流程支持 IP 的集成，包括：

- 位置反馈——具有高精度位置反馈功能的编码器，例如，EnDAT、Hiperface，以及 BiSS，支持 10 倍速和位置数据。
- IGBT 控制——使用绝缘栅极双极晶体管 (IGBT)，切换到驱动 AC 电机所需要的高电压。在 FPGA 中使用空间矢量调制 (SVM) 技术，对 IGBT 栅极输入进行脉冲宽度调制 (PWM)，产生驱动电机所需要的正弦电压波形。IGBT 可以是 2 级或者 3 级变量。
- ADC 接口——与外部模数转换器 (ADC) 连接，测量电机的反馈电流。很容易将 Sigma-delta ADC 与大驱动电压实现光电隔离，降低噪声，支持 FPGA 对其输出进行采样，实现快速精确的读取操作。
- 网络接口——在 FPGA 中实现实时协议，以适应实际应用所需要的工业以太网协议标准，例如，Ethernet/IP、PROFINET IO/IRT 和 EtherCAT。工业以太网在工业驱动中的应用越来越广泛。

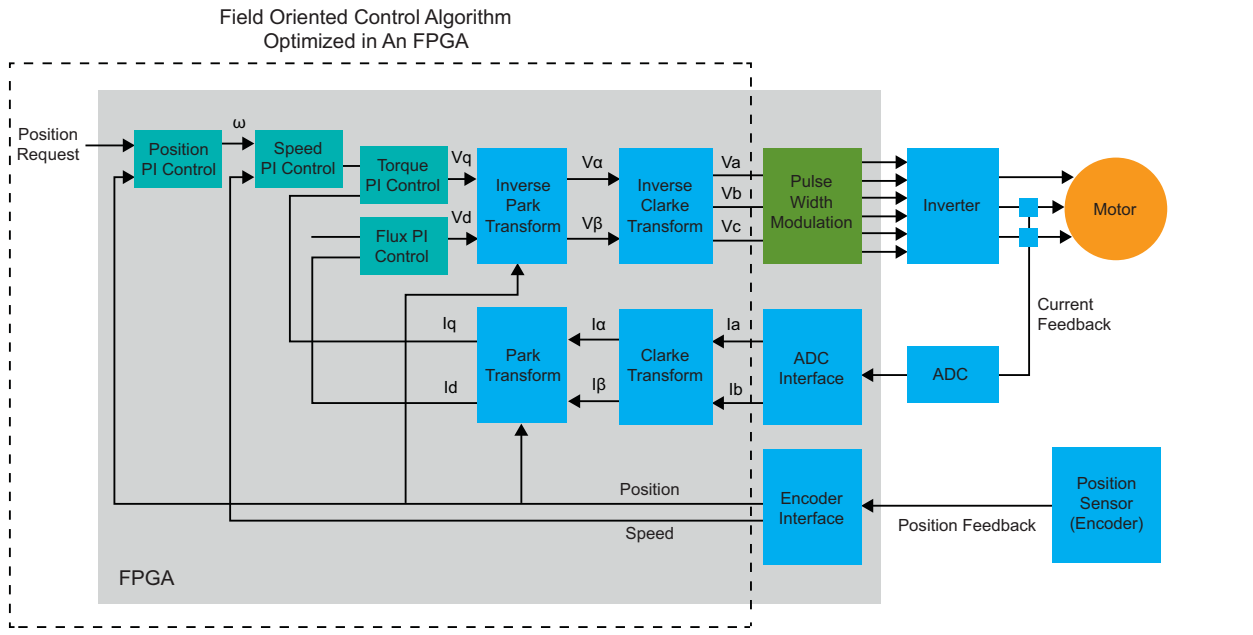
这些基于 DSP 的电机控制功能、通信以及接口标准的大量应用使得 FPGA 成为工业电机驱动的理想平台。

处理复杂数学算法

各种驱动技术的差别很大，具体取决于电机类型及其应用。例如，控制电泵和风扇的驱动器有不同的要求和反馈机制，以便控制 CNC 电机或者封装设备。从这些编码器和传感器中采集到的数据被反馈到控制系统中，用于进行数学计算，以确定目标系统负载和扭矩的电压电平。

例如，常用的永磁体同步电机 (PMSM) 使用需要大量数学算法的现场定位控制 (FOC) 功能作为控制环算法的一部分，该功能也称为矢量控制。在需要精确扭矩控制的工业伺服电机中，也经常采用 FOC。FOC 技术有助于减小电机体积、成本和功耗。FOC 精确的测量电压电平以及相应的电机速度，更好的控制速度和扭矩，即使负载变化了也能够提供恒定扭矩。FOC 还减小了扭矩波纹和电磁干扰。但是，如图 3 所示，这一数学模型非常复杂，高速运行这一算法时需要很强的计算能力。

