

Продолжение. Начало в № 12 `2008

Валерий ЗОТОВ  
walerry@km.ru

### Формирование описаний сумматоров и вычитающих устройств, реализуемых на базе аппаратных секций DSP48E в ПЛИС FPGA серии Virtex-5, с помощью «мастера» Architecture Wizard САПР серии Xilinx ISE

Чтобы сформировать описание сумматора или вычитающего устройства, реализуемого на базе аппаратных секций ЦОС DSP48E в ПЛИС FPGA семейств Virtex-5 LX, Virtex-5 LXT, Virtex-5 SXT и Virtex-5 FXT [2, 5], следует в списке генерируемых элементов, отображаемом во второй диалоговой панели «мастера» Architecture Wizard с заголовком *Selection* (рис. 6) (КиТ № 12 `2008, стр. 34), выбрать строку *Adder/Subtractor*. Для генерации соответствующего VHDL-кода в среде управляющей оболочки САПР серии Xilinx ISE [3, 4] *Навигатора проекта (Project Navigator)* следует в диалоговой панели *Select IP*, показанной на рис. 8 (КиТ № 12 `2008, стр. 35), вы-

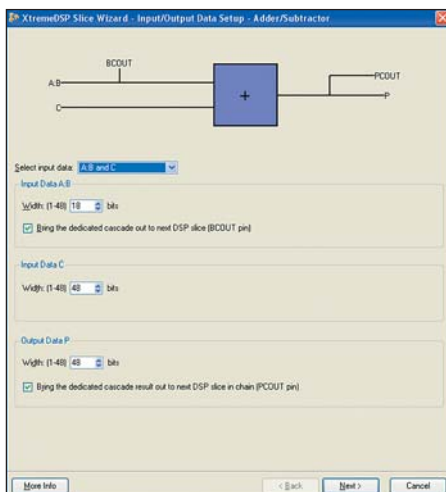


Рис. 17. Вид стартовой диалоговой панели Input/Output Data Setup – Adder/Subtractor

## Разработка компонентов устройств ЦОС, реализуемых на базе аппаратных модулей DSP48E в ПЛИС FPGA серии Virtex-5, с помощью «мастера» Architecture Wizard САПР серии Xilinx ISE

брать параметризованный модуль *Adder/Subtractor v9.1i*. В результате выполнения указанных действий на экране появляется стартовая диалоговая панель «мастера» настройки параметров создаваемого сумматора или вычитающего устройства с заголовком *Input/Output Data Setup – Adder/Subtractor*, которая позволяет выбрать источники входных данных и определить разрядность операндов и результата в формируемом элементе. Вид данной диалоговой панели приведен на рис. 17.

В первую очередь в диалоговой панели *Input/Output Data Setup – Adder/Subtractor* нужно указать входные шины данных, совокупность сигналов которых будет определять значения операндов в создаваемом элементе. Для этой цели следует воспользоваться полем выбора *Select input data*, выпадающий список которого содержит пять вариантов комбинации входных шин данных, которые могут использоваться в формируемых сумматорах и вычитающих устройствах: *A:B and C*, *BCIN and C*, *A:B and PCIN*, *BCIN and PCIN* и *C and PCIN*. По умолчанию в этом поле выбора предлагается вариант *A:B and C*, при использовании которого значение первого операнда определяется совокупностью сигналов на шине A:B, образованной путем объединения входных шин данных A и B, а второго — двоичным кодом, представленным на дополнительной входной шине данных C арифметическо-логического блока аппаратной секции DSP48E. В случае выбора варианта *BCIN and C* источниками операндов являются шина данных BCIN, предназначенная для подключения выходной шины BCOUT предшествующей секции DSP48E при каскадном соединении аппаратных модулей ЦОС, и дополнительная входная шина данных C арифметическо-логического блока. Если в поле выбора *Select input data* указывается вариант *A:B and PCIN*, то значения операндов в формируемом элементе поступают с шины A:B, образованной путем объединения входных шин данных A и B, и ши-

ны PCIN, предназначенной для подключения выходной шины PCOUT предыдущей секции DSP48E при каскадном наращивании аппаратных модулей ЦОС. При выборе варианта *BCIN and PCIN* входными портами данных формируемого сумматора или вычитающего устройства являются одноименные входные шины, используемые при каскадном соединении аппаратных секций ЦОС. Когда выбирается вариант *C and PCIN*, значения операндов определяются двоичными кодами, представленными на дополнительной входной шине данных C арифметическо-логического блока и шине PCIN, предназначенной для подключения выходной шины PCOUT предыдущей секции DSP48E при каскадном наращивании аппаратных модулей ЦОС.

В тех случаях, когда в качестве источника входных данных создаваемого элемента выбрана шина A:B, полученная в результате объединения входных шин данных A и B и/или дополнительная входная шина данных C арифметическо-логического блока, разработчику предоставляется возможность определения разрядности этих шин. Выбор требуемых значений разрядности входных операндов осуществляется с помощью полей редактирования *Width*, которые представлены во встроженных панелях *Input Data Bus1* и *Input Data Bus2*, где *Bus1* и *Bus2* — идентификаторы соответствующих входных шин (рис. 17). Допустимый диапазон значений разрядности входных шин данных A, B и C составляет от 1 до 48 бит. По умолчанию в этих полях редактирования предлагается значение, равное 18 бит. Если в формируемом сумматоре или вычитающем устройстве в качестве источников входных операндов задействованы шины PCIN и/или BCIN, то значения параметра *Width* жестко зафиксированы и составляют 48 и 18 двоичных разрядов соответственно.

При использовании входных шин данных A:B и BCIN в состав интерфейса формируемого элемента может быть добавлена выход-

ная шина BCOUT, которая предназначена для каскадного сопряжения аппаратных модулей ЦОС. Чтобы включить в состав интерфейса выходную шину данных BCOUT, предназначенную для подключения к входной шине BCIN следующей секции DSP48E, нужно оставить индикатор *Bring the dedicated cascade out to next DSP slice (BCOUT pin)*, расположенный в соответствующей встроенной панели *Input Data Bus1* или *Input Data Bus2*, в состоянии «Включено», предлагаемом по умолчанию. В тех случаях, когда не предполагается каскадное соединение создаваемых элементов, присутствие выходной шины данных BCOUT необязательно. Для исключения этой шины из состава интерфейса генерируемого сумматора или вычитающего устройства следует перевести индикатор *Bring the dedicated cascade out to next DSP slice (BCOUT pin)* в сброшенное состояние.

