

## Аппаратные средства

Cyclone III PLLs поддерживают набор средств для генерального управления тактами. В этой главе рассказывается о реализации умножения и деления, фазового сдвига и программировании рабочего цикла.

### Умножение и деление тактов

Каждая PLL Cyclone III предоставляет синтез тактов для выходных портов PLL, используя коэффициенты  $M/(N \cdot \text{пост-масштабируемых счётчиков})$ . Входной такт делится пред-масштабируемым коэффициентом,  $N$ , и умножается на коэффициент обратной связи  $M$ . Контрольная петля управляет VCO, чтобы сделать  $f_{IN}$  ( $M/N$ ). Каждый выходной порт имеет уникальный пост-масштабируемый счётчик, который делит высокую частоту VCO. Для множества выходов PLL с различными частотами, значение VCO - наименьшее общее кратное выходной частоты, которая возможна из спецификации частот. Например, если требуется выходная частота 33 и 66 МГц, программа Quartus II устанавливает VCO 660 МГц (наименьшее общее кратное 33 и 66 МГц внутри диапазона VCO). Затем на пост-масштабируемых счётчиках уменьшается частота для каждого выходного порта.

Есть один пре-масштабирующий счётчик  $N$  и один умножающих счётчик  $M$  на одну PLL с диапазоном от 1 до 512 для  $M$  и  $N$ . Счётчик  $N$  не используется для контроля рабочего цикла, потому что главной задачей этого счётчика является подсчитывать деление частоты. Есть пять генерированных пост-масштабируемых счётчиков в одной PLL, которые посылают сигнал в GCLKs или на внешний вывод тактов. Эти пост-масштабируемые счётчики имеют

настройки диапазона от 1 до 512 с 50% рабочим циклом. Пост-масштабные счётчики имеют настройки диапазона от до 256 с рабочим циклом не 50%. Сумма верхнего/ нижнего значений счёта выбирается в проекте по выбору значений деления заданных счётчиков.

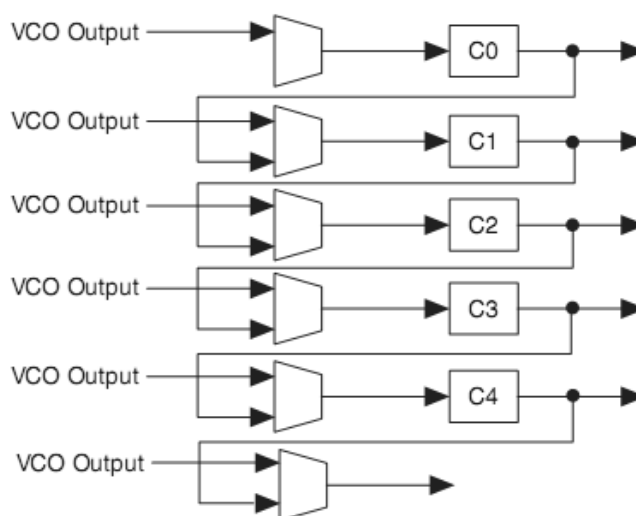
Программа Quartus II автоматически выбирает соответствующие коэффициенты масштаба в соответствие с входной частотой, значением умножения и деления, введёнными в мегафункции altpll.

Выравнивание по фазе между выходными счётчиками определяется использованием спецификации tPLL\_PSERR.

### Каскадирование пост-масштабируемых счётчиков

Cyclone III PLLs поддерживают каскадирование пост-масштабируемых счётчиков, для создания счётчиков более 512. Это реализуется направлением сигнала с выхода одного С счётчика на вход следующего счётчика, как показано на рисунке 6-15.

**Figure 6–15.** Counter Cascading



Когда каскадируются счётчики для реализации большего деления высокой частоты такта VCO, каскадируемые счётчики создают один счётчик с индивидуальными настройками счётчика.

Например, если  $C0 = 4$  и  $C1 = 2$ , то каскадируемое значение  $C0 \times C1 = 8$ .

Пост-масштабируемые счётчики автоматически устанавливаются в программе Quartus II в конфигурационном файле. Это не позволяет использовать реконфигурацию PLL.

### Программируемые рабочие циклы

Программируемые рабочие циклы позволяют PLLs генерировать выходные такты с различными рабочими циклами. Это средство поддерживается только на пост-масштабируемых счётчиках PLL. Вы можете добиться настроек рабочих циклов уменьшая и увеличивая настройку времени счёта для пост-масштабируемых счётчиков. Программа Quartus II использует входную частоту и требуемые характеристики умножителей и делителей для изменения рабочего цикла. Значение пост-масштабируемого счётчика определяет точность рабочего цикла. Точность определяется как 50% поделённых на значение пост-масштабируемых счётчиков. Например, если значение счётчика C0 — 10, шаг в 5% делает возможным реализовать рабочий цикл от 5 до 90 %.

