

Общий вид временного анализатора

В этой главе представлена общая концепция временного анализатора Quartus II TimeQuest. Понимание этой концепции позволит вам получать максимальные результаты от использования всех средств временного анализа, доступных во временном анализаторе Quartus II TimeQuest. Во время анализа вашего проекта, временной анализатор Quartus II TimeQuest работает по алгоритму, показанному на рисунке 7-2. В таблице 7-2 представлен список самых распространенных команд, используемых в каждом шаге.

Figure 7-2. The Quartus II TimeQuest Timing Analyzer Flow

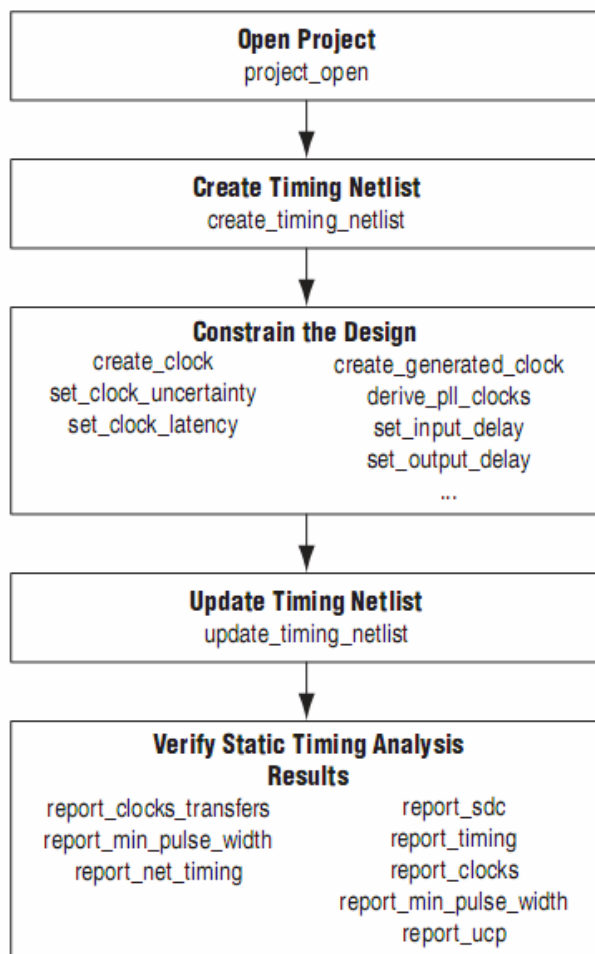


Таблица 7-2. Термины временного анализатора Quartus II TimeQuest

Термин	Определение
Nodes (Узлы)	Основная базовая временная единица списка соединений. Используется для представления портов, выводов и регистров.
Keepers(Защелки)	Порты или регистры (1)
Cells (Ячейки)	Матрица переключений (LUT), регистры, блоки цифровых сигнальных процессоров (DSP), TriMatrix память, IOE и т.п. (2)
Pins (Выводы)	Входы и выходы ячеек
Nets (шины)	Соединения между выводами
Ports (порты)	В головном модуле – входы или выходы, например, выводы чипа
Clocks (такты)	Абстрактные объекты снаружи чипа

Примечания к таблице 7-2:

- (1) Выводы могут косвенно относиться к защелкам. Например, когда значение в поле ограничений -from является тактовым выводом к определенной памяти. В этом случае тактовый вывод относится к группе регистров.
- (2) Для чипов Stratix® и других ранних семейств чипов, LUT и регистры находились в логическом элементе и являлись ячейкой в этом семействе чипов.

Временному анализатору Quartus II TimeQuest требуется временной список соединений прежде, чем он будет выполнять временной анализ какого-нибудь проекта. Например, для проекта, показанного на рисунке 7-3, временной анализатор генерирует список соединений, эквивалентный показанному на рисунке 7-4.

