

РЕАЛИЗАЦИЯ РАСПРЕДЕЛЕННОГО ЦИФРОВОГО ФИЛЬТРА С НАСТРАИВАЕМЫМИ ПАРАМЕТРАМИ НА МИКРОСХЕМАХ ПРОГРАММИРУЕМОЙ ЛОГИКИ

Ушенина И. В., Елизаров В. Н., Варнавский В. А.

ФГБОУ ВПО «Пензенская государственная технологическая академия», Пенза, <http://www.pgta.ru>

Представлен модуль цифрового фильтра с конечной импульсной характеристикой без умножителей, работающий по принципу распределенных вычислений. Основные характеристики модуля: возможность настройки порядка, разрядности входного и выходного сигналов и коэффициентов; малая латентность по сравнению с классической структурой; доступность для реализации на программируемых ресурсах ПЛИС FPGA. Представление чисел – в дополнительном коде, формат с фиксированной точкой. Проанализированы возможности реализации модуля на базе ПЛИС FPGA. Приведена структура фильтра, в состав которой входят регистры, оперативная память, сумматор, управляемый инвертор, управляющий автомат, счетчик, комбинационная логика. Получены зависимости от настраиваемых параметров фильтра разрядностей его элементов и шин, длительностей управляющих сигналов, и др. Приведены результаты функционального моделирования модуля при заданных параметрах. Рассмотрены вопросы, связанные с ограничениями применения фильтров на базе ОЗУ.

Ключевые слова: цифровой фильтр, распределенные вычисления, латентность фильтра, ПЛИС.

IMPLEMENTATION OF DISTRIBUTED DIGITAL FILTER WITH ADJUSTABLE PARAMETERS ON THE BASE OF PROGRAMMABLE LOGIC INTEGRATED CIRCUITS

Ushenina I. V., Elizarov V. N., Varnavskiy V. A.

Penza State Technological Academy, Penza, <http://www.pgta.ru>

Multiplier-free distributed digital filter with finite impulse response is presented. The main characteristics of the filter are: the possibility of adjusting such filter's parameters as order, input and output signals capacity, coefficients; low latency in comparison with classical structure; availability for implementation on the base of programmable resources of the FPGA. There is two's complement, fixed-point number representation in a filter. Possibilities of the filter implementation on the base of FPGA are analyzed. It is shown that filter consists of registers, RAM memory, adder, controllable inverter, finite state machine, counter, combinatorial logic. The dependencies of the filter's elements and buses capacities, control signals durations, etc. from it's given parameters are obtained. The results of the functional simulation of the filter are shown. The problems concerned with restrictions of using RAM-based digital filters are considered.

Key words: digital filter, distributed arithmetic, filter's latency, FPGA.

Введение

В современных условиях важно завершить проектирование изделия в оговоренный срок. При этом большую роль играют формализация процесса проектирования и, по возможности, использование готовых технических решений. В случае проектирования устройств цифровой обработки сигналов (ЦОС) готовым техническим решением могут стать программный код или описание на аппаратном уровне (HDL) цифрового фильтра, позволяющие выполнить настройку основных параметров устройства.

Целью данной работы является создание доступного для реализации на микросхемах программируемой логики (ПЛИС) VHDL-модуля цифрового фильтра с конечной импульсной характеристикой (КИХ), работающего по принципу распределенных вычислений. Основные параметры фильтра – порядок, разрядность входных отсчетов,

разрядность целой и дробной частей входных отсчетов и результата фильтрации – могут настраиваться по требованию проектировщика, поэтому модуль может использоваться как готовое техническое решение в различных проектах.

Элементная база для реализации устройств ЦОС

В настоящее время для аппаратной реализации цифровых фильтров обычно выбираются сигнальные процессоры или ПЛИС архитектуры FPGA (field programmable gates array). Главным преимуществом FPGA перед сигнальными процессорами является возможность параллельных вычислений [2]. Кроме того, FPGA позволяют реализовывать цифровые фильтры различными способами. Наиболее скоростное устройство с гарантированными временными характеристиками можно получить при использовании встроенных аппаратных модулей цифровой обработки сигналов с умножителями-аккумуляторами. По составу модулей (умножители 18-разрядных переменных, сумматоры / вычитатели, управляющая логика, регистры) и их количеству на кристалле FPGA ведущих производителей [4, 5] одинаковых классов сопоставимы (таблица 1). Тем не менее, недорогие кристаллы FPGA могут не располагать достаточным количеством ЦОС-модулей или не иметь их вовсе.