Далее необходимо определить разрядность выходной шины данных формируемого элемента, воспользовавшись полем редактирования *Width*, которое находится во встроенной панели *Output Data P* (рис. 17). В этом поле редактирования нужно указать требуемое значение разрядности в пределах от 1 до 48 бит. Во встроенной панели *Output Data P* также находится индикатор состояния *Bring the dedicated cascade result out to next DSP Slice in chain (PCOUT pin)*, с помощью которого осуществляется управление включением выходной шины PCOUT в состав интерфейса разрабатываемых сумматоров и вычитающих устройств. По умолчанию эта шина, предназначенная для подключения к входной шине PCIN следующего аппаратного модуля ЦОС при каскадном соединении, добавляется в состав интерфейса формируемых элементов. Чтобы заблокировать применение шины PCOUT в формируемом элементе, следует перевести индикатор *Bring the dedicated cascade result out to next DSP Slice in chain (PCOUT pin)* в состояние «Выключено».

После выбора конфигурации и требуемых значений параметров интерфейсных портов в генерируемом элементе, следует нажатием клавиши *Далее (Next)* в стартовой диалоговой панели *Input/Output Data Setup – Adder/Subtractor* перейти к очередной диалоговой панели «мастера» настройки параметров сумматора/вычитающего устройства, реализуемого на базе аппаратной секции DSP48E в кристаллах семейств Virtex-5 LX, Virtex-5 LXT, Virtex-5 SXT и Virtex-5 FXT. Вид второй диалоговой панели с заголовком *Operation Mode Setup – Adder/Subtractor*, предназначенной для выбора типа создаваемого элемента представлен на рис. 18.

Тип формируемого элемента указывается с помощью группы кнопок с зависимой фиксацией *Add/Subtract control*. Чтобы сгенерировать описание сумматора, следует нажать кнопку *Add*. Для создания описания вычитающего устройства необходимо нажать кнопку *Subtract*. При необходимости формирова-

ния элемента с динамически изменяемым типом выполняемой операции (сложения или вычитания) нужно нажать кнопку *Dynamic control of the ALUMODE*.

Выбрав тип выполняемой операции для создаваемого элемента, следует определить возможность применения сигнала входного переноса, воспользовавшись индикатором состояния *Use a CARRYIN*, который находится во встроенной панели *Carry option* (рис. 18). Чтобы в составе интерфейса генерируемого сумматора или вычитающего устройства присутствовал порт сигнала входного переноса, следует оставить этот индикатор в состоянии «Включено», предлагаемом по умолчанию. Если операции сложения или вычитания должны осуществляться без учета входного переноса, то нужно переключить индикатор *Use a CARRYIN* в сброшенное состояние. При этом незадействованный вход переноса аппаратной секции DSP48E, используемой для реализации формируемого сумматора или вычитающего устройства, подключается к общей шине.

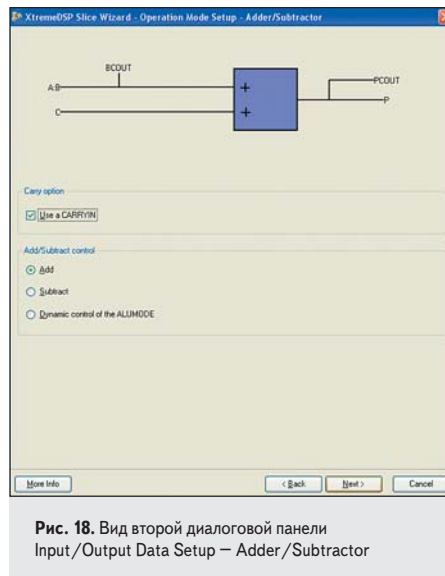


Рис. 18. Вид второй диалоговой панели *Input/Output Data Setup – Adder/Subtractor*

Определение типа создаваемого элемента и возможности использования входного переноса при выполнении соответствующей операции завершается нажатием клавиши *Далее (Next)* в диалоговой панели *Operation Mode Setup – Adder/Subtractor*. В результате этого открывается третья диалоговая панель «мастера» настройки параметров сумматора/вычитающего устройства, выполняемого на основе аппаратной секции DSP48E в ПЛИС семейств Virtex-5 LX, Virtex-5 LXT, Virtex-5 SXT и Virtex-5 FXT. Данная панель, имеющая заголовок *Pipelining and CE/RST Setup – Adder/Subtractor*, позволяет включить в состав генерируемого элемента входные и выходные регистры, а также добавить в состав интерфейса соответствующие входы управления для этих регистров. Вид третьей диалоговой панели «мастера» настройки параметров сумматоров и вычитающих устройств при отсутствии указанных регистров показан на рис. 19.

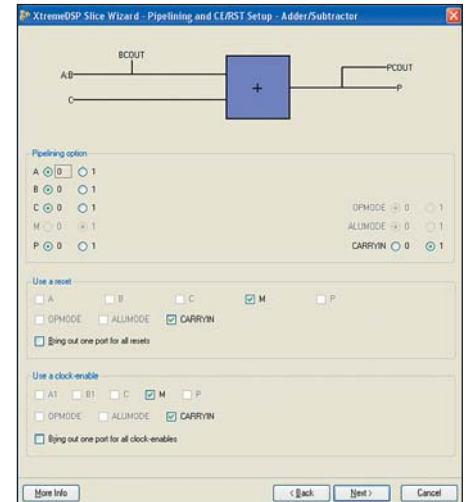


Рис. 19. Вид третьей диалоговой панели *Pipelining and CE/RST Setup – Adder/Subtractor*

В этой диалоговой панели, в первую очередь, рекомендуется выбрать входные и выходные регистры, которые необходимо включить в состав структуры генерируемого сумматора или вычитающего устройства. Для этой цели следует воспользоваться группами кнопок с зависимой фиксацией, представленными во встроенной панели *Pipelining option*. Количество доступных кнопок, расположенных в этой встроенной панели, зависит от источника входных данных, указанного в поле выбора *Select input data* в стартовой диалоговой панели *Input/Output Data Setup – Adder/Subtractor*, и типа формируемого элемента, выбранного с помощью группы кнопок *Add/Subtract control*, находящейся во второй диалоговой панели *Operation Mode Setup – Adder/Subtractor*. При этом разработчику предоставляется возможность выбора различных комбинаций входных и выходных регистров.