Figure 7-3. Sample Design

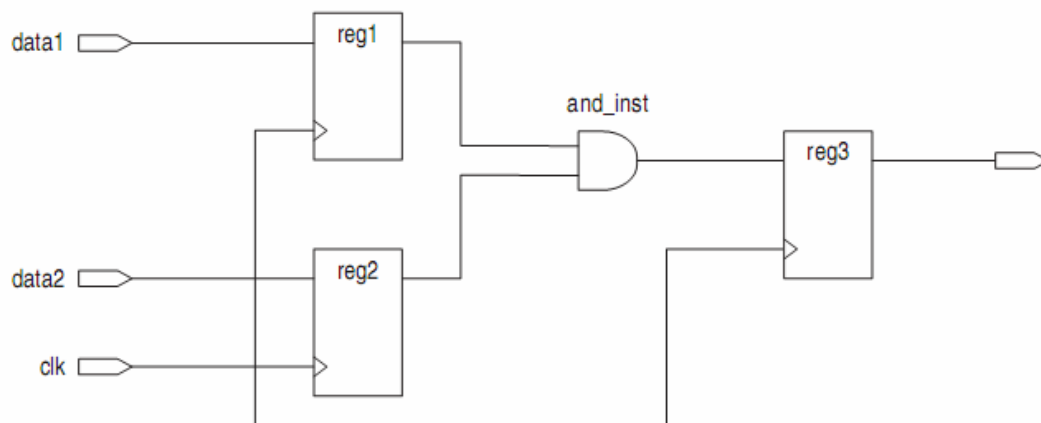
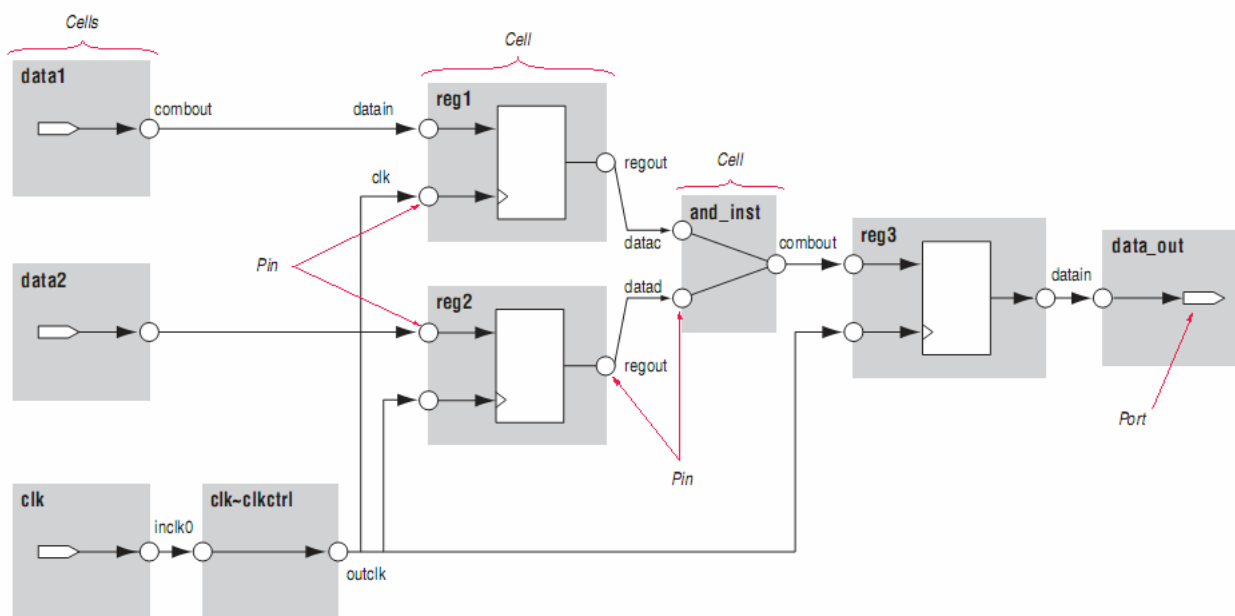


Figure 7-4. The Quartus II TimeQuest Timing Analyzer Timing Netlist



На рисунке 7-4 показаны различные ячейки, выводы, шины и порты. Перечислим названия ячеек:

- reg1
- reg2
- and_inst

Перечислим названия выводов:

- data1|combout
- reg1|regout
- and_inst|combout

Перечислим названия шин:

- data1~combout
- reg1
- and_inst

Перечислим названия портов:

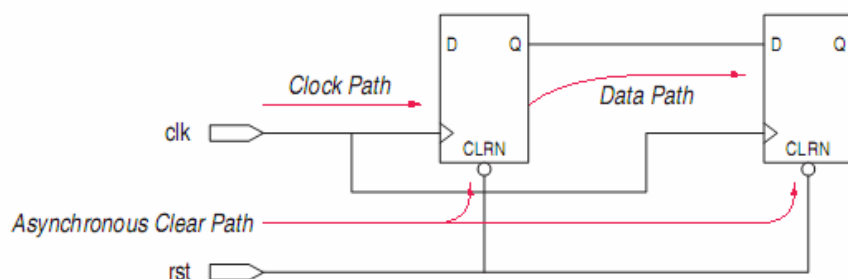
- data1, clk
- data_out

Пути соединяют два узла проекта, такие как выход одного регистра с входом другого регистра. Временные пути играют значительную роль во временном анализе. Понимание видов временных задержек – это главное для временных ограничений и оптимизации. В следующем списке приведены основные анализируемые пути, которые описываются в этом разделе:

- Граничные пути — соединения: между портами и выводами, между выводами, между входами и портами.
- Тактовые пути — от границ портов чипа или внутренне сгенерированного тактового вывода до тактового вывода регистра.
- Пути данных — от границ порта или выходного вывода данных последовательного элемента до порта или входного вывода данных другого последовательного элемента.
- Асинхронные пути — от границ порта или последовательного элемента на вывод асинхронной установки или сброса последовательного элемента.

На рисунке 7-5 показаны некоторые из перечисленных видов анализа путей.

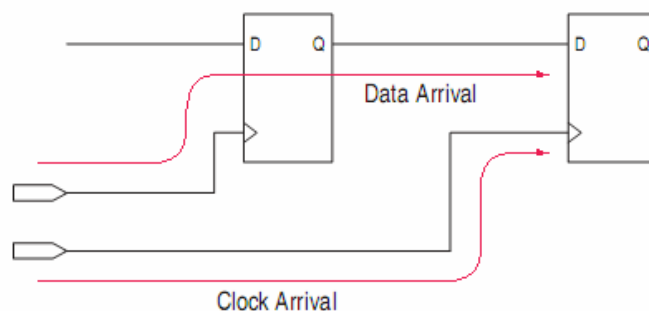
Figure 7-5. Path Types



После того, как временной анализатор Quartus II TimeQuest определит тип путей, он сообщает о времени поступления данных и тактов для действительных путей между регистрами. Временной анализатор Quartus II TimeQuest подсчитывает время поступления данных путем добавления задержки от источника тактов до тактового вывода исходного регистра, микро такт на выход (μtCO) исходного регистра, и задержку от вывода Q исходного регистра до вывода D регистра назначения, при этом μtCO – это внутренняя задержка распространения в регистре для регистров FPGA.