Таблица 1. Ресурсы недорогих FPGA последних поколений от ведущих производителей

Семейство, год выпуска	Программируемая логика	Блочная память (ОЗУ), кбит	Модули ЦОС, шт.
Spartan 6, Xilinx, 2009	300 – 11519 CLB	216–4824	8–180
Artix 7, Xilinx, 2011	625 – 28125 CLB	720–18540	20–1040
Cyclone V, Altera, 2011	3840 – 147443 LE	1760–12200	25–342

Другим подходом является использование программируемой логики (триггеров и распределенного ОЗУ с программируемыми соединениями) и встроенного блочного ОЗУ недорогих FPGA [1, 3]. Количество этих ресурсов у FPGA одного класса также является сопоставимым (таблица 1): состав одного конфигурируемого логического блока (CLB) приблизительно равен четырем логическим элементам (LE). В итоге разработка удешевляется, но за работоспособностью и характеристиками полученного фильтра (в частности, временными) должен следить проектировщик.

При реализации цифровых фильтров на базе программируемых ресурсов FPGA наиболее громоздким и критичным с точки зрения временных характеристик устройством является умножитель, поэтому реализация фильтров по принципу распределенных вычислений, позволяющая обойтись без умножителей, представляет практический интерес.

Постановка задачи

Работа КИХ-фильтра описывается уравнением дискретной свертки (1):

$$Y[n] = \sum_{n=0}^{N-1} C[n] \cdot X[n] \quad (1)$$

где N – порядок (количество звеньев) фильтра, $C[n]$ – коэффициенты фильтра, $X[n]$ – отсчеты входного сигнала, $Y[n]$ – результаты фильтрации.

Суть метода распределенных вычислений [2] заключается в перераспределении вычислительных операций и переходе от классического уравнения дискретной свертки (1) к выражению вида:

$$Y[n] = \sum_{b=0}^{B-1} 2^b \cdot \sum_{n=0}^{N-1} C[n] \cdot x_b[n] \quad (2)$$

где B – разрядность отсчетов входного сигнала фильтра, $x_b[n]$ – разряды одного порядка всех отсчетов входного сигнала, участвующих в формировании $Y[n]$.

Так как $C[n]$ известны, вместо умножения $C[n]$ на $X[n]$ по мере поступления новых отсчетов можно заранее рассчитать и поместить в ОЗУ FPGA все возможные комбинации сумм произведений $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$. Адресами для чтения ОЗУ будут выступать $x_b[n]$ (рисунок 1).

Разряды отсчетов входного сигнала (адресная шина)				Суммы произведений $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$
$x_b[N-1]$	$x_b[N-2]$...	$x_b[0]$	
0	0	...	0	$0 \cdot C[N-1] + 0 \cdot C[N-2] + \dots + 0 \cdot C[0]$
0	0	...	1	$0 \cdot C[N-1] + 0 \cdot C[N-2] + \dots + 1 \cdot C[0]$
...
1	1	...	1	$1 \cdot C[N-1] + 1 \cdot C[N-2] + \dots + 1 \cdot C[0]$

Рисунок 1. Формирование адресов и содержимого ОЗУ

Еще одним преимуществом распределенного фильтра, помимо отсутствия умножителей, является его небольшая латентность, зависящая, в отличие от фильтра классической структуры, не от порядка фильтра N , а от разрядности отсчетов входного сигнала B [2].

В настоящей работе авторы решают следующие задачи:

- обоснование доступной для реализации на базе FPGA структуры модуля КИХ-фильтра с настраиваемыми параметрами, работающего по принципу распределенных вычислений;
- формализация зависимостей внутренних параметров элементов модуля от настраиваемых параметров для параметризованного описания модуля на языке VHDL;
- описание модуля на языке VHDL и подтверждение работоспособности модуля методом моделирования.

Обоснование структуры модуля фильтра

Формуле (2) в самом общем виде можно поставить в соответствие устройство, представленное на рисунке 2. Помимо ОЗУ, оно содержит цепочку регистров, принимающих

и поразрядно сдвигающих $X[n]$, схему сдвига, выполняющую умножение на 2^b , и аккумулятор, накапливающий результаты чтения ОЗУ и сдвига.