Комбинирование программируемого рабочего цикла с программируемым сдвигом фазы позволяет генерировать точные неперекрывающиеся такты.

## Контрольные сигналы PLL

Вы можете использовать три следующих контрольных сигнала для наблюдения и/или контроля работы PLL и ресинхронизации.

### ***pfdena***

Используйте сигнал *pfdena* для поддержания последней защёлкнутой частоты, так чтобы ваша система имела время для сохранения текущих настроек перед выключением. Сигнал *pfdena* контролирует выход PFD с помощью программируемой защёлки. Если вы запрещаете PFD, VCO выдаёт своё последнее установленное значение контроля напряжения и частоты с некоторым долгосрочным дрейфом на низкой частоте. PLL продолжает работать даже если она вышла из защёлки или запрещён вход таков. Когда запрещён PFD, не используйте сигнал *locked* для определения была ли защёлкнута PLL или нет. Вы можете использовать свой собственный контрольный сигнал или контрольные сигналы, доступные в схеме переключения тактов (*activeclock*, *clkbad[0]* или *clkbad[1]*) для контроля над сигналом *pfdena*.

### ***areset***

Сигнал *areset* — это сброс или ресинхронизация входа каждой PLL. Входные выводы чипа или внутренняя логика могут управлять этими входными сигналами. Когда подаётся "1" — сбрасываются счётчики PLL, выходы PLL и устанавливается выход из защёлки PLL. VCO восстанавливает свои начальные настройки. Когда подаётся "0" — PLL ресинхронизируется по своим входам и перезажёлкивается.

Вы должны назначать сигнал *areset* каждый раз, когда PLL теряет защёлку, чтобы гарантировать правильную фазовую зависимость между входными и выходными тактами PLL. Вы должны включать сигнал *areset* в свой проект, если одно из следующих состояний верно:

- В проекте разрешена реконфигурация PLL и переключение тактов;
- Фазовая зависимость между входным и выходным тактами PLL должна поддерживаться после состояния потери защёлки.

Если входной такт для PLL не стабилен при включении питания, установите сигнал *areset* после того, как входные такты стабилизируются.

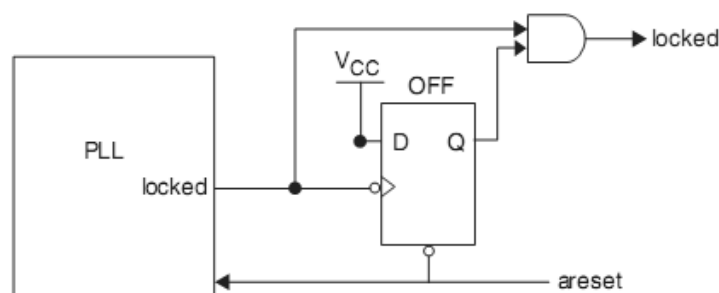
### ***locked***

Выход *locked* показывает, что PLL защёлкнулось по входному такту и выходы тактов PLL выдают требуемые фазу и частоту, установленную в менеджере плагинов Quartus II MegaWizard®. Без дополнительных схем, сигнал защёлки переключается, когда PLL начал процесс защёлкивания. Блок детектирования защёлки создаёт сигнал для ядра логики, который даёт индикацию, если такт в цепи обратной связи защёлкнулся по поступающему такту по фазе и частоте.

Altera рекомендует вам использовать сигналы *areset* и *locked* в вашем проекте для контроля и наблюдения за состоянием вашей PLL. Когда оба сигнала *areset* и *locked* разрешены, в мегафункцию ALTPLL добавляется специальная логики, чтобы улучшить устойчивость сигнала *locked*.

Реализация показана на рисунке 6-16.

**Figure 6–16.** Locked Signal Implementation



Если вы используете инструмент SignalTap® II для пробирования сигнала *locked* перед D-триггером, сигнал *locked* — "0" только когда *areset* снят. Если сигнал *areset* не разрешён, специальная логика не размещается в мегафункции ALTPLL.