Временной анализатор Quartus II TimeQuest подсчитывает время поступления тактов путем добавления задержки от источника тактов до тактового вывода регистра назначения. На рисунке 7-6 показаны время поступления тактов и данных. Временной анализатор Quartus II TimeQuest подсчитывает требуемое время, рассчитывая время прихода тактов и микро время установки (μtSU) регистра назначения, при этом μtSU – это внутренняя установка для регистров FPGA.

Figure 7-6. Data Arrival and Clock Arrival

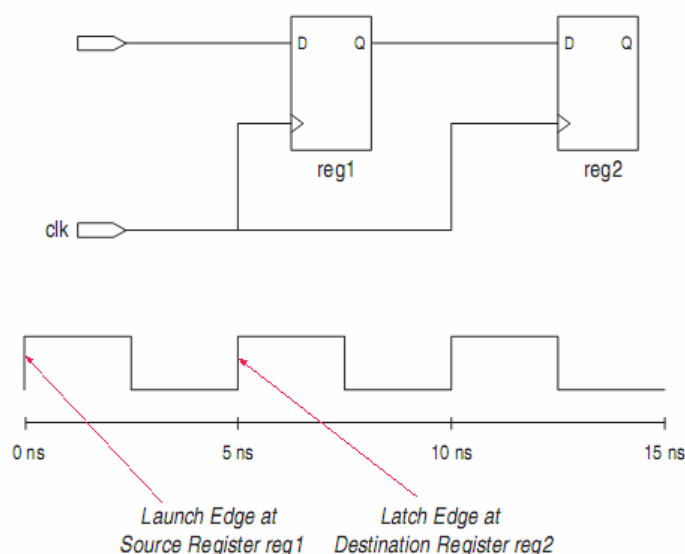


Дополнительно, для идентификации различных путей в проекте, временной анализатор Quartus II TimeQuest анализирует тактовые характеристики для вычисления наихудших условий между двумя различными регистрами в одном пути от регистра к регистру. Вы можете ограничить все такты в своем проекте, прежде чем выполнять этот анализ.

Фронт запуска – это активный тактовый фронт, который перемещает данные на выход последовательного элемента, используется в качестве исходного для передачи данных. Фронт защелки – это активный тактовый фронт, который фиксирует данные на входе данных последовательного элемента, используется в качестве целевого для передачи данных.

На рисунке 7-7 показаны: одноканальная система, которая использует последовательные фронты тактов для перемещения и фиксации данных, путь от регистра к регистру и соответствующие фронты запуска и фиксации на временной диаграмме. В этом примере, фронт запуска передает данные на выход от регистра reg1 на 0 ns, а регистр reg2 защелкивает данные по фронту фиксации на 5 ns.

Figure 7-7. Launch Edge and Latch Edge



Временной анализатор Quartus II TimeQuest проверяет достоверность величины установки и задержки тактов, требуемых для фронтов запуска и защелки.

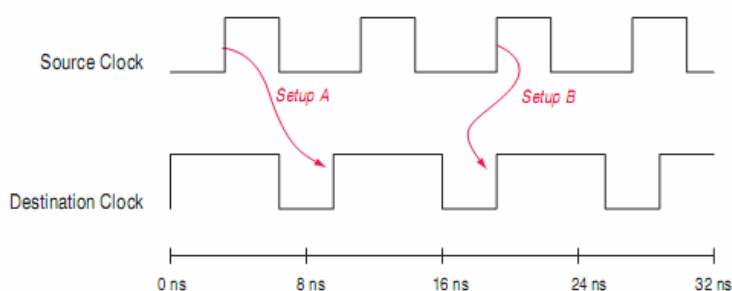
Анализ тактов

Комплексный статический анализ содержит анализ путей: из одного регистра в другой, I/O и асинхронного сброса. Временной анализатор Quartus II TimeQuest использует требуемое время данных, время поступления данных и время поступления тактов для верификации характеристик схемы и определяет возможные временные нарушения. Временной анализатор Quartus II TimeQuest определяет временные взаимосвязи, которые должны обеспечивать правильное функционирование проекта, и контролирует время поступления в сравнении с требуемым временем для верификации времени.

Контроль времени установки

Для выполнения контроля времени установки тактов, временной анализатор Quartus II TimeQuest определяет соотношение установки путём анализа каждого фронта запуска и защёлки для каждого пути от регистра к регистру. Для каждого фронта защёлки в регистре назначения, временной анализатор Quartus II TimeQuest использует ближайший предыдущий тактовый фронт в исходном регистре в качестве фронта запуска. На рисунке 7-8, два соответствия установки определены и помечены: установка А и установка В. Для фронта защёлки в 10 нс, ближайший такт, работающий как фронт запуска, является в 3 нс и помечен как А. Для фронта защёлки в 20 нс, ближайший такт, работающий как фронт запуска, является в 19 нс и помечен как В.