图 3. 现场定位控制 (FOC) 模型



FOC 涉及到对电机正弦 3 相电流的实时控制，以建立平滑的旋转磁通量模型，其旋转频率对应于正弦波的频率。该方法控制电流矢量的振幅，将其维持在相对于转子磁通量轴 90 度的位置（“正交”电流）。这样，设计人员可以控制扭矩，同时将“直流”电流分量（0 度）保持在零。这一算法涉及到以下步骤：

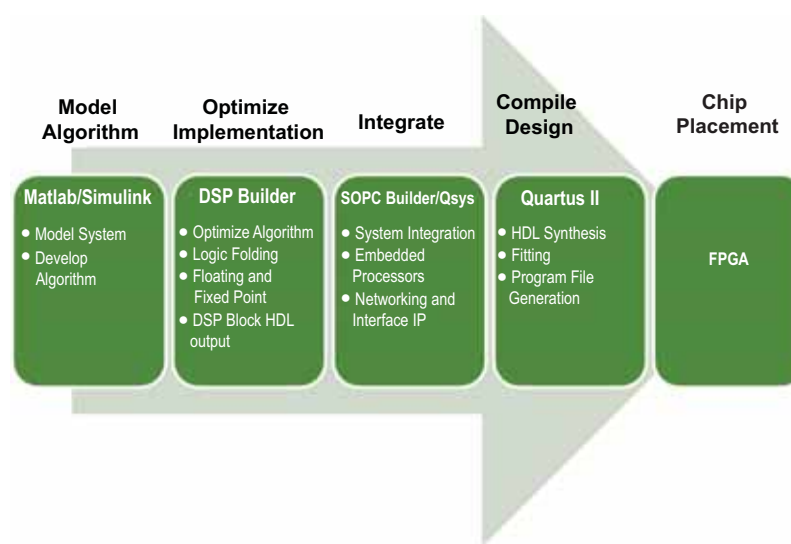
1. 使用 Clarke 和 Park 变换，将编码器的 3 相反馈电流输入和转子位置转换为“正交”和“直流”电流分量。
2. 使用这些电流分量作为两个并行运行的比例和积分 (PI) 控制器的输入，将“直流”电流限制为零，把“正交”电流保持在所需的扭矩水平上。
3. 通过 Clarke 和 Park 反变换，将 PI 控制器的“直接”和“正交”电流输出转换成 3 相电流。

Altera FPGA 提供业界的第一款精度可调 DSP 模块，能够灵活的选择满足要求的精度级，而且还支持单精度和双精度浮点类型。这些因素使得 DSP 模块成为实现 FOC 控制环和其他复杂数学算法的理想选择。集成 DSP 模块是很多 Altera 28-nm FPGA 体系结构所具有的特性，支持每一模块在编译时配置为 18 位或者高精度模式。

利用功能强大的开发工具

优化电机控制设计需要通用工具（以及实用的工具流程）对系统进行建模和仿真，实现低延时复杂算法，将系统集成在一起，精确的调整性能，以满足电机驱动的实际需求。设计人员使用图 4 所示的集成工具流程，能够缩短开发时间，提供更灵活、功能更强大的模型，以适应不同类型的驱动系统。

图 4. 电机控制 FPGA 优化设计流程



Altera 为嵌入式工业设计人员提供功能强大而且使用方便的开发工具，例如，Quartus® II 设计软件和 MegaCore® IP 库。Altera 还提供系统集成工具，例如 Qsys 或者 SOPC Builder 实现面向任务的操作，以及 DSP Builder 来优化 DSP，如图 4 所示。此外，Altera 提供基于 Eclipse 的 Nios® II 嵌入式设计套装 (EDS)，更方便的通过设计流程来完善 FPGA 硬件。