## Переключатель тактов

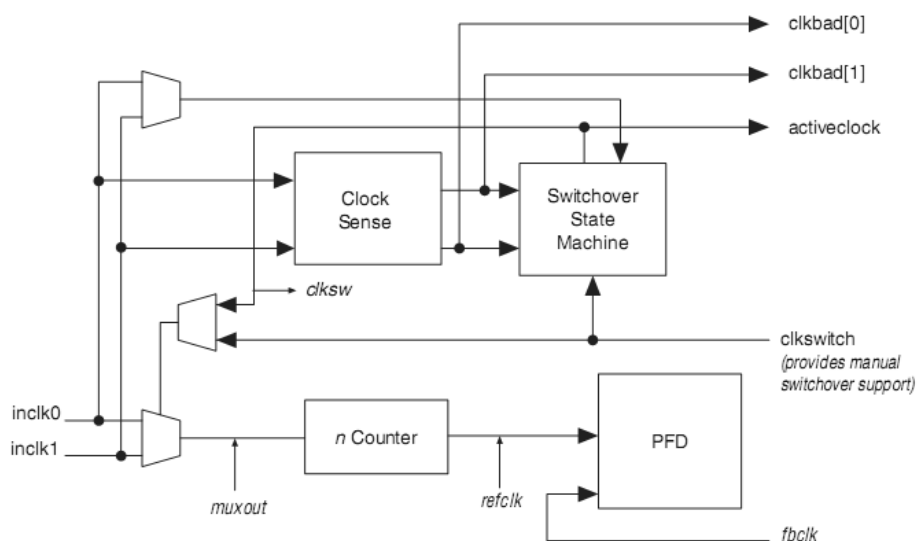
Средство переключения тактов позволяет переключаться между двумя поступающими тактами. Используйте это средство для резервирования или для приложений с двух тактовыми доменами, например, системы включает резервный такт, когда предыдущий такт остановился. Проект может выполнять переключение тактов автоматически, когда такты долго не переключаются или основываясь на пользовательском контрольном сигнале *clkswitch*.

### Автоматический переключатель тактов

Cyclone III PLLs поддерживают способность к полной конфигурации переключателя тактов. Когда текущий поступающий такт не установлен, блок чувствительности к такту автоматически переключает на запасной такт для PLL. Схема переключателя тактов также выдаёт три статусных сигнала - *clkbad[0]*, *clkbad[1]* и *activeclock* — от PLL для реализации различных схем переключения. Вы можете выбрать источник тактов для запасного такта, подключив его к порту *inclk1* PLL в вашем проекте.

На рисунке 6-17 показана блок-схема переключателя тактов, реализованная в PLL.

**Figure 6–17.** Automatic Clock Switchover Circuit



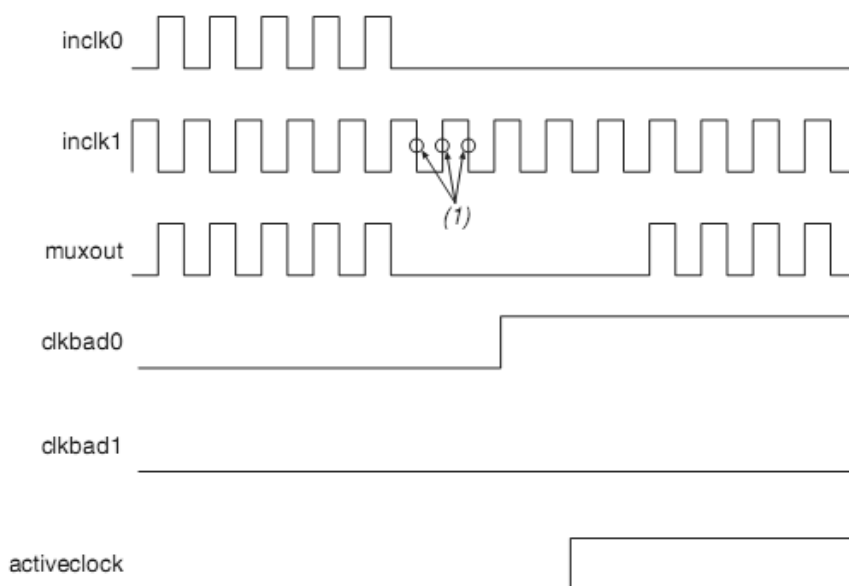
Есть два возможных способа использовать средство переключателя тактов:

■ Используйте схему переключателя тактов для переключения с *inclk0* на *inclk1* работающих на одной частоте. Например, в приложениях, которым требуется резервный такт с той же частотой, что и поступающий такт, конечный автомат переключателя генерирует сигнал, который контролирует выбор мультиплексора, показанный на рисунке 6-17. В этом случае, *inclk1* становится поступающим тактом для PLL. Автоматический переключатель может переключать вперёд и назад между тактами *inclk0* и *inclk1* любое количество раз, когда один из тактов пропадает, а другой остаётся доступным.