Если источником первого операнда в создаваемом элементе является шина A:B, полученная в результате объединения входных шин данных A и B, то наличие буферных регистров на этих шинах определяется состоянием групп кнопок с зависимой фиксацией A и B. Когда в нажатом положении находятся кнопки 0, входные регистры на указанных шинах не используются. Для установки входных регистров на шинах данных A и B нужно переключить в нажатое положение кнопки 1. При этом диалоговая панель *Pipelining and CE/RST Setup – Adder/Subtractor* автоматически преобразуется к виду, показанному на рис. 20. В этой панели становятся доступны индикаторы состояния, которые предназначены для выбора требуемых входов управления для буферных регистров, устанавливаемых на входных шинах данных A и B.

Когда в качестве источника входных данных в формируемом сумматоре или вычитающем устройстве выбирается дополнительная входная шина данных C арифметико-логического блока аппаратного модуля

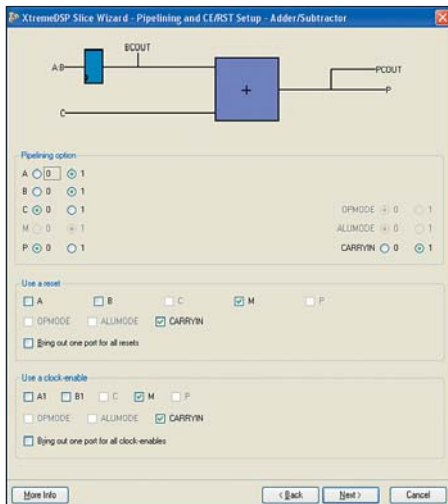


Рис. 20. Вид третьей диалоговой панели Pipelining and CE/RST Setup – Adder/Subtractor при буферизации сигналов одной входной шины данных

ЦОС, становится доступной группа кнопок C. Установка входного регистра на этой шине осуществляется переключением в нажатое состояние кнопки 1. В этом случае диалоговая панель *Pipelining and CE/RST Setup – Adder/Subtractor* приобретает вид, представленный на рис. 21.

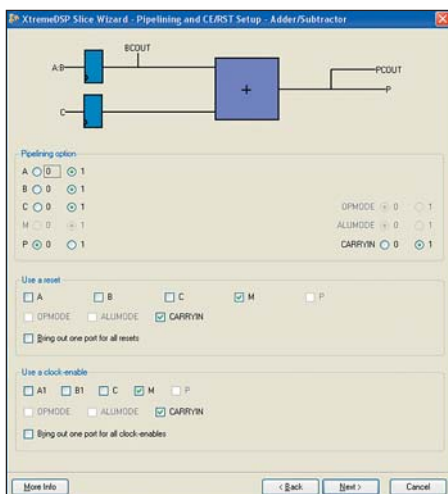


Рис. 21. Вид третьей диалоговой панели Pipelining and CE/RST Setup – Adder/Subtractor при буферизации сигналов каждой входной шины данных

При использовании в качестве источника первого операнда шины данных BCIN, предназначенной для каскадного сопряжения аппаратных модулей DSP48E, возможность применения входного регистра зависит от выбранного источника второго операнда. Если значение второго операнда определяется двоичным кодом, представленным на дополнительной входной шине данных C, то на первом входе генерируемого сумматора или вычитающего устройства можно установить конвейерный регистр, воспользовавшись группой кнопок B. В том случае, когда источниками операндов

в создаваемом элементе являются шины данных BCIN и PCIN, используемые при каскадном соединении аппаратных секций DSP48E, входные регистры не применяются.

Если входными портами создаваемого сумматора или вычитающего устройства являются шина A:B, образованная путем объединения входных шин A и B, и шина PCIN (в поле выбора *Select input data* указывается вариант *A:B and PCIN*), то разработчику предоставляется возможность выбора двухступенчатой схемы буферизации входных данных на шине A:B. При этом диалоговая панель *Pipelining and CE/RST Setup – Adder/Subtractor* принимает вид, показанный на рис. 22.

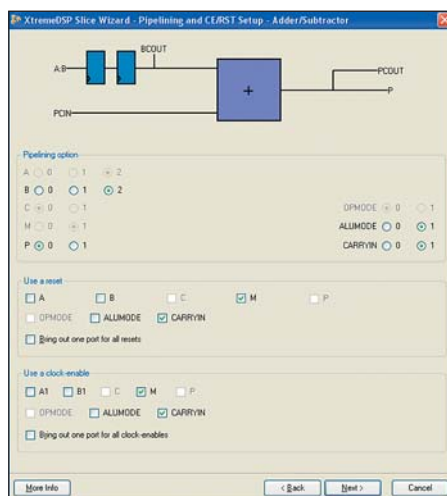


Рис. 22. Вид третьей диалоговой панели Pipelining and CE/RST Setup – Adder/Subtractor при выборе двухступенчатой схемы буферизации входных данных

Для того чтобы добавить в состав разрабатываемого сумматора или вычитающего устройства выходной регистр, следует воспользоваться группой кнопок с зависимой фиксацией P, находящейся во встроенной панели *Pipelining option*. Установка выходного регистра осуществляется при переключении в нажатое состояние кнопки 1, в результате чего диалоговая панель *Pipelining and CE/RST Setup – Adder/Subtractor* преобразуется к виду, изображенному на рис. 23.

Если при выполнении операций сложения или вычитания в генерируемом элементе используется значение входного переноса, то в цепи сигнала входного переноса также можно установить входной регистр, используя группу кнопок с зависимой фиксацией CARRYIN, расположенную во встроенной панели *Pipelining option* (рис. 19–23). При подготовке описаний комбинированных элементов с возможностью динамического выбора типа осуществляемой операции в процессе их функционирования можно задействовать буферные регистры на входах шины управления, которая предназначена для определения функции, выполняемой арифметическо-логическим блоком аппаратного модуля DSP48E.

Для этого следует зафиксировать в нажатом положении кнопку 1, которая входит в состав группы ALUMODE, представленной во встроенной панели *Pipelining option*.

Завершив выбор входных, выходных и буферных регистров, включаемых в структуру формируемого сумматора или вычитающего устройства, следует определить для каждого из них возможность использования входов сброса и разрешения синхронизации. Для включения в состав интерфейса разрабатываемого элемента входов сброса конвейерных регистров, указанных во встроенной панели *Pipelining option*, нужно установить в состояние «Включено» одноименные индикаторы, представленные во встроенной панели *Use a reset* (рис. 19–23). Если необходимо использовать один общий вход сброса для всех входных, выходных и буферных регистров, то следует переключить в активное состояние, отмеченное маркером, индикатор *Bring out one port for all resets*, который расположен в этой же встроенной панели.