Nios II 嵌入式设计套装

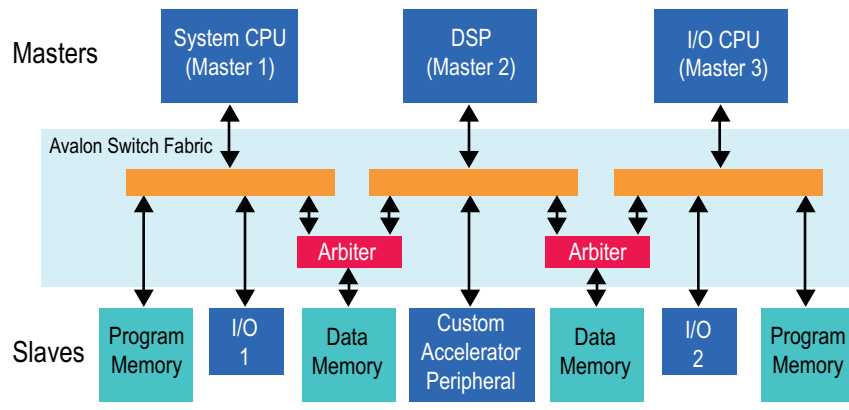
Altera 提供基于 Eclipse 的 Nios II EDS、Nios II 嵌入式处理器等功能强大而且使用方便的嵌入式开发工具，支持标准操作系统 (OS) 以及来自多家流行供应商的实时操作系统 (RTOS)。Nios II EDS 能够对多种通用 32 位 RISC 软核处理器进行例化。这些处理器的性能高达 340 MIPS (Dhrystones 2.1)，可以采用自己的定制指令集、数据通路和地址空间来独立运行。

Qsys 和 SOPC Builder 系统集成工具

Altera 的 Quartus II 开发软件包括最新的 Qsys (以及传统的 SOPC Builder) 系统集成工具。如图 7 所示，这些工具有助于设计人员自动完成硬件组件集成任务，定义并生成完整的芯片系统 (SoC)。Qsys 或者 SOPC Builder 并不是使用传统的设计方法来手动定义和连接 HDL 模块，而是在 GUI 中帮助您定义系统组件，然后自动生成互联逻辑。这些工具生成定义系统中所有组件的 HDL 文件，然后，顶层 HDL 文件将所有组件连接起来。这些工具生成 Verilog HDL 或者 VHDL。

这些系统集成工具使用 Avalon 接口来连接逻辑器件 (片内或者片外)。在电机系统中，Avalon 接口连接软核处理器以及驱动系统的其他单元。这降低了系统集成的复杂度，实现了连续直观的最优系统。互联架构支持通过从机侧仲裁进行同时多主机处理，从而管理这些连接。这些工具在每一从机端口前插入仲裁模块，以满足不同主机的要求，从主机和从机端口中提取出系统详细的互联信息。图 5 显示了连接系统中多个从机和主机的互联架构。

图 5. 采用 Qsys 或者 SOPC Builder 管理系统互联



在 DSP Builder 设计中，Altera 的系统集成和嵌入式开发工具帮助设计人员迅速构建连接处理器和硬件加速电机控制算法的接口。

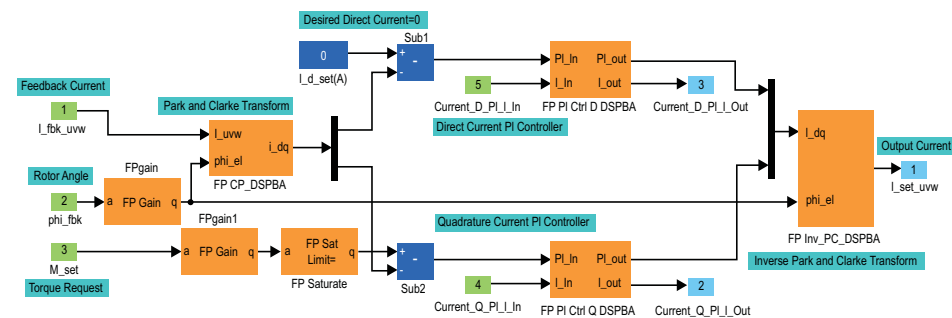
DSP Builder 和 Matlab/Simulink

电机控制系统设计人员可以采用 FPGA 中的 DSP 功能来实现需要大量数学功能的高速电机控制算法。Altera 的 DSP Builder 在友好算法开发环境中，帮助设计人员生成 DSP 设计的硬件表征，缩短了 DSP 设计周期。DSP Builder 结合了 The MathWorks MATLAB® 和 Simulink® 系统级设计工具以及 Altera Quartus II 软件和第三方综合和仿真工具，实现算法开发、仿真和验证功能。您可以组合使用 Simulink 模块和 DSP Builder 模块以及 IP 模块，验证系统级规范，进行仿真。