■ Используйте вход *clkswitch* для пользовательского или системного переключения состояний. Это возможно для переключений на одинаковой частоте или для переключения между входами различных частот. Например, если *inclk0* — 66 МГц и *inclk1* — 200 МГц, вы должны контролировать переключатель тактов, поскольку автоматическая схема чувствительности к такту не может наблюдать первичный и вторичный такт, частоты которых отличаются более чем на 20%. Это средство особенно полезно, когда источники тактов могут происходить от различных соединительных плат, и требуется контролируемый системой переключатель между рабочими частотами. Выбирайте вторую тактовую частоту так, чтобы VCO работало в рекомендованном частотном диапазоне. Так, устанавливайте счётчики M, N и C, чтобы сохранить рабочую частоту VCO в рекомендованном диапазоне.

На рисунке 6-18 показан пример временных диаграмм средства переключателя тактов, когда используется автоматическое детектирование потерянных тактов. Здесь, сигнал *inclk0* установился в "0". После того, как сигнал *inclk0* остался в "0" в течение двух циклов, схема чувствительности к такту устанавливает в "1" сигнал *clkbad[0]*. А поскольку поступающий тактовый сигнал не переключается, конечный автомат переключений контролирует

**Figure 6–18.** Automatic Switchover Upon Clock Loss Detection (Note 1)



мультиплексор сигналом *clksw* для переключения на *inclk1*.

Примечание к рисунку 6-18:

- (1) Переключатель разрешается на фронте спада *inclk0* или *inclk1*, в зависимости от того, какой такт доступен. На этом рисунке, переключатель разрешается по спаду *inclk1*.

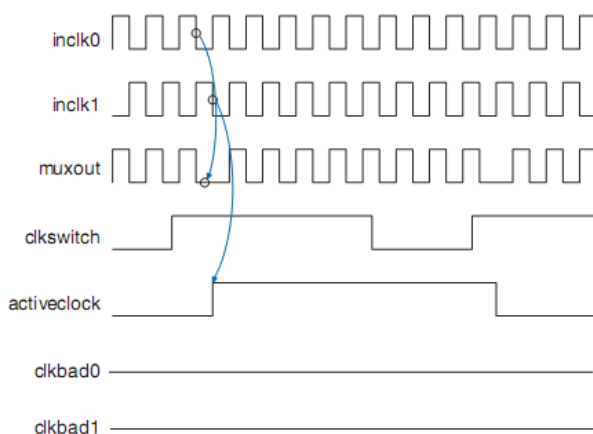
## Ручная отмена

Когда используется автоматический переключатель, переключайте входы тактов, используя средство ручного переключения по входу *clkswitch*.

На рисунке 6-19 показан пример временных диаграмм, иллюстрирующих средство переключения, контролируемое *clkswitch*. В этом случае, оба источника тактов функционируют, а *inclk0* выбран в качестве поступающего такта. Переход из "0" в "1" сигнала *clkswitch* инициирует процесс переключения. Сигнал *clkswitch* должен быть в "1" в течение не менее трёх тактовых циклов (не менее трёх наибольших тактовых циклов, если *inclk0* и *inclk1* имеют разные частоты). На фронте спада *inclk0* счётчики поступающего такта, *muxout*, закрываются, чтобы предотвратить тактовые сбои. На фронте спада *inclk1*, мультиплексор поступающего такта переключает с входа *inclk0* на вход *inclk1* поступающей частоты PLL. Тогда, когда изменяется сигнал *activeclock*, показывается, что такт поступает на PLL.

В этом режиме, сигнал *activeclock* копирует сигнал *clkswitch*. Так как оба блока функционально неизменны во время ручного переключения, никакой сигнал *clkbad* не устанавливается в "1". Так как схема переключения чувствительна к положительному фронту, фронт спада сигнала *clkswitch* не может переключить назад от *inclk1* к *inclk0*. Когда сигнал *clkswitch* снова переходит в "1", процесс повторяется. Сигнал *clkswitch* и автоматическое переключение работают только тогда, когда доступны такты для переключения. Если такты не доступны, конечный автомат ожидает, пока такты не станут доступны.