Figure 7-8. Setup Check



Временной анализатор Quartus II TimeQuest выдаёт отчёт о контроле установки тактов в виде значений временного резерва (slack). Временной резерв – это запас, по которому определяется соответствие или не соответствие временным ограничениям. Положительный временной резерв показывает, что запас соответствует ограничениям; отрицательный – не соответствует. Временной анализатор Quartus II TimeQuest определяет временной резерв установки тактов, как показано в равенстве 7-1, для внутренних путей от регистра к регистру.

Equation 7-1.

$$\begin{aligned}\text{Clock Setup Slack} &= \text{Data Required Time} - \text{Data Arrival Time} \\ \text{Data Arrival Time} &= \text{Launch Edge} + \text{Clock Network Delay to Source Register} + \\ &\quad \mu t_{CO} + \text{Register-to-Register Delay} \\ \text{Data Required} &= \text{Clock Arrival Time} - \mu t_{SU} - \text{Setup Uncertainty} \\ \text{Clock Arrival Time} &= \text{Latch Edge} + \text{Clock Network Delay to Destination Register}\end{aligned}$$

Если путь данных от входного порта до внутреннего регистра, то временной анализатор Quartus II TimeQuest использует равенство 7-2, для подсчёта временного резерва установки.

Equation 7-2.

$$\begin{aligned}\text{Clock Setup Slack Time} &= \text{Data Required Time} - \text{Data Arrival Time} \\ \text{Data Arrival Time} &= \text{Launch Edge} + \text{Clock Network Delay} + \\ &\quad \text{Input Maximum Delay of Pin} + \text{Pin-to-Register Delay} \\ \text{Data Required Time} &= \text{Latch Edge} + \text{Clock Network Delay to Destination Register} - \mu t_{SU}\end{aligned}$$

Если путь данных от внутреннего регистра до выходного порта, то временной анализатор Quartus II TimeQuest использует равенство 7-3, для подсчёта временного резерва установки.

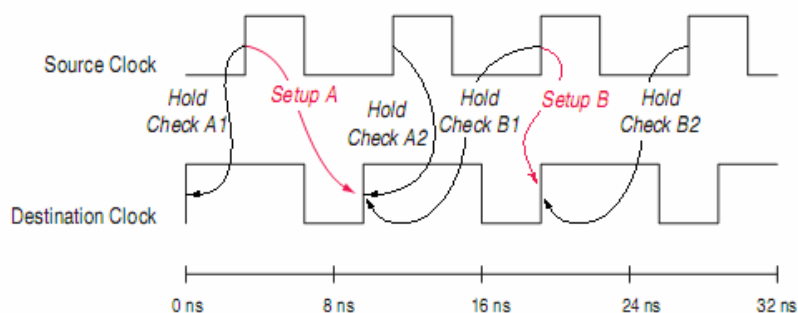
Equation 7-3.

$$\begin{aligned}\text{Clock Setup Slack Time} &= \text{Data Required Time} - \text{Data Arrival Time} \\ \text{Data Arrival Time} &= \text{Launch Edge} + \text{Clock Network Delay to Source Register} + \\ &\quad \mu t_{CO} + \text{Register-to-Pin Delay} \\ \text{Data Required Time} &= \text{Latch Edge} + \text{Clock Network Delay} - \text{Output Maximum Delay of Pin}\end{aligned}$$

Контроль удержания тактов

Для выполнения контроля времени удержания тактов, временной анализатор Quartus II TimeQuest определяет соотношение удержания для каждого возможного соотношения установки, которое существует для всех путей от исходных до регистров назначения. Временной анализатор Quartus II TimeQuest контролирует все соседние тактовые фронты соотношения установки для определения соотношения удержания. Временной анализатор Quartus II TimeQuest выполняет два контроля удержания для каждого соотношения установки. Во время первого контроля удержания определяется, что данные, запущенные по текущему фронту запуска, не захватывают предыдущий фронт защёлки. Во время второго контроля удержания определяется, что данные, запущенные по следующему фронту запуска, не захватывают текущий фронт защёлки. На рисунке 7-9 показаны два соотношения установки, обозначенные А и В. Первый контроль удержания обозначен как А1 и В1 соответственно. Второй контроль удержания обозначен как А2 и В2 соответственно.

Figure 7–9. Hold Checks



Из возможных соотношений удержания, временной анализатор Quartus II TimeQuest выбирает соотношений удержания самое ограничивающее. Соотношение удержания с большей разницей между фронтами защёлки и запуска (это означает, защёлка – запуск, а не абсолютное значение защёлки и запуска) будет выбрано, потому что это определяет минимальную допустимую задержку для пути от регистра к регистру. На рисунке 7-9, выбирается соотношение удержания A2.

Временной анализатор Quartus II TimeQuest определяет величину временного резерва удержания в равенстве 7-4.

Equation 7–4.

$$\text{Clock Hold Slack} = \text{Data Arrival Time} - \text{Data Required Time}$$

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay to Source Register} + \mu t_{CO} + \text{Register-to-Register Delay}$$

$$\text{Data Required Time} = \text{Clock Arrival Time} + \mu t_H + \text{Hold Uncertainty}$$

$$\text{Clock Arrival Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register}$$

Если путь данных от входного порта до внутреннего регистра, то временной анализатор Quartus II TimeQuest использует равенство 7-5, для подсчёта временного резерва удержания.

Equation 7–5.

$$\text{Clock Hold Slack Time} = \text{Data Arrival Time} - \text{Data Required Time}$$

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay} + \text{Input Minimum Delay of Pin} + \text{Pin-to-Register Delay}$$

$$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register} + \mu t_H$$

Если путь данных от внутреннего регистра до выходного порта, то временной анализатор Quartus II TimeQuest использует равенство 7-6, для подсчёта временного резерва удержания.