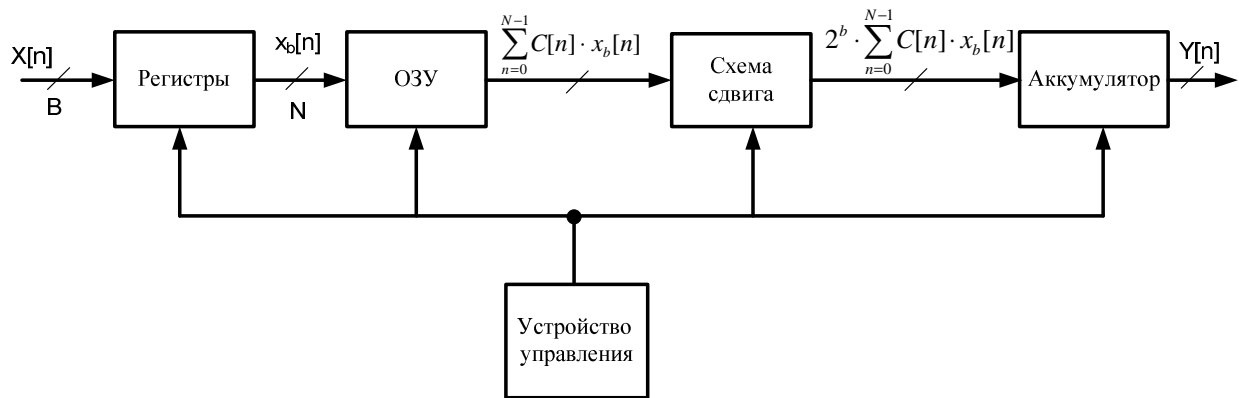


Рисунок 2. Обобщенная структура распределенного КИХ-фильтра

Практическая реализация распределенного фильтра требует уточнений формулы (2) и структуры, представленной на рисунке 2.

Так как $X[n]$, $C[n]$ и $Y[n]$ могут принимать как положительные, так и отрицательные значения, для представления чисел целесообразно использовать дополнительный код. С учетом того, что старшим разрядом $X[n]$ является знаковый, уравнение распределенной фильтрации будет описываться соотношением вида:

$$Y[n] = -2^{B-1} \cdot \sum_{n=0}^{N-1} C[n] \cdot x_{B-1}[n] + \sum_{b=0}^{B-2} 2^b \cdot \sum_{n=0}^{N-1} C[n] \cdot x_b[n] \quad (3)$$

Компонента $-2^{B-1} \cdot \sum_{n=0}^{N-1} C[n] \cdot x_{B-1}[n]$ соответствует обработке знаковых разрядов $X[n]$,

участвующих в формировании $Y[n]$ [2]. Значение $\sum_{n=0}^{N-1} C[n] \cdot x_{B-1}[n]$ наряду с остальными суммами произведений сохраняется в ОЗУ FPGA.

Использование дополнительного кода позволяет обойтись сумматором при сложении положительных и отрицательных чисел. Однако чтобы получить корректный результат при вычислении компоненты формулы (3), формируемой знаковыми разрядами, результат чтения из ОЗУ требуется преобразовать в равное по модулю, но противоположное по знаку число (т.е. выполнить преобразование прямого кода в дополнительный или наоборот). Для выполнения этой операции после ОЗУ ставится управляемый инвертор (рисунок 3). К результату инверсии прибавляется единица.

На рисунке 3 представлена предлагаемая структура КИХ-фильтра, которая может быть реализована на программируемых ресурсах общего назначения FPGA.

