

## ЛАБОРАТОРНАЯ РАБОТА № 2 "Управляемый генератор синхросигналов"

### 1. ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

#### 1.1. Использование синхросигналов в проектах САПР Quartus.

Все сигналы, используемые для синхронизации чтения/записи в регистры, память и для многого другого будем называть синхросигналами или тактирующими сигналами(clock).

В результате компиляции проекта с предыдущей лабораторной работой при использовании синхронных элементов вы наверняка получили следующее предупреждение:

*Warning: "Found pins functioning as undefined clocks and/or memory enables"*

То есть система определила, что некоторые пины используются для синхронизации, но в свойствах проекта они не определены как синхросигналы. Об этом вам система дает пояснение в помощи (рис. 1), которую можно вызвать по этому предупреждению и не только. Помощь вызывается в окне Processing из пункта Help в меню по правому клику мыши.

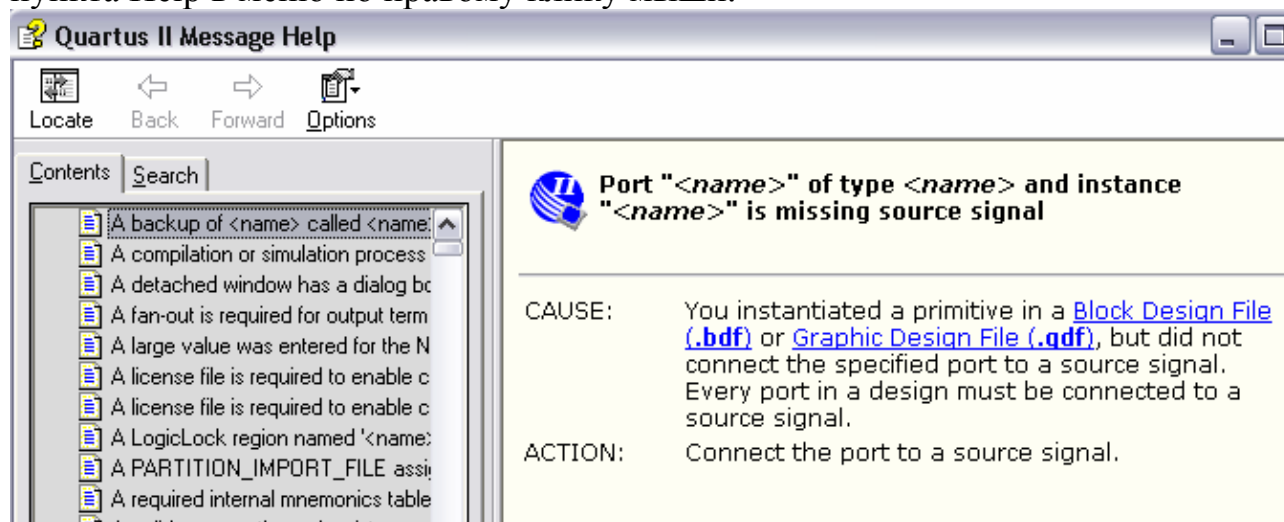


Рис.1. Получение помощи по предупреждениям и ошибкам.

**CAUSE:** *The Timing Analyzer found one or more pins that are functioning as undefined clocks in the design. The Quartus II software identifies a pin that feeds the clock input of a register that does not have a Clock Settings timing assignment as an undefined clock.*

**ACTION:** *Altera recommends that you create clock settings and assign the clock settings to the pin(s) that are functioning as clocks in the design. If you do not want the Quartus II software to treat a pin feeding the clock input of a register as an undefined clock, you can turn on the Not a Clock assignment for the pin.*

Если не наложить временные ограничения, то Quartus не будет знать, с какой максимальной частотой эти сигналы будут переключаться.

В случае отсутствия временных ограничений (timing constraints) на синхросигнал, Quartus оптимизирует пути распространения сигналов для всех син-

хросигналов в системе. Если существует всего один сигнал, явно определенный как синхросигнал с ограничениями, оптимизация будет начинаться с него; если несколько – оптимизация будет производиться для всех сигналов в порядке уменьшения степени жесткости ограничений.

Поэтому в том случае, когда в проекте всего один синхросигнал, временные ограничения необязательны, но рекомендуются, поскольку в противном случае система будет выдавать предупреждение на каждый сигнал, зависящий от этого синхросигнала.