Чтобы добавить в состав интерфейса генерируемого сумматора или вычитающего устройства входы сигналов разрешения синхронизации для используемых конвейерных регистров, нужно перевести в состояние «Включено» одноименные индикаторы состояния, расположенные во встроенной панели *Use a clock-enable* (рис. 19–23). При этом можно задействовать один общий сигнал разрешения синхронизации для всех входных, выходных и буферных регистров, включенных в структуру создаваемого элемента, установив индикатор *Bring out one port for all clock-enables* во включенное состояние.

После определения значений всех параметров конфигурации разрабатываемого компонента нужно с помощью клавиши *Далее (Next)*, расположенной в нижней части диалоговой панели *Operation Mode Setup – Adder/Subtractor*, перейти к заключительной информационной панели «мастера» настройки параметров сум-

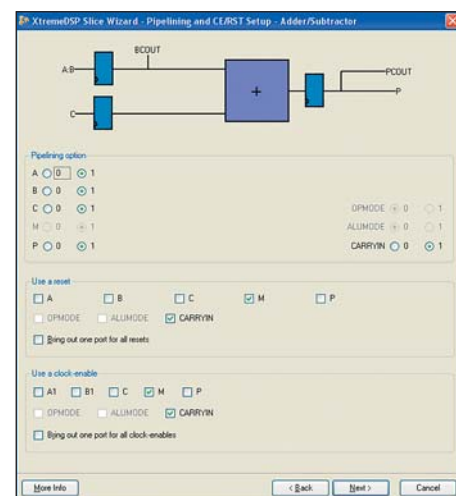


Рис. 23. Вид третьей диалоговой панели Pipelining and CE/RST Setup – Adder/Subtractor при установке выходного регистра

матора/вычитающего устройства, которая имеет заголовок *Summary – Adder/Subtractor*. Эта информационная панель содержит встроенные панели *Feature Summary*, *Files To Be Generated* и *Block Attributes*, в которых отображаются те же сведения, что и в информационной панели *Summary – Accumulator*, которая была рассмотрена в предыдущей части данной статьи [7]. Если все значения параметров конфигурации установлены корректно, то далее можно запустить процесс автоматической генерации файлов описания сумматора или вычитающего устройства, нажав кнопку «Готово» (*Finish*), расположенную в нижней части заключительной информационной панели «мастера». В последующих разделах приведены примеры описаний сумматора и вычитающего устройства, предназначенных для реализации на базе аппаратных секций DSP48E в ПЛИС FPGA серии Virtex-5, которые подготовлены с помощью «мастера» Architecture Wizard.

### Пример описания сумматора, сформированного с помощью «мастера» Architecture Wizard для реализации на базе аппаратных модулей DSP48E в ПЛИС FPGA серии Virtex-5