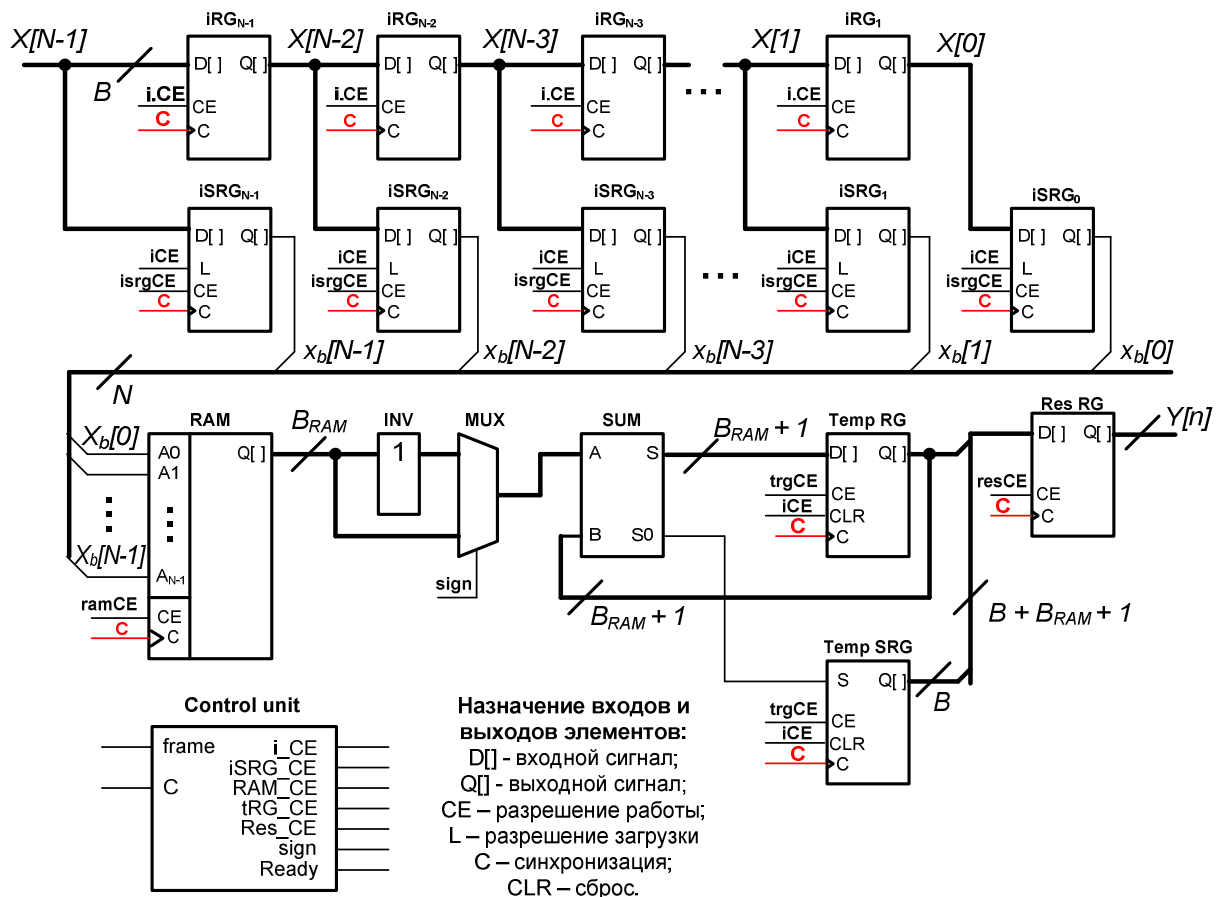


Рисунок 3. Схема модуля распределенного КИХ-фильтра, реализуемая на программируемых ресурсах FPGA

Входные регистры iRG принимают и перемещают по цепочке отсчеты входного сигнала $X[n]$; перезапись регистров iRG происходит при поступлении нового отсчета $X[n]$. Содержимое сдвиговых регистров iSRG обновляется одновременно с регистрами iRG, после чего разряды $X[n]$ последовательно выдаются на адресную шину, формируя адреса для ОЗУ.

ОЗУ, обозначенное на рисунке «RAM», хранит суммы произведений $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$ для всех возможных комбинаций $x_b[n]$ при заданном N . Результат чтения ОЗУ поступает на вход А сумматора Sum.