Настроить временные ограничения можно в меню Assignments->Timer Analysis Settings или непосредственно в настройках проекта открыть соответствующую ветку. Мы будем использовать Classic Timer Analyzer, поэтому надо убедиться в том, что выбран именно этот анализатор. После этого можно перейти к ветке Classic Timer Analyzer Settings (рис. 2).

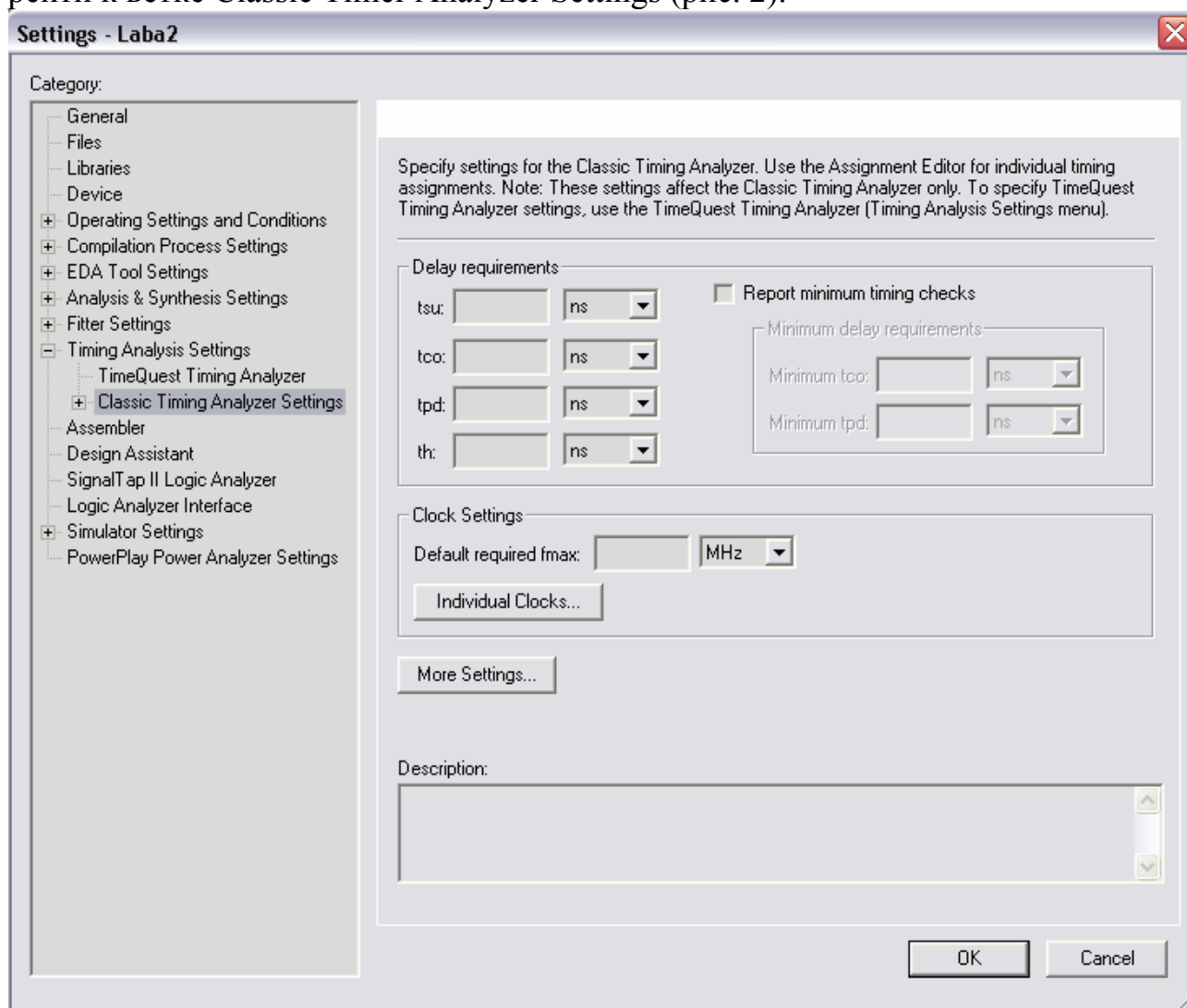


Рис. 2. Настройки временного анализатора в настройках проекта

В настройках можно указать требования к задержкам, а также создать настройки для синхросигналов. Нас будет интересовать последнее, для этого нажатием на Individual Clocks перейдем к настройкам синхросигналов (рис. 3).