Equation 7–6.

$$\text{Clock Hold Slack Time} = \text{Data Arrival Time} - \text{Data Required Time}$$

$$\text{Data Arrival Time} = \text{Latch Edge} + \text{Clock Network Delay to Source Register} + \mu t_{CO} + \text{Register-to-Pin Delay}$$

$$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay} - \text{Output Minimum Delay of Pin}$$

Восстановление и удаление

Время восстановления – это минимальная величина времени снятия асинхронного контролирующего сигнала; например, сброс и предустановка, они должны быть постоянными перед следующим активным фронтом импульса. Подсчёт временного резерва восстановления схож с подсчётом временного резерва установки тактов, но применяется для асинхронных контрольных сигналов. Если регистрируется асинхронный сигнал контроля, то временной анализатор Quartus II TimeQuest использует равенство 7-7 для подсчёта временного резерва восстановления.

Equation 7-7.

$$\text{Recovery Slack Time} = \text{Data Required Time} - \text{Data Arrival Time}$$

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay to Source Register} + \\ \mu t_{CO} + \text{Register-to-Register Delay}$$

$$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register} - \mu t_{SU}$$

Если асинхронный контролирующий сигнал не регистрируется, то временной анализатор Quartus II TimeQuest использует равенство 7-8 для подсчёта временного резерва восстановления.

Equation 7-8.

$$\text{Recovery Slack Time} = \text{Data Required Time} - \text{Data Arrival Time}$$

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay} + \text{Maximum Input Delay} + \\ \text{Port-to-Register Delay}$$

$$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register Delay} - \mu t_{SU}$$

Если сигнал асинхронного сброса идёт от порта (I/O чипа), вам необходимо сделать назначение **Максимальная Задержка на Входе** для порта асинхронного сброса для временного анализатора Quartus II TimeQuest, чтобы выполнить анализ восстановления для этого пути.

Время удаления – это минимальная величина времени снятия асинхронного контролирующего сигнала, которое должно быть постоянно после активного тактового фронта. Временной анализатор Quartus II TimeQuest подсчитывает временной резерв схоже с подсчётом временного резерва удержания тактов, только применительно для асинхронного контролирующего сигнала. Если асинхронный контролирующий сигнал регистрируется, то временной анализатор Quartus II TimeQuest использует равенство 7-9 для подсчёта временного резерва удаления.

Equation 7-9.

$$\text{Removal Slack Time} = \text{Data Arrival Time} - \text{Data Required Time}$$

$$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay to Source Register} + \\ \mu t_{CO} \text{ of Source Register} + \text{Register-to-Register Delay}$$

$$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register} + \mu t_H$$

Если асинхронный контролирующий сигнал не регистрируется, то временной анализатор Quartus II TimeQuest использует равенство 7-10 для подсчёта временного резерва удаления.

Equation 7-10.

$\text{Removal Slack Time} = \text{Data Arrival Time} - \text{Data Required Time}$

$\text{Data Arrival Time} = \text{Launch Edge} + \text{Clock Network Delay} + \text{Input Minimum Delay of Pin} +$
 $\text{Minimum Pin-to-Register Delay}$

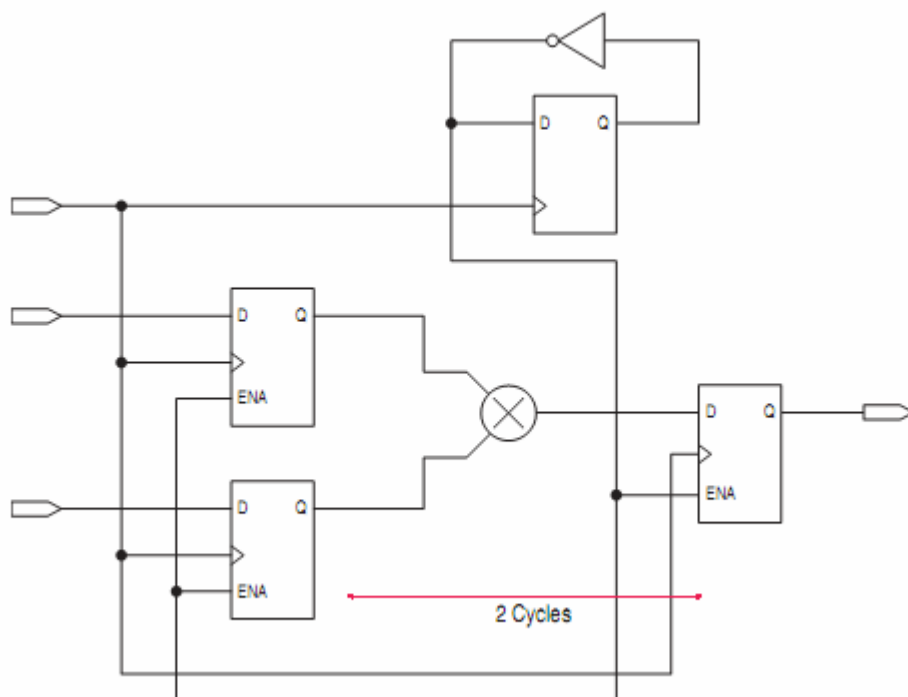
$\text{Data Required Time} = \text{Latch Edge} + \text{Clock Network Delay to Destination Register} + \mu t_d$

Если сигнал асинхронного сброса идёт от порта (I/O чипа), вам необходимо сделать назначение **Минимальная Задержка на Входе** для порта асинхронного сброса для временного анализатора Quartus II TimeQuest, чтобы выполнить анализ удаления для этого пути.