Сумматор и временные регистры tempRG и tempSRG образуют аккумулятор и схему сдвига. Умножение на 2^b (3) может быть выполнено, либо сдвигом на разряд влево результата чтения ОЗУ перед подачей на вход сумматора А, либо сдвигом на разряд вправо результата суммирования перед подачей на вход сумматора В. Выбран второй вариант как более удобный для аппаратной реализации. Таким образом, младший разряд результата суммирования, S_0 , не участвует в дальнейших операциях накопления, а отправляется во временный регистр tempSRG, формирующий младшие разряды $Y[n]$. Перед старшим разрядом выходного сигнала сумматора S в регистре tempRG добавляется один дополнительный разряд, равный старшему разряду

сигнала S . Регистр tempRG также участвует в преобразовании значения $\sum_{n=0}^{N-1} C[n] \cdot x_{B-1}[n]$ из прямого кода в дополнительный и обратно: на первом шаге вычислений, при поступлении на вход А сумматора суммы $\sum_{n=0}^{N-1} C[n] \cdot x_0[n]$, на вход В сумматора из регистра tempRG подается 2^{B-1} , т.е. имитируется прибавление единицы к инверсии $\sum_{n=0}^{N-1} C[n] \cdot x_{B-1}[n]$.

После выполнения B циклов чтения ОЗУ, суммирования и сдвига, регистр результата ResRG собирает $Y[n]$ из текущего содержимого регистров tempRG и tempSRG; при этом часть разрядов с обеих сторон может быть отброшена.

Основным элементом управляющего устройства, Control unit, является управляющий автомат, который формирует для остальных элементов модуля управляющие сигналы – iCE, isrgCE, ramCE, sign, trgCE, resCE. Активным (разрешающим) уровнем для всех управляющих сигналов является логическая единица. Назначение управляющих сигналов следующее.

- iCE – сигнал разрешения загрузки $X[n]$ во входные регистры iRG и регистры сдвига iSRG; формируется при поступлении на управляющее устройство «кадрового» сигнала frame, сопровождающего новый отсчет $X[n]$; активен в течение одного периода тактового сигнала.
- isrgCE – сигнал разрешения сдвига для регистров iSRG; активен в течение $B-1$ периодов тактового сигнала.
- ramCE – сигнал разрешения чтения ОЗУ; становится активным одновременно с isrgCE; активен в течение B периодов тактового сигнала.
- trgCE – разрешающий сигнал для регистров tempRG и tempSRG; становится активным на такт позже, чем isrgCE; активен в течение B периодов тактового сигнала.
- sign – сигнал разрешения инверсии результата чтения ОЗУ по адресу, сформированному знаковыми разрядами $X[n]$; активен в течение одного периода тактового сигнала.
- resCE – сигнал, разрешающий запись в регистр результата ResRG. Становится активным по окончании B циклов «чтение ОЗУ – суммирование – сдвиг»; активен в течение одного периода тактового сигнала.
- ready – не является управляющим сигналом для элементов фильтра, но сигнализирует об окончании обработки текущего набора $X[n]$ и готовности $Y[n]$. Активен с момента сброса сигнала resCE до поступления импульса frame.

Формализация зависимостей параметров элементов модуля от настраиваемых параметров

Возможность задания N , B и $C[n]$ предполагает описание элементов фильтра с использованием формул, определяющих зависимость от N , B и $C[n]$ внутренних параметров

фильтра, таких как разрядности элементов и шин, длительность управляющих сигналов и др. Исходными данными при выводе формул, помимо N , B и $C[n]$, являются диапазоны значений коэффициентов фильтра и входного сигнала, разрядности целой и дробной частей отсчетов входного сигнала (таблица 2).

Таблица 2. Зависимость параметров элементов модуля фильтра от настраиваемых параметров

Параметр	Обозначение
Исходные данные (настраиваемые параметры)	
Порядок фильтра	N
Разрядность входных отсчетов	B
Диапазон значений входных отсчетов	$X[n]_{min} \leq X[n] \leq X[n]_{max}$
Разрядности целой и дробной частей отсчетов входного сигнала	B_{int}, B_{fract}
Коэффициенты фильтра	$C[N-1] - C[0]$
Рассчитываемые параметры	
Количество бит ячеек ОЗУ, выделяемых под хранение целой части сумм $\sum_{n=0}^{N-1} C[n] \cdot X_b[n]$	$B_{RAM.int}$, определяется максимальным модулем суммы $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$ при $0 \leq b \leq B-1$
Разрядность ячеек ОЗУ	$B_{RAM} = B_{RAM.int} + B_{fract} + 1$
Разрядности входов сумматора (А, В)	$B_{RAM}, B_{RAM}+1$
Разрядность выхода сумматора (с учетом разряда S0)	$B_{RAM}+1$
Разрядность выхода регистра tempRG	$B_{RAM}+1$
Разрядность выхода регистра tempSRG	B
Исходное значение, хранящееся в tempRG	2^{B-1}
Номер младшего разряда $Y[n]$	$B - B_{RAM.int} - 1$
Номер старшего разряда $Y[n]$	$B + B_{fract}$
Количество состояний счетчика управляющего автомата	$B-2$