Примером описания сумматора, подготовленного с помощью «мастера» Architecture Wizard, является VHDL-описание элемента *adder\_18\_18\_24*, который предназначен для применения в составе проектов устройств ЦОС, реализуемых на базе кристаллов FPGA семейства Virtex-5 SXT. Сформированный сумматор выполняет операции сложения двух 18-разрядных значений входных данных с учетом входного переноса. Полученный результат отображается на 24-разрядной выходной шине данных. Значения слагаемых в элементе *adder\_18\_18\_24* определяются двоичным кодом, представленным на шине данных BCIN, предназначенной для подключения выходной шины BCOUT предшествующей секции DSP48E при каскадном соединении аппаратных модулей ЦОС, и дополнительной входной шиной данных C арифметическо-логического блока. В состав структуры сгенерированного сумматора включены входные и выходные регистры. Кроме того, буферный регистр задействован также в цепи сигнала входного переноса. Сформированный текст VHDL-описания сумматора *adder\_18\_18\_24* выглядит следующим образом:

```
--Command: xaw2vhdl-st D:\PRJ\New\adder_18_18_24.xaw
D:\PRJ\New\adder_18_18_24
--Design Name: adder_18_18_24
--Device: xc5vsx35t-ff665-3
--
-- Module adder_18_18_24
-- Generated by Xilinx Architecture Wizard
-- Written for synthesis tool: XST
--
library ieee;
use ieee.std_logic_1164.ALL;
use ieee.numeric_std.ALL;
library UNISIM;
```

```
use UNISIM.Vcomponents.ALL;
--
entity adder_18_18_24 is
  port (
    BCIN_IN          : in  std_logic_vector (17 downto 0);
    CARRYIN_IN       : in  std_logic;
    CEB1_IN          : in  std_logic;
    CEB2_IN          : in  std_logic;
    CECARRYIN_IN     : in  std_logic;
    CEC_IN           : in  std_logic;
    CEMULTCARRYIN_IN : in  std_logic;
    CEM_IN           : in  std_logic;
    CEP_IN           : in  std_logic;
    CLK_IN           : in  std_logic;
    C_IN             : in  std_logic_vector (17 downto 0);
    RSTALLCARRYIN_IN : in  std_logic;
    RSTB_IN          : in  std_logic;
    RSTC_IN          : in  std_logic;
    RSTM_IN          : in  std_logic;
    RSTP_IN          : in  std_logic;
    BCOUT_OUT        : out std_logic_vector (17 downto 0);
    PCOUT_OUT        : out std_logic_vector (47 downto 0);
    P_OUT            : out std_logic_vector (23 downto 0)
  );
end adder_18_18_24;
--
architecture BEHAVIORAL of adder_18_18_24 is
  signal GND_ALUMODE : std_logic;
  signal GND_BUS_3   : std_logic_vector (2 downto 0);
  signal GND_BUS_18  : std_logic_vector (17 downto 0);
  signal GND_BUS_30  : std_logic_vector (29 downto 0);
  signal GND_BUS_48  : std_logic_vector (47 downto 0);
  signal GND_OPMODE  : std_logic;
  signal P_float     : std_logic_vector (23 downto 0);
  signal VCC_OPMODE  : std_logic;
begin
  GND_ALUMODE <= '0';
  GND_BUS_3(2 downto 0) <= <000>;
  GND_BUS_18(17 downto 0) <= <00000000000000000000>;
  GND_BUS_30(29 downto 0) <= <00000000000000000000000000000000>;
  GND_BUS_48(47 downto 0) <=
    <0000000000000000000000000000000000000000000000000000000>;
  GND_OPMODE <= '0';
  VCC_OPMODE <= '1';
  DSP48E_INST : DSP48E
  generic map(
    ACASCREG => 1,
    ALUMODEREG => 0,
    AREG => 0,
    AUTORESET_PATTERN_DETECT => FALSE,
    AUTORESET_PATTERN_DETECT_OPTINV => "MATCH",
    A_INPUT => "DIRECT",
    BCASCREG => 1,
    BREG => 1,
    B_INPUT => "CASCADE",
    CARRYINREG => 1,
    CARRYINSELREG => 0,
    CREG => 1,
    MASK => x<3FFFFFFF>,
    MREG => 1,
    MULTCARRYINREG => 1,
    OPMODEREG => 0,
    PATTERN => x<00000000000000>,
    PREG => 1,
    SEL_MASK => "MASK",
    SEL_PATTERN => "PATTERN",
    SEL_ROUNDING_MASK => "SEL_MASK",
    USE_MULT => "NONE",
    USE_PATTERN_DETECT => "NO_PATDET",
    USE_SIMD => "ONE48"
  )
  port map (
    A(29 downto 0) => GND_BUS_30(29 downto 0),
    ACIN(29 downto 0) => GND_BUS_30(29 downto 0),
    ALUMODE(3) => GND_ALUMODE,
    ALUMODE(2) => GND_ALUMODE,
    ALUMODE(1) => GND_ALUMODE,
    ALUMODE(0) => GND_ALUMODE,
    B(17 downto 0) => GND_BUS_18(17 downto 0),
    BCIN(17 downto 0) => BCIN_IN(17 downto 0),
    C(47) => C_IN(17),
    C(46) => C_IN(17),
    C(45) => C_IN(17),
    C(44) => C_IN(17),
    C(43) => C_IN(17),
    C(42) => C_IN(17),
    C(41) => C_IN(17),
    C(40) => C_IN(17),
    C(39) => C_IN(17),
    C(38) => C_IN(17),
    C(37) => C_IN(17),
    C(36) => C_IN(17),
    C(35) => C_IN(17),
    C(34) => C_IN(17),
    C(33) => C_IN(17),
    C(32) => C_IN(17),
    C(31) => C_IN(17),
    C(30) => C_IN(17),
    C(29) => C_IN(17),
    C(28) => C_IN(17),
    C(27) => C_IN(17),
    C(26) => C_IN(17),
    C(25) => C_IN(17),
    C(24) => C_IN(17),
    C(23) => C_IN(17),
    C(22) => C_IN(17),
    C(21) => C_IN(17),
    C(20) => C_IN(17),
    C(19) => C_IN(17),
    C(18) => C_IN(17),
    C(17 downto 0) => C_IN(17 downto 0),
    CARRYASCIN => GND_ALUMODE,
    CARRYIN => CARRYIN_IN,
    CARRYINSEL(2 downto 0) => GND_BUS_3(2 downto 0),
    CEALUMODE => VCC_OPMODE,
    CEA1 => VCC_OPMODE,
    CEA2 => VCC_OPMODE,
    CEB1 => CEB1_IN,
    CEB2 => CEB2_IN,
    CEC => CEC_IN,
    CECARRYIN => CECARRYIN_IN,
    CECTRL => VCC_OPMODE,
    CEM => CEM_IN,
    CEMULTCARRYIN => CEMULTCARRYIN_IN,
    CEP => CEP_IN,
    CLK => CLK_IN,
    MULTSIGNIN => GND_ALUMODE,
    OPMODE(6) => GND_OPMODE,
    OPMODE(5) => VCC_OPMODE,
    OPMODE(4) => VCC_OPMODE,
    OPMODE(3) => GND_OPMODE,
    OPMODE(2) => GND_OPMODE,
    OPMODE(1) => VCC_OPMODE,
    OPMODE(0) => VCC_OPMODE,
    PCIN(47 downto 0) => GND_BUS_48(47 downto 0),
    RSTA => GND_ALUMODE,
    RSTALLCARRYIN => RSTALLCARRYIN_IN,
    RSTALUMODE => GND_ALUMODE,
    RSTB => RSTB_IN,
    RSTC => RSTC_IN,
    RSTCTRL => GND_ALUMODE,
    RSTM => RSTM_IN,
    RSTP => RSTP_IN,
    ACOUT => open,
    BCOUT(17 downto 0) => BCOUT_OUT(17 downto 0),
    CARRYASCOUT => open,
    CARRYOUT => open,
    MULTSIGNOUT => open,
    OVERFLOW => open,
    P(47 downto 24) => P_float(23 downto 0),
    P(23 downto 0) => P_OUT(23 downto 0),
    PATTERNDETECT => open,
    PATTERNDETECT => open,
    PCOUT(47 downto 0) => PCOUT_OUT(47 downto 0),
    UNDERFLOW => open
  );
end BEHAVIORAL;
```

Представленный текст содержит типовые разделы, соответствующие стандартной структуре VHDL-описания. В состав интерфейса сумматора *adder\_18\_18\_24* включены дополнительные выходные шины BCOUT и PCOUT, предоставляющие возможность каскадного наращивания разрядности. В сформированном элементе предусмотрены входы сброса и разрешения синхронизации входных и выходных регистров. Описание архитектуры сумматора *adder\_18\_18\_24* выполнено на базе примитива DSP48E, рассмотренного в первой части данной статьи [6].

В описании сформированного сумматора применяется следующая система условных обозначений интерфейсных портов:

- BCIN\_IN[17:0] — 18-разрядная входная шина данных, совокупность сигналов которой определяет значение первого слагаемого;
- C\_IN[17:0] — 18-разрядная входная шина данных, совокупность сигналов которой определяет значение второго слагаемого;
- CARRYIN\_IN — вход сигнала переноса;
- CEB1\_IN, CEB2\_IN — входы сигнала разрешения синхронизации для входных регистров, устанавливаемых на шине данных BCIN;

- CEC\_IN — вход сигнала разрешения синхронизации для входного регистра шины данных C;
- CECARRYIN\_IN — вход сигнала разрешения синхронизации для буферного регистра, устанавливаемого в цепи внешнего сигнала входного переноса;
- CEMULTCARRYIN\_IN — вход сигнала разрешения синхронизации для буферного регистра, устанавливаемого в цепи внутреннего сигнала входного переноса;
- CEM\_IN — вход сигнала разрешения синхронизации для буферного регистра, устанавливаемого на входе арифметическо-логического блока;
- CEP\_IN — вход сигнала разрешения синхронизации для выходного регистра;
- CLK\_IN — вход тактового сигнала;
- RSTALLCARRYIN\_IN — вход сигнала сброса буферных регистров, применяемых в цепи входного переноса;
- RSTB\_IN и RSTC\_IN — входы сигналов сброса буферных регистров, устанавливаемых на входных шинах данных BCIN и C соответственно;
- RSTM\_IN — вход сигнала сброса буферного регистра, устанавливаемого на входе арифметическо-логического блока;
- RSTP\_IN — вход сигнала сброса выходного регистра;
- BCOUT\_OUT [17:0] — 18-разрядная выходная шина, предназначенная для каскадного соединения сумматоров;
- PCOUT\_OUT — 48-разрядная выходная шина данных, используемая при каскадном соединении сумматоров;
- P\_OUT — выходная 24-разрядная шина данных, представляющая значение вычисленной суммы.