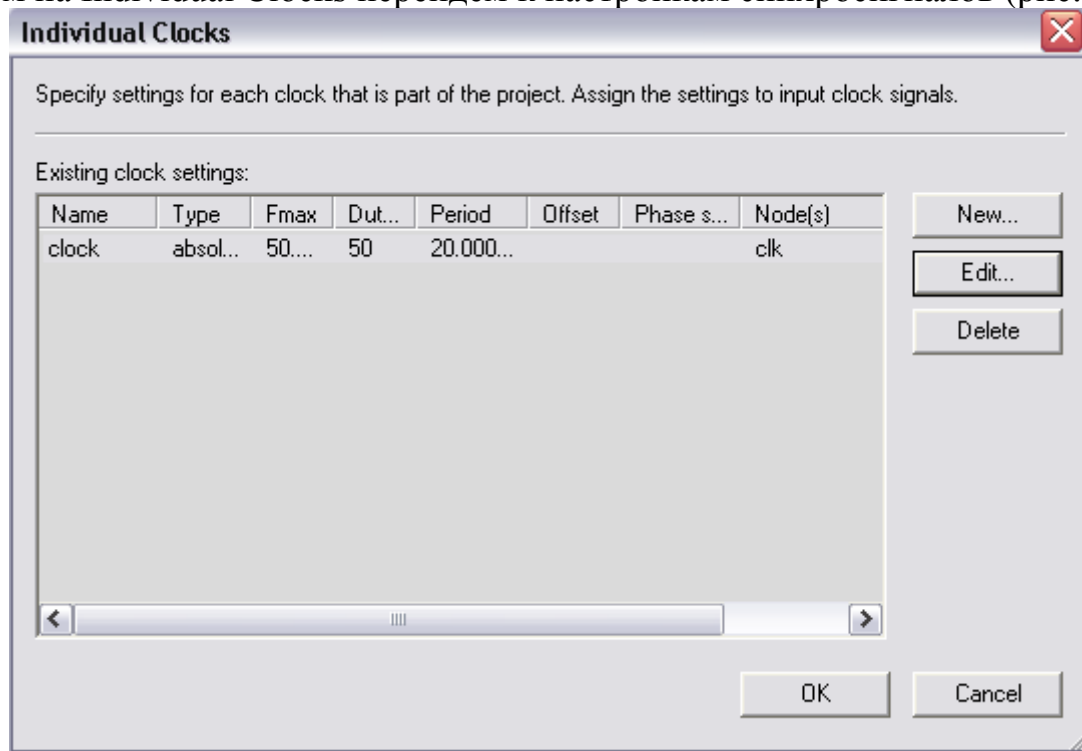


Рис. 3. Настройки синхросигналов

Здесь можно добавлять/изменять/удалять шаблоны настроек. Рассмотрим добавление нового шаблона (рис. 4).

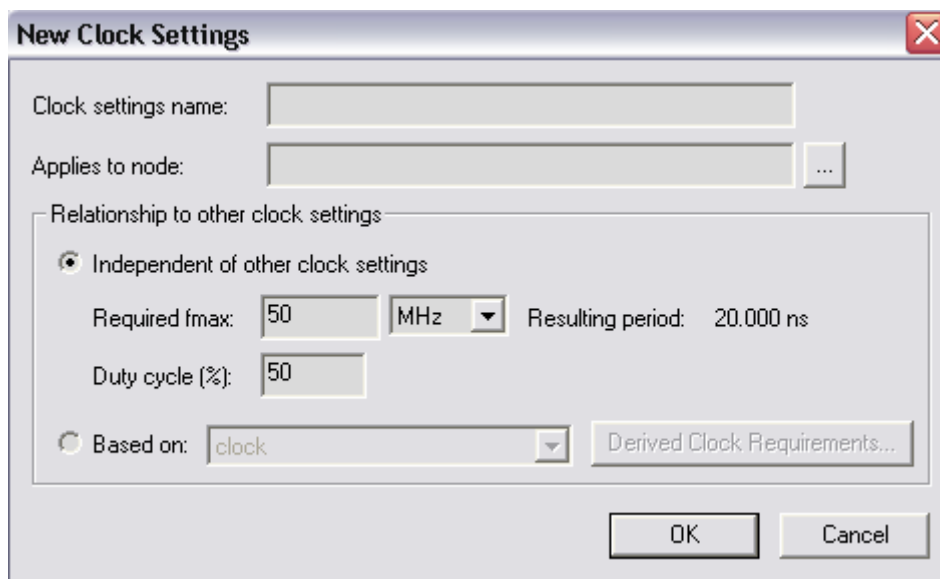


Рис. 4. Добавление нового шаблона настроек.