Мультицикловые пути

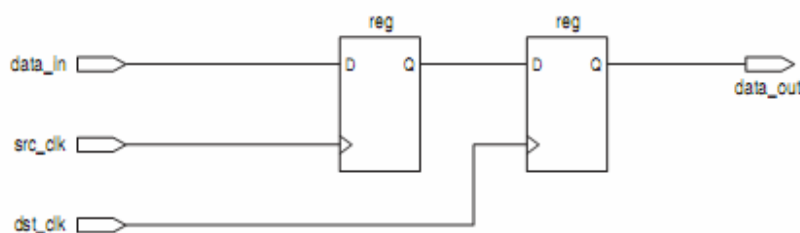
Мультицикловые пути – это все пути данных, которым требуется более одного тактового цикла для защёлкивания данных в регистре назначений. Например, регистр используется для сбора данных на каждый второй или третий фронт спада тактового импульса. На рисунке 7-10 показан пример мультициклового пути между умножителем входных регистров и выходного регистра назначения, который защёлкивает данные по каждому другому тактовому фронту.

Figure 7-10. Example Diagram of a Multicycle Path



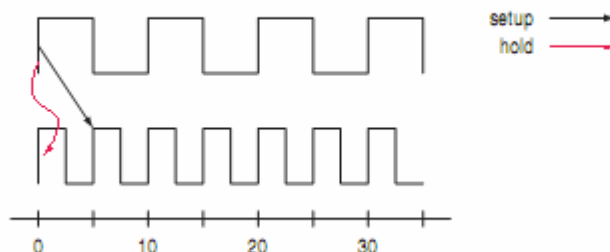
На рисунке 7-11 показан путь от регистра к регистру, на котором исходный такт, `src_clk`, имеет период 10 нс, а такт назначения, `dst_clk`, имеет период 5 нс.

Figure 7-11. Register-to-Register Path



На рисунке 7-12 показаны соответствующие временные диаграммы для исходного такта и такта назначения, а также соотношения установки и удержания. Соотношение установки по умолчанию – 5 нс; соотношение удержания по умолчанию – 0 нс.

Figure 7-12. Default Setup and Hold Timing Diagram



Соотношения по умолчанию – установки и удержания – могут быть изменены с помощью команды: `set_multicycle_path` – для подгонки системных требований.

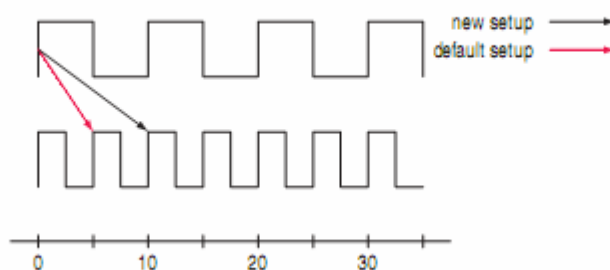
В таблице 7-3 показаны команды, использующиеся для модификации времени фронта запуска или защёлкивания, который будет использовать временной анализатор Quartus II TimeQuest для определения соотношений установки и удержания.

Таблица 7-3. Команды модификации времени фронта

Команда	Назначение модификации
<code>set_multicycle_path -setup -end</code>	Время фронта защёлки соотношения установки
<code>set_multicycle_path -setup -start</code>	Время фронта запуска соотношения установки
<code>set_multicycle_path -hold -end</code>	Время фронта защёлки соотношения удержания
<code>set_multicycle_path -hold -start</code>	Время фронта запуска соотношения удержания

На рисунке 7-13 показаны временные диаграммы после того как применена установка в два мультицикла. Команда изменила фронт защёлкивания до 10 нс из 5 нс по умолчанию.

Figure 7-13. Modified Setup Diagram



Метастабильность

Проблемы метастабильности появляются тогда, когда сигнал переходит между схемами в несвязанных или асинхронных тактовых областях, поэтому разработчик не может гарантировать, что сигнал будет соблюдать требования по установке и удержанию времени. Чтобы минимизировать сбои метастабильности, разработчики схем обычно используют последовательность из регистров (цепочка регистров синхронизации или синхронизатор) в назначенной тактовой области для ресинхронизации сигналов данных в новой тактовой области.

Среднее время перед сбоем (MTBF) – это оценка среднего времени сбоев между модулями, приводящего к метастабильности.

Временной анализатор TimeQuest анализирует запас прочности проекта от метастабильности и может подсчитать MTBF для цепочек регистров синхронизации в проекте. MTBF каждого проекта – это ограничения, основанные на предполагаемой цепочке синхронизации.

В дополнении к отчету о найденной в проекте цепочке регистров синхронизации, программа Quartus II также защищает эти регистры от оптимизации, которая негативно скажется на MTBF, так например, дублирование регистров и логическое восстановление синхронизации. Программа Quartus II может также оптимизировать MTBF для вашего проекта, если MTBF слишком мало.

Обратитесь к “report_metastability” на странице 7-57 за информацией о том, каким образом разрешать анализ метастабильности и об отчете метастабильности во временном анализаторе TimeQuest.