Совокупность выражений, определяющих требуемые значения всех необходимых атрибутов библиотечного примитива DSP48E для сумматора *adder\_18\_18\_24*, которые представлены в файле с расширением *ucf*, генерируемом «мастером» Architecture Wizard в дополнение к VHDL-описанию, имеет следующий вид:

```
# Generated by Xilinx Architecture Wizard
# --- UCF Template Only ---
# Cut and paste these attributes into the project's UCF file, if desired
INST DSP48E_INST ACASCREG = 1;
INST DSP48E_INST ALUMODEREG = 0;
INST DSP48E_INST AREG = 0;
INST DSP48E_INST AUTORESET_PATTERN_DETECT = FALSE;
INST DSP48E_INST AUTORESET_PATTERN_DETECT_OPTINV = MATCH;
INST DSP48E_INST A_INPUT = DIRECT;
INST DSP48E_INST BCASCREG = 1;
INST DSP48E_INST BREG = 1;
INST DSP48E_INST B_INPUT = CASCADE;
INST DSP48E_INST CARRYINREG = 1;
INST DSP48E_INST CARRYINSELREG = 0;
INST DSP48E_INST CREG = 1;
INST DSP48E_INST MASK = 3FFFFFFF;
INST DSP48E_INST MREG = 1;
INST DSP48E_INST MULTCARRYINREG = 1;
INST DSP48E_INST OPMODEREG = 0;
INST DSP48E_INST PATTERN = 000000000000;
INST DSP48E_INST PREG = 1;
INST DSP48E_INST SEL_MASK = MASK;
INST DSP48E_INST SEL_PATTERN = PATTERN;
INST DSP48E_INST SEL_ROUNDING_MASK = SEL_MASK;
INST DSP48E_INST USE_MULT = NONE;
INST DSP48E_INST USE_PATTERN_DETECT = NO_PATDET;
INST DSP48E_INST USE_SIMD = ONE48;
```

### Пример описания вычитающего устройства, сформированного с помощью «мастера» Architecture Wizard для реализации на базе аппаратных модулей DSP48E в ПЛИС FPGA семейств Virtex-5 LX, Virtex-5 LXT, Virtex-5 SXT и Virtex-5 FXT

В качестве примера вычитающего устройства, реализуемого на базе аппаратных модулей DSP48E в ПЛИС FPGA семейств Virtex-5 LX, Virtex-5 LXT, Virtex-5 SXT и Virtex-5 FXT, в настоящем разделе приводится описание элемента *subtractor\_48\_48\_48*, предназначенного для вычисления разности двух 48-разрядных операндов с учетом значения входного переноса (заема). Источниками операндов в сгенерированном вычитающем устройстве являются шина A:B, образованная путем объединения входных шин данных A и B, и шина PCIN, предназначенная для подключения выходной шины PCOUT предыдущей секции DSP48E при каскадном соединении аппаратных модулей ЦОС. В составе архитектуры элемента *subtractor\_48\_48\_48* предусмотрены входные регистры на шинах данных A и B, цепи сигнала входного переноса (заема) и выходной регистр. Во всех входных, выходных и буферных регистрах сформированного вычитающего устройства задействованы входы сброса и разрешения синхронизации, которые представлены в составе интерфейса элемента *subtractor\_48\_48\_48* в виде соответствующих портов сигналов управления. В состав архитектуры данного элемента включены дополнительные шины данных, обеспечивающие возможность каскадного наращивания. Текст VHDL-описания вычитающего устройства *subtractor\_48\_48\_48*, сформированный с помощью «мастера» Architecture Wizard, выглядит следующим образом:

```
--Command: xaw2vhdl-st D:\PRJ\New\subtractor_48_48_48.xaw
D:\PRJ\New\subtractor_48_48_48
--Design Name: subtractor_48_48_48
--Device: xc5vsx35t-ff665-3
--
-- Module subtractor_48_48_48
-- Generated by Xilinx Architecture Wizard
-- Written for synthesis tool: XST
--
library ieee;
use ieee.std_logic_1164.ALL;
use ieee.numeric_std.ALL;
library UNISIM;
use UNISIM.vcomponents.ALL;
--
entity subtractor_48_48_48 is
    port (
        AB_IN          : in std_logic_vector (47 downto 0);
        CARRYIN_IN     : in std_logic;
        CE1_IN         : in std_logic;
        CE2_IN         : in std_logic;
        CEMULTCARRYIN_IN : in std_logic;
        CEM_IN         : in std_logic;
        CEP_IN         : in std_logic;
        CLK_IN         : in std_logic;
        PCIN_IN        : in std_logic_vector (47 downto 0);
        RSTALLCARRYIN_IN : in std_logic;
        RSTA_IN        : in std_logic;
        RSTB_IN        : in std_logic;
        RSTM_IN        : in std_logic;
        RSTP_IN        : in std_logic;
        BCOUT_OUT      : out std_logic_vector (17 downto 0);
        PCOUT_OUT      : out std_logic_vector (47 downto 0);
        P_OUT          : out std_logic_vector (47 downto 0)
    );
```

```
end subtractor_48_48_48;
--
architecture BEHAVIORAL of subtractor_48_48_48 is
    signal GND_ALUMODE         : std_logic;
    signal GND_BUS_3          : std_logic_vector (2 downto 0);
    signal GND_BUS_18         : std_logic_vector (17 downto 0);
    signal GND_BUS_30         : std_logic_vector (29 downto 0);
    signal GND_BUS_48         : std_logic_vector (47 downto 0);
    signal GND_OPMODE         : std_logic;
    signal VCC_ALUMODE        : std_logic;
    signal VCC_OPMODE         : std_logic;
begin
    GND_ALUMODE <= '0';
    GND_BUS_3(2 downto 0) <= <000>;
    GND_BUS_18(17 downto 0) <= <000000000000000000000000>;
    GND_BUS_30(29 downto 0) <= <00000000000000000000000000000000>;
    GND_BUS_48(47 downto 0) <=
        <0000000000000000000000000000000000000000000000000000000>;
    GND_OPMODE <= '0';
    VCC_ALUMODE <= '1';
    VCC_OPMODE <= '1';
    DSP48E_INST : DSP48E
    generic map (
        ACASCREG => 1,
        ALUMODEREG => 0,
        AREG => 2,
        AUTORESET_PATTERN_DETECT => FALSE,
        AUTORESET_PATTERN_DETECT_OPTINV => "MATCH",
        A_INPUT => "DIRECT",
        BCASCREG => 1,
        BREG => 2,
        B_INPUT => "DIRECT",
        CARRYINREG => 1,
        CARRYINSELREG => 0,
        CREG => 0,
        MASK => x<3FFFFFFF>,
        MREG => 1,
        MULTCARRYINREG => 1,
        OPMODEREG => 0,
        PATTERN => x<000000000000>,
        PREG => 1,
        SEL_MASK => "MASK",
        SEL_PATTERN => "PATTERN",
        SEL_ROUNDING_MASK => "SEL_MASK",
        USE_MULT => "NONE",
        USE_PATTERN_DETECT => "NO_PATDET",
        USE_SIMD => "ONE48"
    )
    port map (
        A(29 downto 0) => AB_IN(47 downto 18),
        ACIN(29 downto 0) => GND_BUS_30(29 downto 0),
        ALUMODE(3) => GND_ALUMODE,
        ALUMODE(2) => GND_ALUMODE,
        ALUMODE(1) => VCC_ALUMODE,
        ALUMODE(0) => VCC_ALUMODE,
        B(17 downto 0) => AB_IN(17 downto 0),
        BCIN(17 downto 0) => GND_BUS_18(17 downto 0),
        C(47 downto 0) => GND_BUS_48(47 downto 0),
        CARRYCASCIN => GND_ALUMODE,
        CARRYIN => CARRYIN_IN,
        CARRYINSEL(2 downto 0) => GND_BUS_3(2 downto 0),
        CEALUMODE => VCC_ALUMODE,
        CE1 => CE1_IN,
        CE2 => CE2_IN,
        CEC => VCC_ALUMODE,
        CECTRL => VCC_ALUMODE,
        CEM => CEM_IN,
        CEMULTCARRYIN => CEMULTCARRYIN_IN,
        CEP => CEP_IN,
        CLK => CLK_IN,
        MULTSIGNIN => GND_ALUMODE,
        OPMODE(6) => GND_OPMODE,
        OPMODE(5) => GND_OPMODE,
        OPMODE(4) => VCC_OPMODE,
        OPMODE(3) => GND_OPMODE,
        OPMODE(2) => GND_OPMODE,
        OPMODE(1) => VCC_OPMODE,
        OPMODE(0) => VCC_OPMODE,
        PCIN(47 downto 0) => PCIN_IN(47 downto 0),
        RSTA => RSTA_IN,
        RSTALLCARRYIN => RSTALLCARRYIN_IN,
        RSTALUMODE => GND_ALUMODE,
        RSTB => RSTB_IN,
        RSTC => GND_ALUMODE,
        RSTCTRL => GND_ALUMODE,
        RSTM => RSTM_IN,
        RSTP => RSTP_IN,
        ACOUT => open,
        BCOUT(17 downto 0) => BCOUT_OUT(17 downto 0),
        CARRYCASCOUT => open,
        CARRYOUT => open,
        MULTSIGNOUT => open,
        OVERFLOW => open,
        P(47 downto 0) => P_OUT(47 downto 0),
        PATTERNBDETECT => open,
        PATTERNDETECT => open,
        UNDERFLOW => open
    );
end BEHAVIORAL;
```