Начинать расчет следует с разрядности ОЗУ, B_{RAM} , т.к. остальные операции выполняются с результатами чтения из памяти. Разрядность дробной части хранимых в ОЗУ переменных принимается равной разрядности дробной части $X[n]$, B_{fract} . Разрядность ячеек ОЗУ, требуемая для представления целой части хранящихся там чисел, $B_{RAM.int}$, будет равна разрядности, требуемой для представления целой части максимального модуля суммы $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$ при $0 \leq b \leq B-1$. Еще один разряд резервируется для представления знака.

Определим разрядности входов и выходов сумматора. На вход А сумматора подается результат чтения ОЗУ разрядностью B_{RAM} (рисунок 3). Покажем, что для выходного сигнала сумматора и входа сумматора В достаточно разрядности $B_{RAM}+1$, чтобы предотвратить переполнение. Максимальное значение разрядностью B_{RAM} , считываемое из ОЗУ, будет равно 0111..11, где старший разряд является знаковым. Регистр tempRG, формирующий сигнал для входа В, добавляет к входному сигналу слева разряд, равный старшему разряду, поэтому максимальное значение разрядностью $B_{RAM}+1$ на входе В сумматора – 00111..11. При

суммировании этих значений на выходе сумматора получим 01111..10. Младший разряд этой суммы отправляется на вход регистра tempSRG, а к старшему в регистре tempRG добавляется 0 (т.к. старший разряд в данном случае равен нулю). Таким образом, на вход В сумматора снова подается 00111..11, и т.д. Минимальное значение разрядностью B_{RAM} , считываемое из ОЗУ, будет равно 1000..00, где старший разряд является знаковым. Минимальное значение, формируемое tempRG, будет равно 11000..00. В результате суммирования получим 10000..00; после обработки в tempRG на вход В будет подано значение 11000..00, и т.д. Отдельно следует рассмотреть работу схемы при $b=0$, когда к сумме $\sum_{n=0}^{N-1} C[n] \cdot x_0[n]$ прибавляется 2^{B-1} . Сумматор

сработает верно только при условии $B \leq B_{RAM}$, в противном случае сдвинутая на $B-1$ разрядов единица окажется на месте знакового разряда входа В сумматора или выйдет за пределы разрядной сетки.

Разрядность выхода временного сдвигового регистра tempSRG (рисунок 3) равна количеству выполняемых им операций сдвига, т.е. разрядности $X[n]$ В.

Разрядность входа регистра ResRG равна $B+B_{RAM}+1$. Для формирования $Y[n]$ принятый в ResRG результат должен быть сдвинут влево на разрядность целой части чисел, поступающих от tempRG – $B_{RAM.int}+1$. Таким образом, номер старшего разряда $Y[n]$ можно определить как $B+B_{RAM} - B_{RAM.int} - 1$, или $B+B_{fract}$. Номер младшего разряда $Y[n]$ при условии равенства B_{fract} разрядности дробной части $Y[n]$ определяется как $B - B_{RAM.int} - 1$.

Схема управляющего устройства приведена на рисунке 4. Управляющий автомат имеет 6 состояний. Состояние s0: ready=1; s1: iCE=1; s2: isrgCE=1, ramCE=1; s3: isrgCE=1, ramCE=1, trgCE=1; s4: ramCE=1, trgCE=1; s5: trgCE = 1, sign=1; s6: resCE=1.