在 FPGA 中对 FOC 进行建模

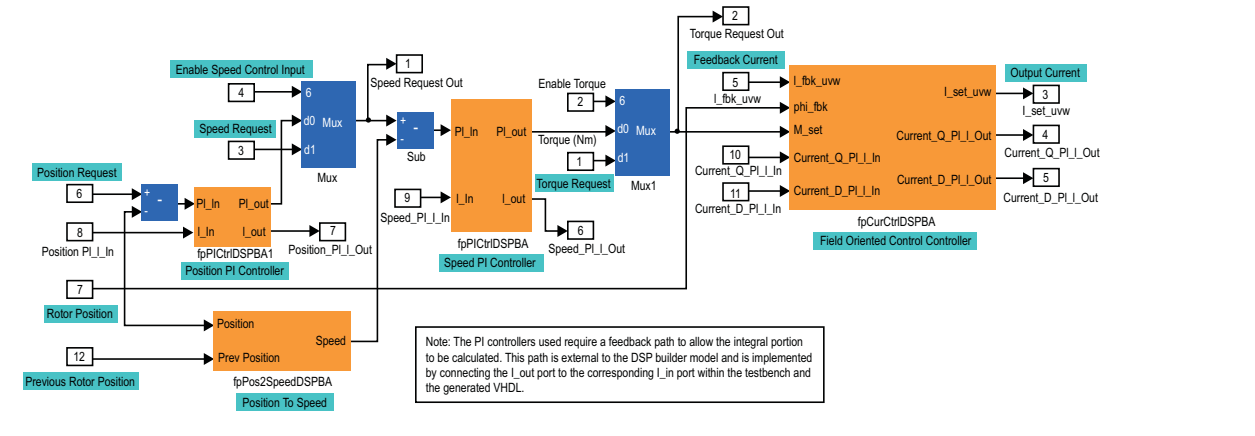
利用 Altera 的集成 DSP Builder 工具，设计人员在 Matlab/Simulink 环境中直接对 FOC 算法进行建模，构建结构图，连接表示系统的基本模块，如图 6 所示。这一实例中使用的基本模块是 ADD、SUB、MULTIPLY、CONSTANT、COSINE 和 SINE。Simulink 支持您对模型或者系统算法行为运行比特精度数学仿真。完成了算法开发后，DSP builder 针对您所选择的 Altera FPGA 器件，自动生成流水线 RTL 并进行优化。

图 6. 永磁体同步电机的 FOC 实例



这一设计实例还包括位置和速度控制环，支持对转子速度和角度的控制，如图 7 所示。典型的电机控制 IP 系统包括空间矢量 PWM、电流和扭矩控制环以及速度和位置控制环。根据 FPGA 和 CPU 资源的使用情况，设计人员在硬件和软件之间分配这些单元。

图 7. 永磁体同步电机的位置、速度和 FOC 控制器

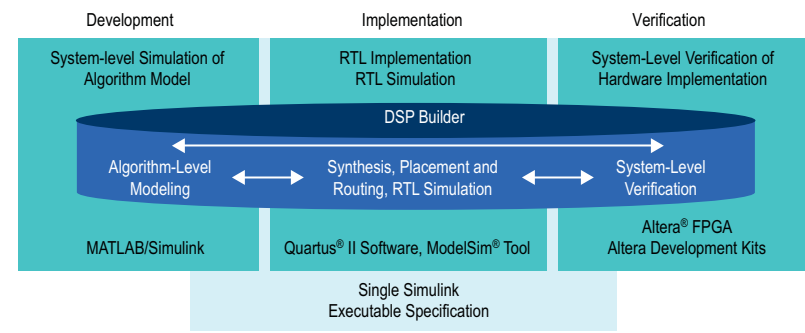


一个典型的传统设计流程包括建立电信号和电机系统模型，通过仿真开发算法，然后编写“C”程序，实现算法，在 DSP 上运行。这一设计流程具有以下缺点。

- 通常以浮点方式实现算法建模，然后转换为定点方式，在 DSP 中实现。一般以手动方式实现浮点到定点的转换，缩放和上溢保护功能非常复杂。
- 必须针对模型重新验证“C”程序。
- 提高算法运行时性能还需要进行以下步骤：
 - a. 手动优化“C”程序以提高性能。
 - b. 更新到速度更快但是更昂贵的 DSP 处理器。
 - c. 在多个 DSP 器件上并行运行算法。


设计人员可以采用 DSP Builder，优化基于 FPGA 的 DSP 系统设计。DSP Builder 优化流水线以及共享资源，生成高效的 RTL 表征。您可以组合使用现有的 MATLAB 功能和 Simulink 模块以及 Altera DSP Builder 模块和其他 IP 内核，连接系统级设计，实现 DSP 算法开发，如图 8 所示。利用 DSP Builder，系统、算法和硬件设计人员可以共享公共开发平台。