В шаблоне необходимо указать имя шаблона в поле Clock settings name и сигнал, к которому этот шаблон будет применяться в поле Applies to node. Кон-

кретные параметры синхросигнала можно указать двумя способами: с нуля или основываясь на уже существующих шаблонах. В первом случае необходимо выбрать пункт Independent of other clock settings и задать максимальную частоту  $f_{max}$  и коэффициент заполнения (Duty Cycle) – отношение длительности импульса к промежутку между импульсами. Во втором случае необходимо указать шаблон, на основе которого будет формироваться новый, в поле Based on и указать требования к заимствованным настройкам (рис. 5). Там можно указать множитель частоты, сдвиг, инверсию и др.

**Derived Clock Requirements**

Specify the timing requirements for derived clock settings

Absolute timing requirements

Clock settings name: clock

Required fmax: 50 MHz

Resulting period: 20.000 ns

Multiply base absolute clock fmax by: 1

Divide base absolute clock fmax by: 1

Duty cycle (%): 50

Resulting fmax: 50.0 MHz

Resulting period: 20.000 ns

Offset from base absolute clock fmax: ns

Clock phase shift (in degrees with respect to derived clock): °

☐ Invert base clock

OK Cancel

Рис. 5. Требования к заимствованным настройкам

Существующие шаблоны можно использовать при моделировании. При вводе синхросигналов в файле временных диаграмм при использовании инструмента Clock можно указать шаблон и к сигналу будут применены настройки из шаблона (рис. 6). Сигналу будет задана соответствующая частота, коэффициент заполнения и другие настройки, если они были указаны.

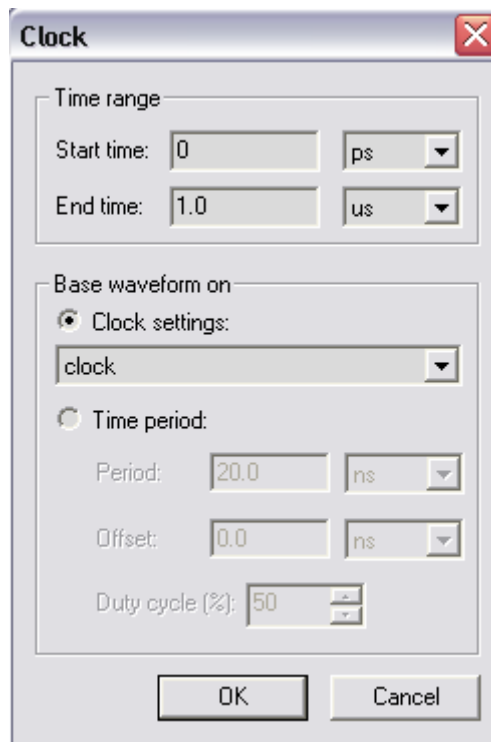


Рис. 6. Использование настроек синхронизации в инструменте Clock

### 1.2. Проектирование генератора синхросигналов.

Сгенерированные сигналы предполагается использовать для синхронизации работы различных частей системы как единого целого. В соответствии с заданием генератор будет представлять собой блок, состоящий из двух частей: блока выработки единичного синхросигнала с заданным периодом и длиной импульса и блока выработки группы опорных сигналов. Для составных блоков предварительно создаются символы.

Создание символа выполняется командой меню File->Create/Update->Create symbol files for current file при открытом файле дизайна (\*.bdf).