**Figure 6–19.** Clock Switchover Using the *clkswitch* Control



## Ручное переключение тактов

Cyclone III PLLs поддерживают ручное переключение, когда сигнал *clkswitch* контролирует, какой из входов *inclk0* или *inclk1* будет поступающим тактом PLL. Характеристики ручного переключения тактов схожи со средством ручной отмены в автоматическом переключателе тактов, поскольку схема переключения чувствительна ко фронту. Когда сигнал *clkswitch* устанавливается в "1", стартует процесс переключения. Фронт спада сигнала *clkswitch* не может переключить обратно на предыдущий входной такт.

За дополнительной информацией о поддержке программы PLL в программе Quartus II, обратитесь к руководству пользователя мегафункцией ALTPLL.

## Рекомендации

Используйте следующие рекомендации для проекта с переключением тактов в PLLs:

- Детектирование потерянного такта и автоматическое переключение тактов требует, чтобы частоты *inclk0* и *inclk1* отличались не более, чем на 20% одна от другой. Неследование этим требованиям приведёт к ошибочному выработыванию сигналов *clkbad[0]* и *clkbad[1]*.

■ Когда используется ручное переключение тактов, различие между *inclk0* и *inclk1* может быть более 20%. Однако, различия между двумя источниками тактов (по частоте и/или фазе) может привести к потере защёлки в PLL. Сбрасывайте PLL, чтобы добиться корректной фазовой зависимости между входным и выходным тактами.

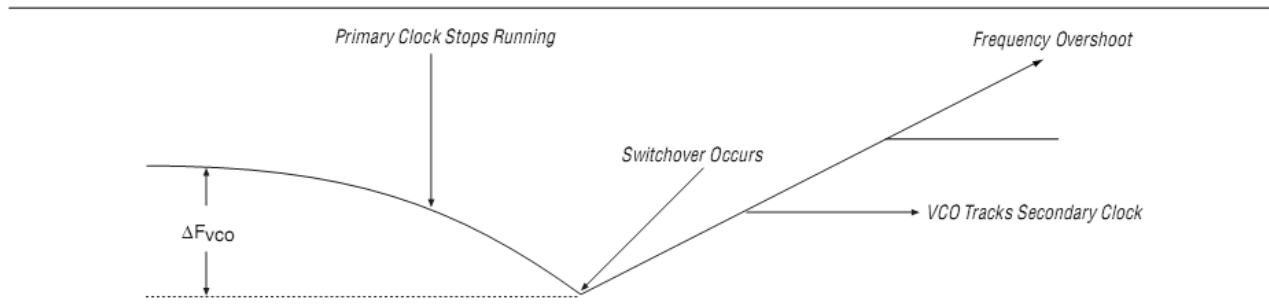
■ Приложениям, которым требуется средство переключения тактов и малый дрейф частоты, необходимо использовать узкополосную PLL. Узкополосная PLL реагирует медленнее, чем широкополосная PLL по отношению к изменению тактов. Когда происходит переключение тактов, узкополосная PLL передаёт остановку тактов для выхода медленнее, чем широкополосная PLL. Узкополосная PLL фильтрует джиттер поступающего такта. Однако знайте, что узкополосная PLL также увеличивает время защёлки.

■ После выполнения переключения тактов, может начаться конечный период ресинхронизации PLL, для защёлкивания по новому такту. Точное время, нужное для перезащёлкивания PLL зависит от конфигурации PLL.

■ Если фазовая зависимость между входным тактом PLL и выходным тактом PLL очень важна в вашем проекте, установите сигнал *areset* на 10 нс, после выполнения переключения тактов. Подождите перехода сигнала *locked* (или переключения защёлки) в "1", прежде чем разрешать выходы тактов от PLL.

■ На рисунке 6-20 показано, как частота VCO постепенно уменьшается, когда первичный такт пропадает и, затем, увеличивается, когда VCO защёлкивается по вторичному такту. После того, как VCO защёлкнулось по вторичному такту, может наступить перегрузка (состояние перенасыщения частоты) в VCO.

**Figure 6-20.** VCO Switchover Operating Frequency



■ Сбросьте систему во время переключения, если она не толерантна к изменению частоты во время ресинхронизации PLL. Вы можете использовать статусные сигналы *clkbad[0]* и *clkbad[1]* для выключения PFD (*pfdena* = 0), так чтобы VCO сохранило свою последнюю частоту. Вы можете также использовать конечный автомат для переключения на вторичный такт. Разрешая PFD, сигналы разрешения выходного такта (*clkena*) могут запретить выходы тактов во время переключения и ресинхронизации. Когда индикация защёлки стабилизируется, система может снова разрешить выход такта или тактов.