图 8. DSP Builder 系统级设计



DSP Builder 高级模块库自然支持定点或者单精度和双精度浮点类型的算法建模。设计人员可以使用比要求更高的精度，先在 Simulink 中对算法建模，然后，在工具中调整精度，以满足最终要求。DSP Builder 具有以下优势：

- 采用高级模块库，以按键 FPGA 方式实现算法。不需要手动转换步骤。
- 在硬件中运行之前，在 Simulink 中直接观察运行时延时、数据吞吐量以及算法应用。采用 Simulink 进行设计空间研究，选择最合适的实现方式。
- 在生成时优化并调整基本算子，包括 SQRT 和三角等函数，在软件中实现这些算子时，其运行较慢而且时间可变。这一方法提供了可预测的算法运行时间，显著加速了某些算子的运行。

 关于 DSP Builder 的详细信息，包括标准模块库和高级模块库的信息，请访问 [DSP 手册网站](#)。

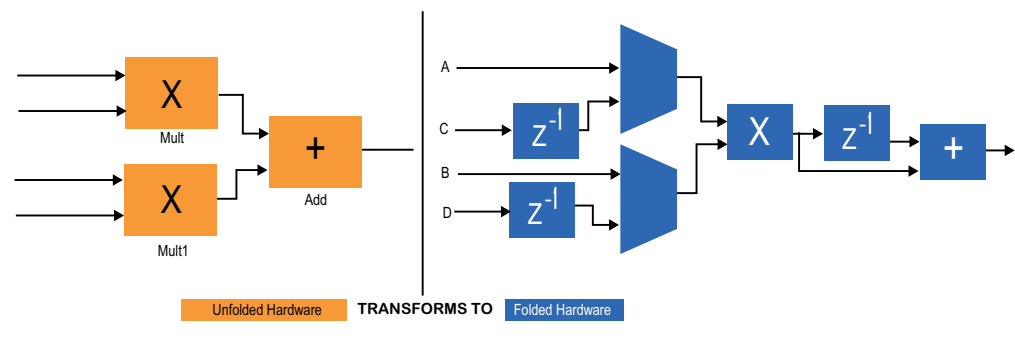
折叠理论提高了性能

折叠理论与时分复用 (TDM) 密切相关，能够大幅度提高 DSP 电机控制设计的性能。系统时钟速率高于数据速率或者采样速率时，一个硬件组件（例如，乘法器）能够有效地处理多个数据点。折叠理论支持多个通道使用乘法器和加法器等系统资源，其使用方式与 TDM 因子相似，从而节省了系统资源。

不同的数据点使用 TDM 因子来访问共享硬件资源。类似的，在具有多个并行数据源或者数据通道的系统中，设计人员不需要对每个通道或者数据源进行硬件复制，而是使用一个数据通路来处理多个数据通道。

默认情况下，DSP Builder 为基本子系统生成的硬件能够在每个时钟周期接收并处理新数据。但是，某些设计人员可能不需要每个时钟周期都进行计算。对于采样速率低于时钟速率的设计，DSP Builder 高级模块库的折叠功能可以利用这种速率差来优化使用所生成的硬件。设计人员能够以最直观的方式来实现内核算法，看起来好像没有折叠或者 TDM 因子。采用折叠理论，不需要采用信号复用和数据缓冲方法，这是在手动折叠设计中通常需要的方法。如图 9 所示，折叠功能可以减少每个周期中使用的硬件以及模块。

