

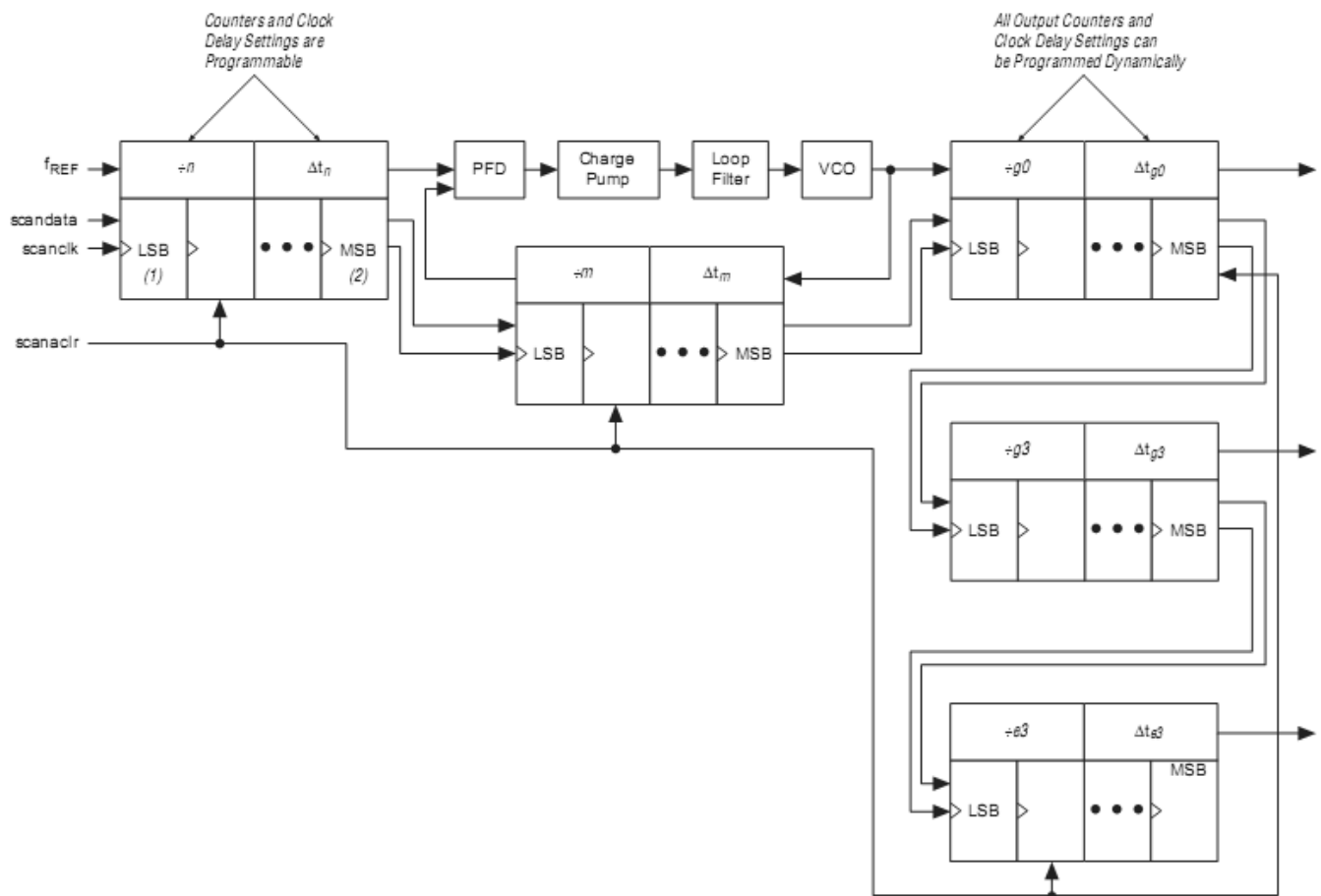
Пример проектирования: Динамическая реконфигурация PLL

Простота в использовании средства исходники и пробники в системе может быть чрезвычайно полезно в создании виртуальной лицевой панели на этапе прототипирования вашего проекта. Относительно простые проекты большой функциональности могут быть созданы в кратчайшие сроки. Следующий пример реконфигурации PLL показывает, как используется средство исходники и пробники в системе в качестве графической оболочки для динамической реконфигурации Stratix PLL.

Stratix PLL позволяют вам динамически обновлять коэффициенты PLL в реальном времени. Каждая усовершенствованная PLL в чипе Stratix содержит цепь регистров, которая позволяет вам модифицировать масштабируемые счётчики (значения m и n), выходные делители и счётчики задержки. Дополнительно, мегафункция *altpll_reconfig* предоставляет вам простой интерфейс доступа к цепям регистров счётчиков. Мегафункция *altpll_reconfig* имеет кэш, содержащий все модифицированные параметры PLL. После того, как вы обновите все параметры PLL в КЭШе, мегафункция *altpll_reconfig* управляет цепью регистров PLL для обновления PLL модифицированными параметрами. На рисунке 17-6 показана усовершенствованная Stratix PLL с реконфигурируемыми коэффициентами.

Чипы Stratix II и Stratix III также позволяют вам динамически реконфигурировать параметры PLL. За дополнительной информацией о динамической реконфигурации обратитесь к "AN 282: Выполнение реконфигурации PLL в чипах Stratix & Stratix GX" или к "AN 367: Выполнение реконфигурации PLL в чипах Stratix II".