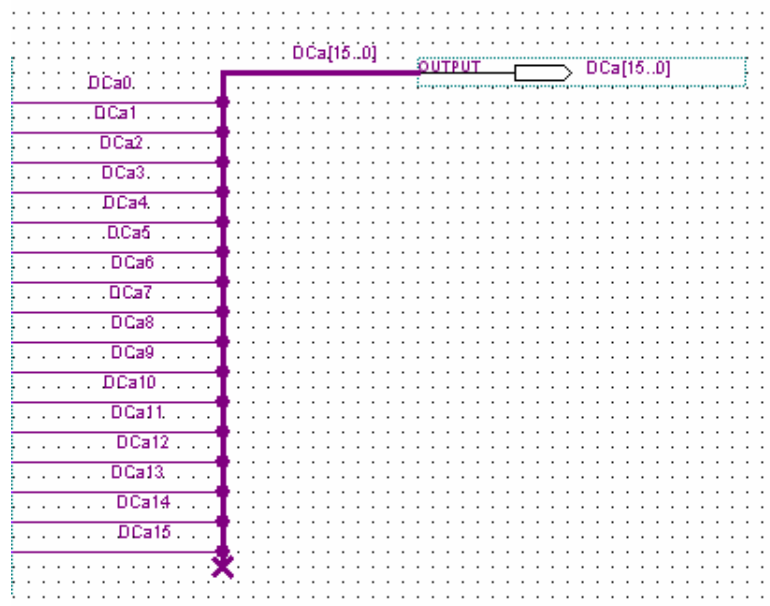


Рис. 7. Объединение сигналов на шину

Опорные сигналы второго блока должны быть объединены в шину DCa[15..0]. Стоит сказать несколько слов об использовании шин в Quartus. Шины на дизайне обозначаются утолщенными линиями. Названия шинам даются в формате **name[n..0]**, где  $n+1$  – разрядность шины, name – название шины. Чтобы входной или выходной пин стал шиной достаточно использовать в его названии такую нотацию. С шины можно снять определенный бит, для этого отводится обычная тонкая линия и подписывается в формате **nameN** или **name[N]**, где **N** – номер разряда шины, начиная с нуля. Перед отведением от шины определенного бита, ее необходимо обязательно подписать, даже если она отводится или заводится на входной/выходной пин. При моделировании с использованием шин рекомендуется использовать 16-ричную(Hexidecimal) систему счисления (Radix), поэтому лучше сразу указать 16-ричную систему для всех пинов добавляемых в файл векторных диаграмм. Значение участку шины присваивается двойным левым щелчком или с помощью инструмента Arbitrary Value (рис. 8).

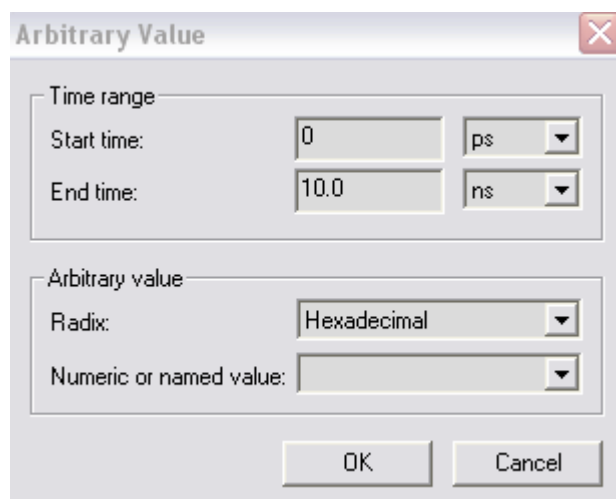


Рис. 8. Окно инструмента Arbitrary Value

Для выполнения задания по лабораторной работе могут понадобиться счетчики, дешифраторы, вычитатели, триггеры и другие элементы в зависимости от реализации генератора. В качестве этих элементов можно использовать компоненты библиотеки Quartus. Использовать готовые генераторы из библиотеки Quartus запрещается. Большой набор компонентов находится в разделе Others->Maxplus2 библиотеки компонентов Quartus. Также вместе с этим методическим пособием поставляется помощь по этим элементам - папка library help. Для поиска по элементам серии 74 необходимо искать в индексе пункт 74-series macrofunctions.

## 2. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

2.1. Спроектировать и промоделировать блок, на вход которого поступает синхросигнал **clk** с частотой, заданной по варианту, а на выходе формируется сигнал с формой и указанными по варианту параметрами **m** и **n**, **m** и **n** - количество тактов внешнего синхросигнала **clk**. Указать временному анализатору, что **clk** - синхросигнал, и задать его в файле временных диаграмм с помощью шаблонов настроек временного анализатора.

Усложненный вариант: добавить к входам блока два 8-разрядных значения, представляющие собой параметры **m** и **n**. Гарантировать получение дополнительных баллов за усложненный вариант может только оригинальное решение этого задания. Для сдачи усложненного варианта необходимо подготовить мини-отчет, включающий фамилии членов бригады, схему блока и результаты моделирования.