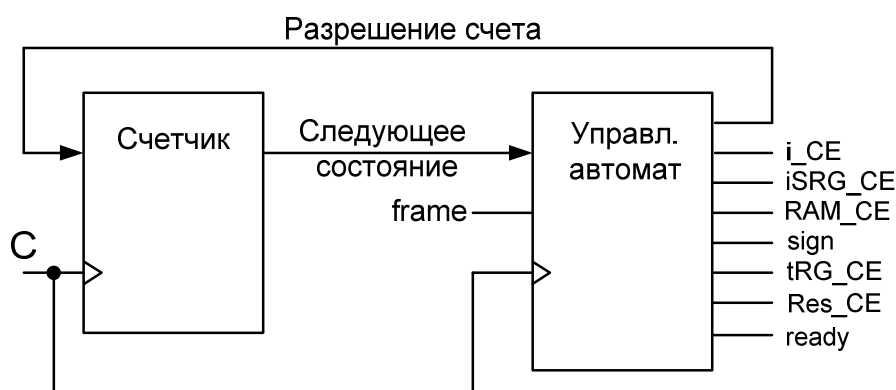


Рисунок 4. Управляющее устройство

Во всех состояниях, кроме s0 и s3, управляющий автомат пребывает по одному такту. Из состояния s0 в состояние s1 автомат переходит по приходу импульса frame. Длительность пребывания в состоянии s3 зависит от разрядности $X[n]$ и составляет $B-2$ такта. Для своевременного перехода автомата из состояния s3 в состояние s4 в управляющем устройстве предусматривается счетчик с настраиваемым числом состояний, который спустя $B-2$ такта от

начала работы формирует для автомата соответствующий сигнал (рисунок 4). Управляющий автомат, перейдя в стояние s3, формирует для счетчика сигнал разрешения счета.

Описание модуля на языке VHDL и функциональное моделирование модуля

Описание модуля на языке VHDL было выполнено в среде WebPACK ISE фирмы Xilinx. На нижнем уровне иерархии проекта модуля находятся описания отдельных элементов фильтра с переменной разрядностью; на верхнем уровне – указание настраиваемых параметров, а также используемых элементов и описание их взаимодействия. Ключевым словом языка VHDL, позволяющим ввести в описание настраиваемый параметр, является generic [2].

Временная диаграмма работы фильтра при $N=3$ и $B=8$ для значений $X[2]=0,125$; $X[1]=0,688$; $X[0]=-3,500$; $C[2]= -2,503$; $C[1]= -0,615$; $C[0]=3,181$; $Y[2]= -11,898$, полученная в результате функционального моделирования схемы, представлена на рисунке 5.

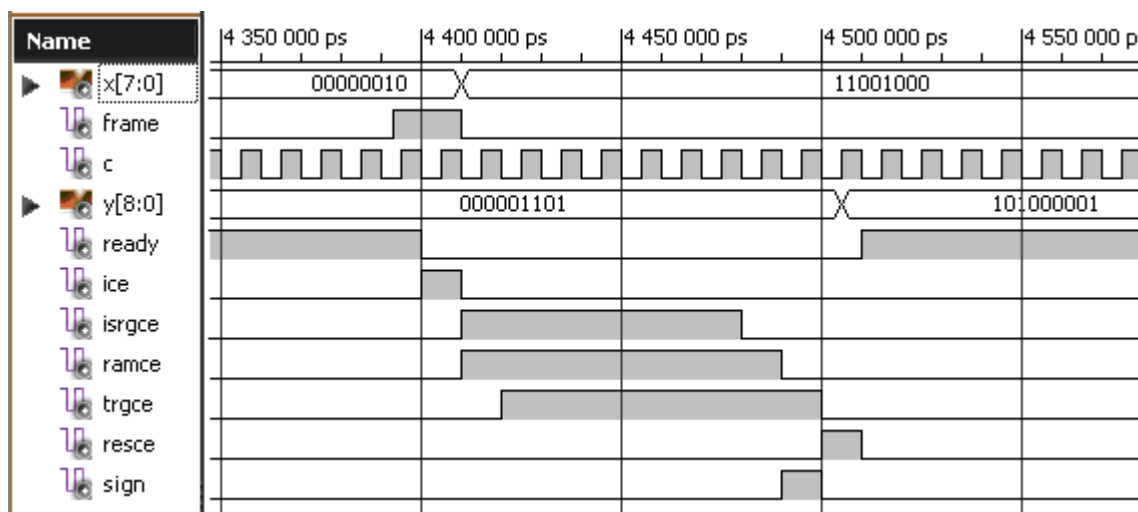


Рисунок 5. Результат моделирования фильтра при $N=3$ и $B=8$

Обсуждение полученных результатов

Согласно поставленным задачам, авторами получены структура распределенного КИХ-фильтра с настраиваемыми порядком, разрядностью входных отсчетов, коэффициентами; формализованы зависимости внутренних параметров фильтра от заданных настроек. Выполнено параметрическое описание фильтра на языке VHDL, работоспособность полученного описания подтверждается результатами функционального моделирования. Полученное описание пригодно для реализации на программируемых ресурсах (CLB) и блочном ОЗУ FPGA.