Figure 17–6. Stratix-Enhanced PLL with Reconfigurable Coefficients



В следующем примере проектирования используется элемент *altsource_probe* для обновления параметров PLL в кэше мегафункции *altpll_reconfig*. Мегафункция *altpll_reconfig* подключена к усовершенствованной PLL в Stratix FPGA, чтобы управлять цепью регистров, содержащей коэффициенты реконфигурации PLL. В этом примере проектирования использован скрипт Tcl/Tk для генерации графической оболочки, где вы будете вводить новые значения m и n для усовершенствованной PLL. Скрипт Tcl забирает значения m и n из графической оболочки, сдвигает значения в элементах *altsource_probe* для обновления значений в кэше мегафункции *altpll_reconfig* и утверждает сигнал реконфигурации в мегафункции *altpll_reconfig*.

Сигнал реконфигурации мегафункции `altpll_reconfig` запускает транзакцию в цепи регистров для обновления всех коэффициентов реконфигурации PLL. Блок-схема примера проекта приведена на рисунке 17-7. Графическая оболочка Tk показана на рисунке 17-8.

Figure 17-7. Block Diagram of Dynamic PLL Reconfiguration Design Example

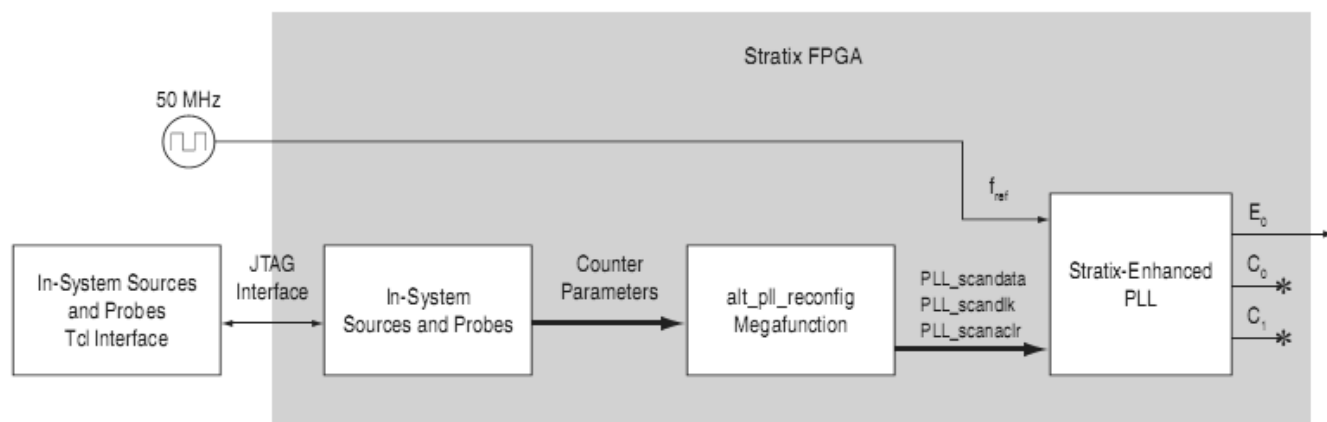
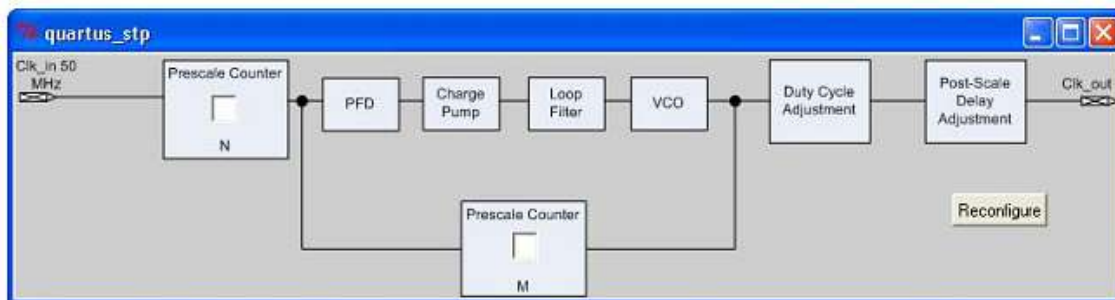


Figure 17-8. Interactive PLL Reconfiguration GUI Created with Tk and In-System Sources and Probes Tcl Package



Этот пример проектирования был создан с использованием набора инструментальных средств разработки Nios® II, версии Stratix.

Файл `sourceprobe_DE_dynamic_pll.zip` содержит все необходимые файлы для запуска этого примера:

- `Readme.txt` – текстовый файл, в котором описываются файлы, включённые в пример проектирования, и содержатся инструкции по запуску Tk GUI, показанную на рисунке 17-8.
- `Interactive_Reconfig.qar` – архив проекта Quartus II для этого примера проектирования.

Вы можете скачать файл `sourceprobe_DE_dynamic_pll.zip`, ассоциированный с этой главой.

Заключение

Исходники и пробники в системе позволяют вам задавать входные сигналы и получать отклики от их воздействия в проекте в реальном времени. С помощью простого интуитивного интерфейса, вы можете получать виртуальные входы в ваш проект в реальном времени без использования внешних устройств. Когда используется совместно с SignalTap II, использование исходников и пробников в системе даёт вам больший контроль над сигналами в вашем проекте, что сокращает цикл верификации. Также эта возможность создавать виртуальные входы в вашем проекте, позволяет вам создавать простые, но мощные приложения для взаимодействия с вашим проектом.