В системе условных обозначений входных и выходных портов вычитающего устройства *subtractor\_48\_48\_48* в большинстве своем используются те же идентификаторы, что и в описании интерфейса сумматора *adder\_18\_18\_24*, который был рассмотрен в предыдущем разделе. Дополнительно в составе интерфейса элемента *subtractor\_48\_48\_48* представлены следующие входные порты:

- **AB\_IN [47:0]** — 48-разрядная входная шина данных, совокупность сигналов которой определяет значение первого операнда;
- **PCIN\_IN[47:0]** — 48-разрядная входная шина данных, совокупность сигналов которой определяет значение второго операнда;
- **CEA1\_IN** — вход сигнала разрешения синхронизации для буферного регистра, установленного на входной шине данных A;
- **RSTA\_IN** — вход сигнала сброса буферного регистра, установленного на входной шине данных A.

При включении VHDL-описания вычитающего устройства *subtractor\_48\_48\_48* в состав разрабатываемого проекта САПР серии Xilinx ISE необходимо добавить в файл временных и топологических ограничений следующую совокупность выражений, определяющих значения параметров библиотечного примитива DSP48E:

```
# Generated by Xilinx Architecture Wizard
# -- UCF Template Only --
# Cut and paste these attributes into the project's UCF file, if desired
INST DSP48E_INST ACASCREG = 1;
INST DSP48E_INST ALUMODEREG = 0;
INST DSP48E_INST AREG = 2;
INST DSP48E_INST AUTORESET_PATTERN_DETECT = FALSE;
INST DSP48E_INST AUTORESET_PATTERN_DETECT_OPTINV = MATCH;
INST DSP48E_INST A_INPUT = DIRECT;
INST DSP48E_INST BCASCREG = 1;
INST DSP48E_INST BREG = 2;
INST DSP48E_INST B_INPUT = DIRECT;
INST DSP48E_INST CARRYINREG = 1;
INST DSP48E_INST CARRYINSELREG = 0;
INST DSP48E_INST CREG = 0;
INST DSP48E_INST MASK = 3FFFFFFFFF;
INST DSP48E_INST MREG = 1;
INST DSP48E_INST MULTCARRYINREG = 1;
INST DSP48E_INST OPMODEREG = 0;
INST DSP48E_INST PATTERN = 000000000000;
INST DSP48E_INST PREG = 1;
INST DSP48E_INST SEL_MASK = MASK;
INST DSP48E_INST SEL_PATTERN = PATTERN;
INST DSP48E_INST SEL_ROUNDING_MASK = SEL_MASK;
INST DSP48E_INST USE_MULT = NONE;
INST DSP48E_INST USE_PATTERN_DETECT = NO_PATDET;
INST DSP48E_INST USE_SIMD = ONE48;
```

### Пример описания комбинированного устройства, сформированного с помощью «мастера» Architecture Wizard для реализации на базе аппаратных модулей DSP48E в ПЛИС FPGA семейств Virtex-5 SXT, Virtex-5 LXT и Virtex-5 FXT