За дополнительной информацией о метастабильности, её эффектах в FPGA и о том, как подсчитывается MTBF, обратитесь к белой книге "Понимание метастабильности в FPGA". За дополнительной информацией об анализе метастабильности, отчётах и средствах оптимизации в программе Quartus II, обратитесь к тому 1 Настольной книги Quartus II, глава "Управление метастабильностью с помощью программы Quartus II".

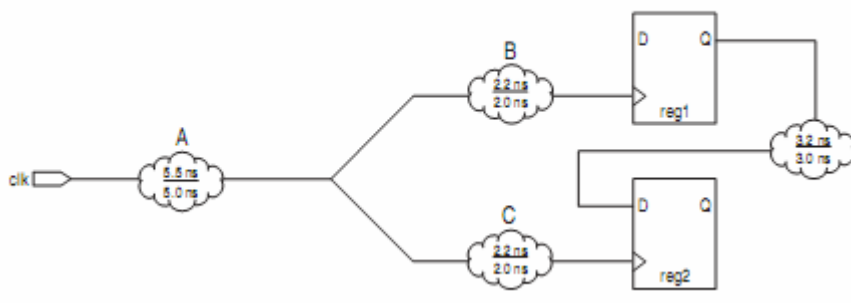
Пессимизм общих тактовых путей

Пессимизм общих тактовых путей (ССРР) удаляет отчётность для разброса минимальной и максимальной задержки, ассоциированной с общими тактовыми путями во время статичного временного анализа. ССРР удаляет отчётность для этого разброса, путём добавления разности между максимальной и минимальной задержкой общих тактовых путей, соответствующего равенства временного запаса.

Минимальный и максимальный разброс значений должен появляться, когда два различных значения задержки используются для одного тактового пути. Например, в простом анализе установки, максимальная задержка такового пути до исходного регистра используется для определения времени поступления данных. Минимальная задержка пути до регистра назначений используется для определения требуемого времени для данных. Тем не менее, если тактовый путь до исходного регистра и до регистра назначений используют общий тактовый путь, то при анализе используются обои максимальная и минимальная задержки для моделирования общего тактового пути. Эти результаты в чрезмерно пессимистичном анализе не могут быть использованы для модели с общим тактовым путём, поскольку имеют для него два значения задержки: максимальная и минимальная.

На рисунке 7-14 показан типичный путь от регистра к регистру с указанием значений максимальной и минимальной задержки.

Figure 7-14. Common Clock Path



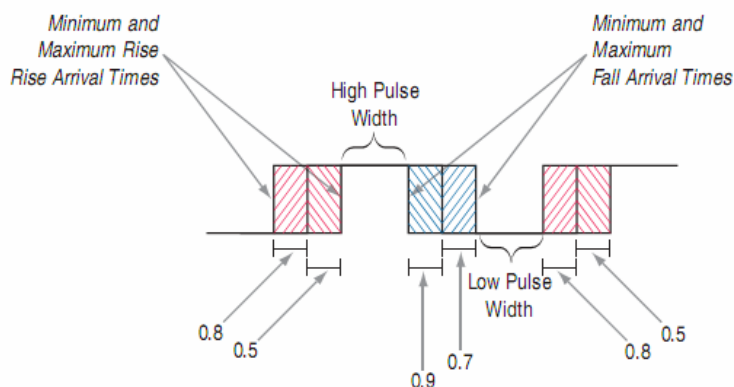
Сегмент А – это общий тактовый путь между reg1 и reg2. Минимальная задержка – 5.0 нс; максимальная задержка – 5.5 нс. Разница между максимальной и минимальной задержками

равна значению удаления ССРР; в этом случае ССРР равно 0.5 нс. Значение удаления ССРР добавляется к соответствующему равенству временного запаса. Поэтому, если временной запас установки от регистра к регистру на рисунке 7-14 равен 0.7 нс без удаления ССРР, то со значением удаления ССРР временной запас становится 1.2 нс.

ССРР также используется, когда определяется минимальная величина импульса для регистра. Тактовый сигнал должен обеспечивать минимальную требуемую величину импульса, чтобы он распознавался регистром. Минимальное время импульса "1" для регистров, защёлкивающихся по переднему фронту. Минимальное время импульса "0" для регистров, защёлкивающихся по заднему фронту.

Тактовые импульсы, в которых нарушена минимальная длительность импульса для регистра, мешают защёлкиваться данным на входе данных регистра. Чтобы подсчитать временной резерв для минимальной длительности импульса, требуется выделить время минимальной длительности импульса из текущей минимальной длительности импульса. Текущая минимальная длительность импульса определяется условиями, определённым для тактов, поступающих на тактовых порт регистра. Требуемое минимальное время длительности импульса определяется по времени: максимального и минимального нарастания фронта, максимального и минимального спада фронта. На рисунке 7-15 показаны диаграммы минимальной длительности для импульсов "0" и "1".

Figure 7-15. Required Minimum Pulse Width



При помощи ССРР, минимальный временной резерв длительности импульса может быть увеличен за счёт меньшего значения максимального фронта нарастания минус минимальный фронт нарастания, или за счёт максимального фронта спада минус минимальный фронт спада. Для примера на рисунке 7-15, временной резерв может быть увеличен на 0,2 нс, поскольку это – наименьшее значение между 0,3 нс (0,8 нс – 0,5 нс) и 0,2 нс (0,9 нс – 0,7 нс). Обратитесь к "report_min_pulse_width" на странице 7-59 за дополнительной информацией об отчётах ССРР во временном анализаторе TimeQuest.