Наиболее существенным недостатком рассматриваемой структуры фильтра является увеличение потребности в ресурсах памяти пропорционально 2^N [2]. Объем памяти, требуемый для реализации фильтра порядка N при разрядности B_{RAM} ячейки ОЗУ, находится из соотношения $B_{RAM} \cdot 2^N$. Решением этой проблемы является разбиение ОЗУ фильтра на m блоков [2]; при этом разрядность адресной шины для каждого блока будет равна N/m , а

объем памяти для реализации фильтра порядка N будет рассчитываться из соотношения: $m \cdot B_{RAM} \cdot 2^{N/m}$. Структура фильтра при этом дополнится сумматорами, складывающими частные суммы вида $\sum_{n=0}^{N-1} C[n] \cdot x_b[n]$ перед дальнейшей обработкой.

Некоторой экономии ресурсов кристалла можно добиться, исключив из модуля регистры iRG и объединив в общую цепочку регистры iSRG. При этом усложнится структура управляющего автомата.

Еще одним ограничением использования фильтра рассматриваемой структуры, реализованного на программируемых ресурсах, может стать максимально допустимая тактовая частота. Она зависит от задержек распространения сигналов в элементах FPGA, на которых реализуется фильтр, а также задержек распространения по соединениям между элементами. Влияние этих факторов частично компенсируется тем, что все элементы фильтра, кроме сумматора и инвертора, являются синхронными. Однако при реализации фильтров высоких порядков расстояния между взаимодействующими элементами и разветвление по выходу у источника тактового сигнала возрастают, увеличивая задержки распространения.

Заключение

Главными преимуществами предлагаемого модуля являются независимость латентности от порядка фильтра и возможность настройки его параметров. Первое делает модуль предпочтительным в приложениях, где латентность является критичным параметром; второе позволяет рассматривать его как готовый элемент, пригодный к использованию в различных проектах. Предлагаемый модуль с настраиваемыми параметрами и его реализацию на FPGA в различных вариациях также можно рассматривать как экспериментальный базис при проектировании заказных микросхем и систем на кристалле. Исходя из требований к порядку и разрядности фильтра, с помощью такого базиса можно оптимизировать структуру и описание фильтра, определить перечень и объем ресурсов, необходимых для его реализации.

** Работа выполнена при поддержке гранта Министерства образования Пензенской области (приказ Министерства образования Пензенской области № 645/01-07 от 11.12.2012).*

Список литературы

1. Logic array blocks and adaptive logic modules in Cyclone V devices [Электронный ресурс]. – Режим доступа: http://www.altera.com/literature/hb/cyclone-v/cv_52001.pdf (дата обращения: 10.06.2013).

2. Meyer-Baese U. Digital Signal Processing with Field Programmable Gate Arrays. – Springer, 2001. – 422 p.
3. Spartan-6 FPGA Configurable Logic Block User Guide [Электронный ресурс]. – Режим доступа: http://www.xilinx.com/support/documentation/user_guides/ug384.pdf (дата обращения: 10.06.2013).
4. Spartan-6 FPGA DSP48A1 Slice User Guide [Электронный ресурс]. – Режим доступа: http://www.xilinx.com/support/documentation/user_guides/ug389.pdf (дата обращения: 10.06.2013).
5. Variable precision DSP blocks in Cyclone V devices [Электронный ресурс]. – Режим доступа: http://www.altera.com/literature/hb/cyclone-v/cv_52003.pdf (дата обращения: 10.06.2013).

Рецензенты:

Бабич М. Ю., д-р техн. наук, главный специалист ОАО «НПП «Рубин», г. Пенза.

Светлов А. В., д-р техн. наук, профессор, заведующий кафедрой «Радиотехника и радиоэлектронные системы» ФБГОУ ВПО «Пензенский государственный университет», г. Пенза.