图 9. 折叠和没有折叠的硬件实例



对比 DSP builder 和处理器之间的性能时，可以使用下面的术语指南，以保证正确的延时和吞吐量测量：

- 一个内核处理器在 x 时钟周期延时中处理一次计算，在第一次完成之前不会开始新的计算。因此，吞吐量是每 x 时钟周期一次计算。

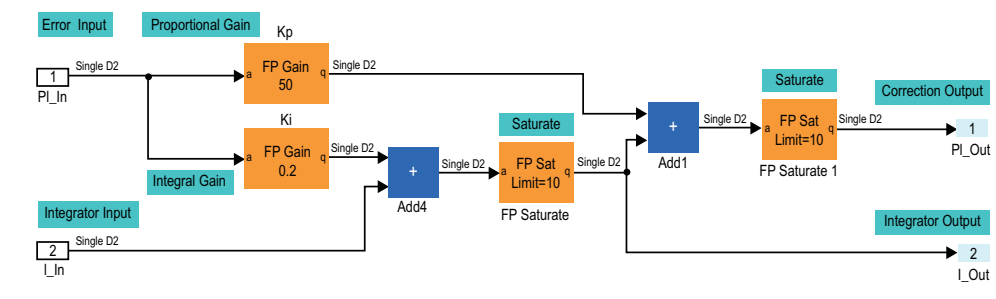
- DSP builder 系统在 y 时钟周期延时中处理一次计算，但是每一“折叠因子”时钟周期可以开始新的计算。因此，吞吐量是每一“折叠因子”时钟周期进行一次计算。

通过调整折叠因子，不需要重新设计，您就可以在吞吐量、资源使用以及生成逻辑延时上达到最优。下面介绍一个系统和测试台实例，演示折叠因子的影响。在这一部分，“折叠因子”时钟周期小于 x 时钟周期。

FOC 算法基准测试

为展示上面介绍的各种特性和功能的优点，Altera 开发了一个基准测试练习，以仿真 FOC 算法，对比标准“非折叠”和“折叠”实现的结果。FOC 算法包括 Park/Clarke 变换以及 PI 控制模块，它需要加法 / 减法、乘法、Sin、Cos 算子以及某些常数和饱和和逻辑，如图 10 和图 11 示。在一个典型的 FOC 控制器中，以 10–100 ksp/s 速率对输入进行采样，100 MHz FPGA 时钟速率很容易达到这一速率要求。在 100 ksp/s 时，每 10us 必须处理一个新采样。将这一处理延时常数保持在最小值，有利于提高控制算法的性能。

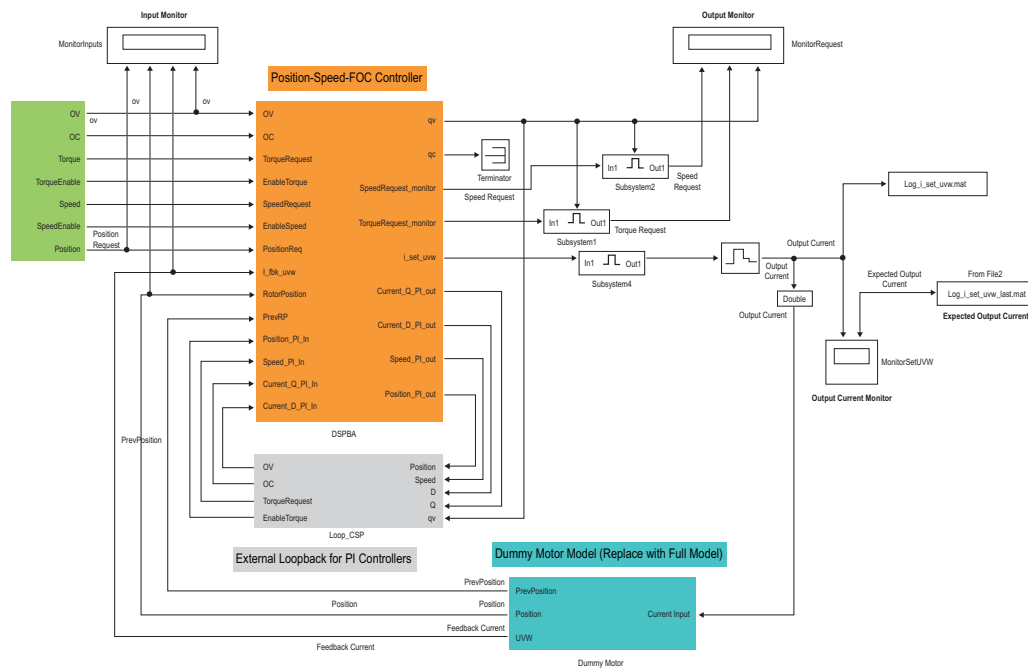
图 10. 简单比例和积分 (PI) 控制器



测试台包括图 11 所示的单元：

- 输入激励（绿色）——提供控制输入（位置申请）
- 电机模型（蓝绿色）——对 PMSM 电机进行建模
- DSP Builder 位置速度 FOC 控制器（黄色）——对控制算法进行模型
- 外部回环模型（灰色）——将积分器反馈输出从 PI 控制器连回至输入