Вы можете использовать опцию **Разрешить удаление пессимизма общих тактовых путей**, чтобы дать отчёт ССРР в Компоновщике и временном анализе. Эта настройка по умолчанию включена для Stratix III, Cyclone III и новых семейств чипов.

Чтобы иметь доступ к этой опции, выполните следующие шаги:

1. В меню **Назначения** кликните **Настройки**. Раскроется диалоговое окно настроек.
2. В списке **Категории**, за **Настройками Временного Анализатора**, кликните на иконку "+", чтобы раскрыть меню. Кликните **Временной Анализатор TimeQuest**.
3. Включите **Разрешить удаление пессимизма общих тактовых путей**.
4. Кликните **ОК**.

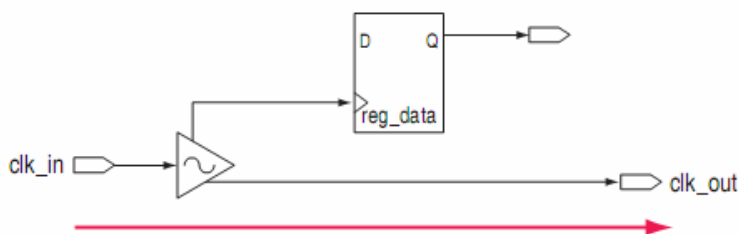
ССРР поддерживается для Stratix III, Cyclone III и новых семейств чипов.

Такты в качестве данных

Большинство проектов FPGA содержат некоторые соединения между двумя узлами, известные как пути данных или как пути тактов. Путь данных - это соединение между выходом синхронного элемента и входом другого синхронного элемента. Путь тактов - это соединения с тактовым входом синхронного элемента. Поскольку проекты FPGA в общем числе - комплексные, в том числе использующие исходные и синхронные интерфейсы, это простейшее представление явно не достаточно.

Соединение между портом `clk_in` и портом `clk_out` может обрабатываться как тактовый путь или как путь данных. Тактовый путь - это путь от порта `clk_in` до входного тактового вывода регистра `reg_data`. Путь данных - это путь от порта `clk_in` до порта `clk_out`. В проекте на рисунке 7-16, пути от порта `clk_in` до порта `clk_out` - оба являются тактовыми путями и путями данных.

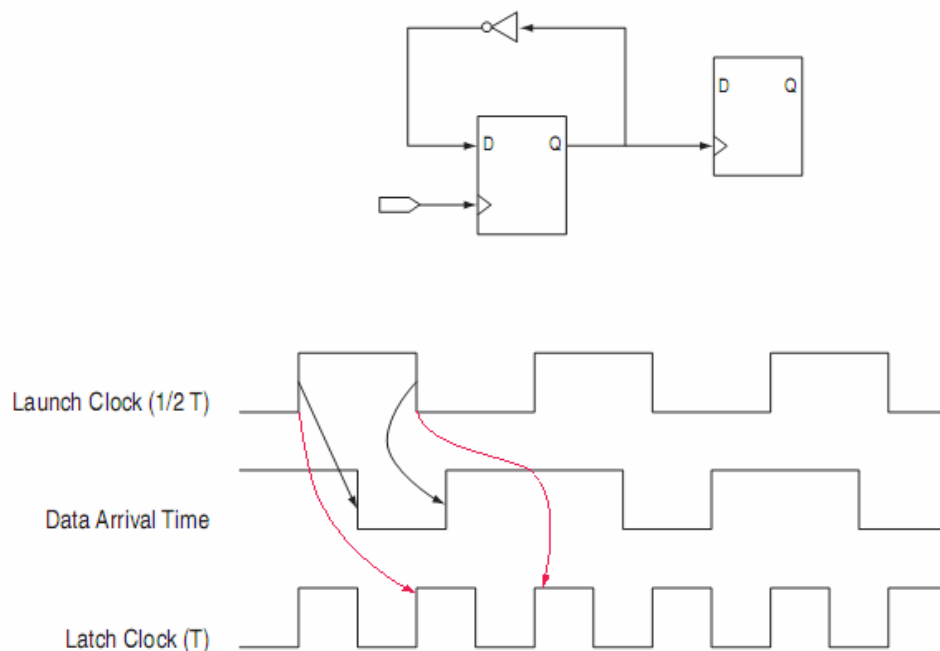
Figure 7-16. Simplified Source Synchronous Output



С помощью анализа такта в качестве данных, временной анализатор TimeQuest обеспечивает более точный анализ путей, основанных на ограничениях пользователя. Для анализа тактовых путей, любой фазовый сдвиг, ассоциированный с PLL, остаётся внутри анализа. Для анализа путей данных, любой фазовый сдвиг, ассоциированный с PLL, остаётся внутри анализа вместо проигнорированных путей.

Анализ тактов в качестве данных также добавляет делители для внутренних сгенерированных тактов, как на рисунке 7-17.

Figure 7-17. Clock Divider (Note 1)



Замечания к рисунку 7-17:

(1) На этом рисунке, инвертор в цепи обратной связи анализируется во время временного анализа. Выход регистра – делителя используется для определения времени запуска, а тактовый вход регистра используется для определения времени защёлкивания.

Исходный и синхронный интерфейс ограничивает тактовый сигнал, который идёт параллельно с сигналом данных. Пара тактов и данных возникает и заканчивается в одном и том же чипе.