Две возможные формы сигнала изображены на приведенных ниже рисунках. В таблице указан вариант рисунка.

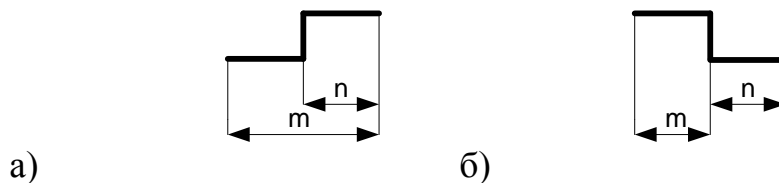


Рис. 8. Форма выходного сигнала.

2.2. Спроектировать и промоделировать блок, на вход которого поступает синхросигнал **clk** с частотой, заданной по варианту, а на выходе формируется 16-разрядный сигнал, представляющий собой 16 опорных синхроимпульсов **DCa** (рис.9). Сигнал должен представлять собой 16 разрядную шину. Форма синхроимпульсов представлена на рис 9. Вариант делителя частоты **clk** для получения сигнала с частотой **clkin** указан в таблице вариантов.

2.3. Объединить два спроектированных блока в один, используя символы составных блоков. Промоделировать этот блок.

Семейство ПЛИС для реализации - Flex10K (изменить семейство ПЛИС(Family) можно в настройках проекта в ветке Device или в пункте меню Assignments->Device)

Для показа выполнения по необходимо продемонстрировать схему по каждому пункту и результаты моделирования пункта 2.3.

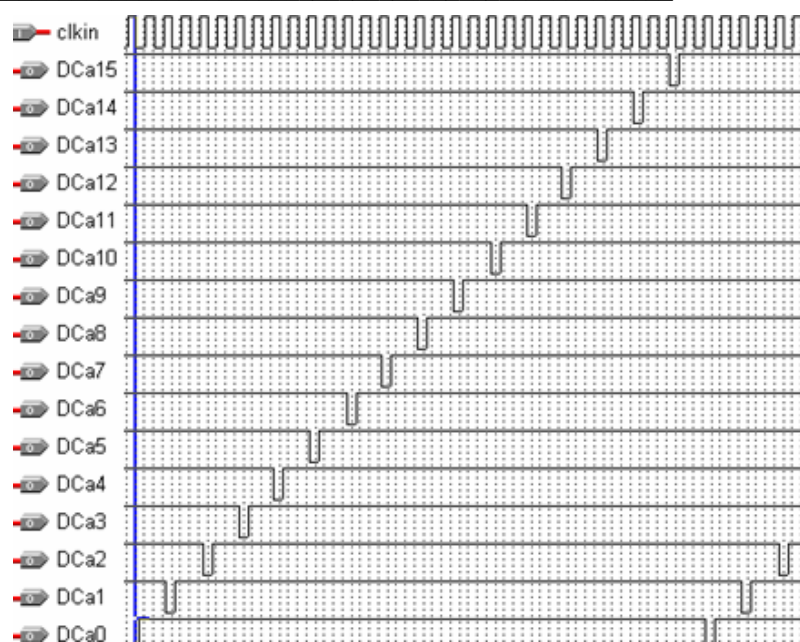


Рис. 9. Вид выходного опорного синхросигнала

### ВАРИАНТЫ ЗАДАНИЙ К ЛАБОРАТОРНОЙ РАБОТЕ № 1

№ варианта	Частота clk, МГц	Форма сигнала	Параметры m и n, тактов	Делитель частоты
1	50	Рис. а	12 и 3	12
2	40	Рис. б	3 и 5	10
3	25	Рис. а	11 и 4	8
4	20	Рис. б	4 и 6	6
5	12,5	Рис. а	10 и 5	4
6	10	Рис. б	5 и 7	2
7	6,25	Рис. а	9 и 6	4
8	5	Рис. б	6 и 8	6

### СОДЕРЖАНИЕ ОТЧЕТА ПО ЛАБОРАТОРНОЙ РАБОТЕ №2

1. Задание.
2. Проектирование блока выработки единичного синхросигнала.
  - 2.1. Схема генератора.
  - 2.2. Результаты моделирования.
3. Проектирование блока опорных сигналов.
  - 3.1. Схема блока.
  - 3.2. Результаты моделирования.
- 4.1. Схема управляемого генератора
- 4.2. Результаты моделирования