图 11. 永磁体同步电机位置、速度和现场定位控制的控制器




采用 DSP Builder 进行设计调整


设计人员可以从 Matlab 工作空间变量中控制关键系统参数，调整设计：

- 折叠因子——扫描延时、吞吐量以及资源使用情况，找到最佳实现点。
- 定点算法精度——观察算法中不同级精度调整的影响，以提高算法性能和资源利用率。
- 算法调整——相对于设备（电机）的物理模型仿真实际算法，调整 PID 控制器、滤波器的参数，在模型级进行观察。

基准测试结果

下面详细介绍通过在 Simulink 中建模所获得的算法基准测试结果，在 Cyclone IV 器件中使用了单精度浮点和定点类型。结果表明，设计实例满足所需要的 100 MHz 时钟速率、资源利用率以及算法延时要求。

 在 Quartus II 软件中成功的进行编译后，设计人员点击 Simulink 图中的 Quartus 模块链接，可以获得正确的资源使用信息。

 一般而言，以定点方式来实现不需要浮点大动态范围的设计。但是，浮点实现避免了算法开发和调整过程中出现的算法上溢问题。

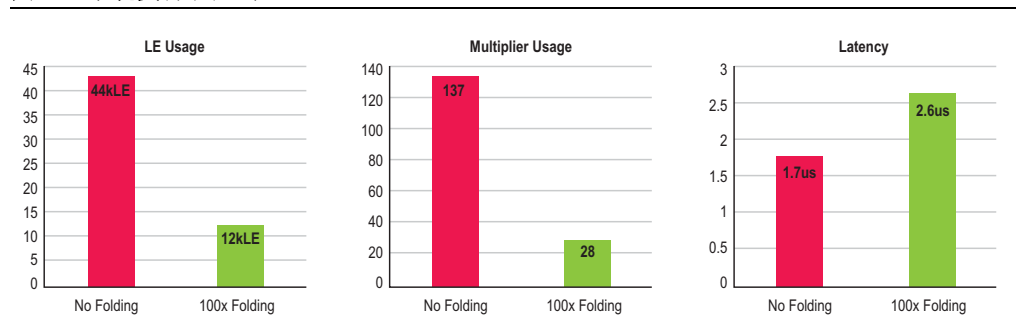
默认情况下，DSP builder 建立一个全流水线 VHDL 表征，可以在每一时钟周期中接收新输入值。然后，对比这一“非折叠”配置结果与“全折叠”配置结果。

表 1 中的结果表明，由于采用了折叠因子，显著减少了算子数量，因此，可以使用低密度 Cyclone IV 器件。而且，对于算法而言，增加的延时是可以承受的。控制环的速度是算法延时加上建立时间。在 5 μs 时，结果是每秒 200k 环（或者 pwm 输出），完全满足所需的规范要求。

表 1. 折叠因子的优点

规范	没有折叠	折叠因子 100x
加减法模块	22	1
乘法器模块	22	1
正弦模块	4	1
最大吞吐量	100 Msps	1 Msps

图 12. 系统资源和延时



DSP Builder 同时支持定点和浮点实现。表 2 对比了“全折叠”定点和浮点实现所需要的资源。使用 Matlab 工作空间变量来控制定点精度，支持设计人员进行简单的“what if”设计试验。

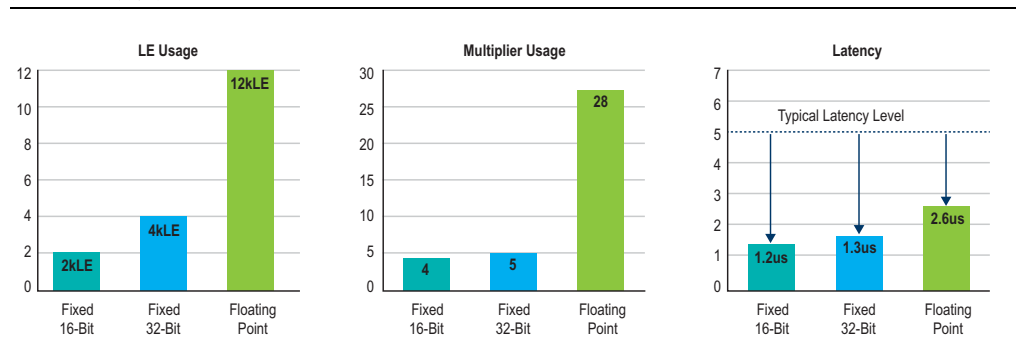
表 2. 浮点和定点对比

规范	定点 16 位	定点 32 位	浮点
LE	2K	4K	12K
18 位乘法器	4	5	28
延时	1.21 μs	1.36 μs	2.65 μs

表 2 说明:

(1) 浮点结果使用了浮点正弦。使用定点实现，占用的 LE 资源减少了 4K LE，乘法器减少了 16 个。

图 13. 系统资源和延时



结果总结

从基准测试试验中得到了以下结论：

- 使用 FOC 模型和浮点精度对 LE 资源使用的确有一些影响。但只是稍微增加了一些延时（不到 5 us），对于运算而言，延时是可以接受的。
- 精度降低到 16 位也减少了对资源的使用，这是因为有较窄的数据通路。
- 折叠因子提高了硬件资源使用率，同时满足了 1 Msps 的吞吐量。这样，可以实时处理 10 通道的 100 ksps FOC 算法。

结论

在下一代电机控制系统中，对目前的现代 MCU 和 DSP 性能要求越来越高。设计人员要求能够灵活的调整电机控制算法，以降低成本和功耗。商用 DSP 解决方案的定点和浮点能力有限，不能适应驱动系统所需其他组件的要求。

相反，Altera FPGA 支持组件集成，例如，能够管理所有操作的处理器，灵活方便的连接定制子系统，优化设计流程，从而简化了复杂并行电机控制环和算法。电机系统结合了各种快速控制环、定时输出脉冲频率以及所需的多传感器接口和滤波功能。Altera FPGA 内置了并行处理功能，提供高性能精度可调 DSP 模块，从而解决了电机控制系统遇到的难题，为其提供最优解决方案。

除了这些内在的 FPGA 优势，Altera 还提供最佳设计方法。在建模时使用 The Mathworks Simulink/Matlab 工具，Altera 的 DSP Builder 对电机算法进行优化，Qsys 或者 SOPC Builder 实现系统集成，Quartus II 软件用于设计综合和滤波，实现了全面的集成设计方法，从而能够处理最复杂的驱动系统。

详细信息

- Altera 工业网站
<http://www.altera.com/end-markets/industrial/ind-index.html>
- 白皮书：降低工业应用的总体拥有成本
<http://www.altera.com/literature/wp/wp-01122-tco-industrial.pdf>
- 白皮书：灵活的工业以太网解决方案
<http://www.altera.com/literature/wp/wp-01037.pdf>
- 白皮书：使用经过 TuV 认证的 FPGA 开发功能安全系统
<http://www.altera.com/literature/wp/wp-01123-functional-safety.pdf>
- 网播：降低工业设计的总体拥有成本
<http://www.altera.com/education/webcasts/all/wc-2010-lower-tco-for-industrial-designs.html>
- 视频：迅速适应变化的以太网协议的 3 种方法
<http://www.altera.com/education/webcasts/videos/videos-adapt-to-changing-ethernet-protocols.html>
- 更多的工业视频和网播：
www.altera.com/servlets/webcasts/search?endmarket=industrial

致谢

- Kevin Smith，技术部高级组员，Altera 公司。

- Wil Florentino, 高级技术市场经理, Altera 公司。
- Jason Chiang, 高级技术市场经理, Altera 公司。
- Stefano J. Zammattio, 产品经理, Altera 公司。

Altera 简介

Altera 作为可编程逻辑的创始企业, 为系统设计人员提供可信赖的创新技术, 帮助他们快速高效地实现创新, 突出产品优势, 赢得市场竞争。通过我们的代工线商业模式, 我们能够集中精力开发技术先进的高级 FPGA、CPLD 和 HardCopy® ASIC。

使用 Altera 工业级 FPGA 作为协处理器或者 SoC 解决方案, 提高了工业应用的灵活性。作为多种工业产品一个高度集成的平台, Altera FPGA 有效的缩短了开发时间, 降低了风险。Altera FPGA 具有以下优势:

- 通过硬核 IP 模块、嵌入式处理器、收发器和其他功能进行设计集成, 增强了应用功能, 降低了总成本。
- 可重新编程功能, 即使在现场, 也可以支持不断发展的工业以太网协议, 满足多变的设计需求。
- 通过嵌入式处理器、定制指令和 DSP 模块来调整性能。
- 过时保护, 以及向未来 FPGA 系列的移植途径, 这些都能够支持工业设备较长的生命周期。
- 熟悉的工具, 使用熟悉的、功能强大的集成工具, 简化设计和软件开发、IP 集成以及调试。

文档修订历史

表 3 列出了本文档的修订历史。

表 3. 文档修订历史

日期	版本	进行的修改
2011 年 7 月	1.1	■ 对文本进行了一些编辑。
2011 年 4 月	1.0	初次发布。