Результат применения «мастера» Architecture Wizard для формирования описания комбинированного устройства, реализуемого на основе аппаратных секций DSP48E в ПЛИС FPGA серии Virtex-5, в котором тип выполняемой операции изменяется динамически в процессе его функционирования, демонст-

рируется в настоящем разделе на примере элемента *add\_sub\_18\_48\_48*. Данное устройство предназначено для выполнения операций сложения и вычитания над 18- и 48-разрядными операндами с учетом значения входного переноса (заема). Тип выполняемой операции задается соответствующим значением кода, подаваемого на входную шину управления **ALUMODE\_IN**. Значения операндов в сформированном элементе определяются двоичными кодами, представленными на входных шинах **BCIN** и **PCIN**, используемых при каскадном соединении аппаратных модулей ЦОС DSP48E. Результат выполнения операций сложения и вычитания отображается на выходах этого элемента в виде 48-разрядного двоичного кода. В структуру сумматора/вычитающего устройства *add\_sub\_18\_48\_48* включены буферные регистры, установленные в цепи сигнала входного переноса и на входной шине управления выбором выполняемой операции, а также выходной регистр. В составе интерфейса рассматриваемого элемента представлены входы сброса и разрешения синхронизации для каждого из этих регистров. Текст описания комбинированного устройства *add\_sub\_18\_48\_48* с возможностью выбора выполняемой операции, сгенерированный с помощью «мастера» Architecture Wizard, имеет следующий вид:

```
--Command: xaw2vhdl-st D:\PRJ\New\add_sub_18_48_48.xaw
D:\PRJ\New\add_sub_18_48_48
--Design Name: add_sub_18_48_48
--Device: xc5vx35t-ff665-3
--
-- Module add_sub_18_48_48
-- Generated by Xilinx Architecture Wizard
-- Written for synthesis tool: XST
--
library ieee;
use ieee.std_logic_1164.ALL;
use ieee.numeric_std.ALL;
library UNISIM;
use UNISIM.vcomponents.ALL;
--
entity add_sub_18_48_48 is
  port (
    ALUMODE_IN       : in std_logic_vector (3 downto 0);
    BCIN_IN          : in std_logic_vector (17 downto 0);
    CARRYIN_IN       : in std_logic;
    CEALUMODE_IN     : in std_logic;
    CECARRYIN_IN     : in std_logic;
    CEMULTCARRYIN_IN : in std_logic;
    CEM_IN           : in std_logic;
    CEP_IN           : in std_logic;
    CLK_IN           : in std_logic;
    PCIN_IN          : in std_logic_vector (47 downto 0);
    RSTALLCARRYIN_IN : in std_logic;
    RSTALUMODE_IN    : in std_logic;
    RSTCTRL_IN      : in std_logic;
    RSTM_IN          : in std_logic;
    RSTP_IN          : in std_logic;
    BCOUT_OUT        : out std_logic_vector (17 downto 0);
    PCOUT_OUT        : out std_logic_vector (47 downto 0);
    P_OUT            : out std_logic_vector (47 downto 0);
  );
end add_sub_18_48_48;
--
architecture BEHAVIORAL of add_sub_18_48_48 is
  signal GND_A      : std_logic_vector (28 downto 0);
  signal GND_BUS_3 : std_logic_vector (2 downto 0);
  signal GND_BUS_18 : std_logic_vector (17 downto 0);
  signal GND_BUS_30 : std_logic_vector (29 downto 0);
  signal GND_BUS_48 : std_logic_vector (47 downto 0);
  signal GND_OPMODE : std_logic;
  signal VCC_A      : std_logic;
  signal VCC_OPMODE : std_logic;
begin
  GND_A(28 downto 0) <= «00000000000000000000000000000000»;
  GND_BUS_3(2 downto 0) <= «000»;
  GND_BUS_18(17 downto 0) <= «00000000000000000000»;
  GND_BUS_30(29 downto 0) <= «00000000000000000000000000000000»;
  GND_BUS_48(47 downto 0) <= «00000000000000000000000000000000»;
  GND_OPMODE <= '0';
  VCC_A <= '1';
  VCC_OPMODE <= '1';
  DSP48E_INST : DSP48E
  generic map(
    ACASCREG => 1,
    ALUMODEREG => 1,
    AREG => 0,
    AUTORESET_PATTERN_DETECT => FALSE,
    AUTORESET_PATTERN_DETECT_OPTINV => "MATCH",
    A_INPUT => "DIRECT",
    BCASCREG => 1,
    BREG => 0,
    B_INPUT => "CASCADE",
    CARRYINREG => 1,
    CARRYINSELREG => 0,
    CREG => 0,
    MASK => x«3FFFFFFFFF»,
    MREG => 1,
    MULTCARRYINREG => 1,
    OPMODEREG => 0,
    PATTERN => x«00000000000000»,
    PREG => 1,
    SEL_MASK => "MASK",
    SEL_PATTERN => "PATTERN",
    SEL_ROUNDING_MASK => "SEL_MASK",
    USE_MULT => "MULT_S",
    USE_PATTERN_DETECT => "NO_PATDET",
    USE_SIMD => "ONE48"
  )
  port map (
    A(29 downto 1) => GND_A(28 downto 0),
    A(0) => VCC_A,
    ACIN(29 downto 0) => GND_BUS_30(29 downto 0),
    ALUMODE(3 downto 0) => ALUMODE_IN(3 downto 0),
    B(17 downto 0) => GND_BUS_18(17 downto 0),
    BCIN(17 downto 0) => BCIN_IN(17 downto 0),
    C(47 downto 0) => GND_BUS_48(47 downto 0),
    CARRYCASCIN => GND_OPMODE,
    CARRYIN => CARRYIN_IN,
    CARRYINSEL(2 downto 0) => GND_BUS_3(2 downto 0),
    CEALUMODE => CEALUMODE_IN,
    CEA1 => VCC_A,
    CEA2 => VCC_A,
    CEB1 => VCC_A,
    CEB2 => VCC_A,
    CEC => VCC_A,
    CECARRYIN => CECARRYIN_IN,
    CECTRL => VCC_A,
    CEM => CEM_IN,
    CEMULTCARRYIN => CEMULTCARRYIN_IN,
    CEP => CEP_IN,
    CLK => CLK_IN,
    MULTSIGNIN => GND_OPMODE,
    OPMODE(6) => GND_OPMODE,
    OPMODE(5) => GND_OPMODE,
    OPMODE(4) => VCC_OPMODE,
    OPMODE(3) => GND_OPMODE,
    OPMODE(2) => VCC_OPMODE,
    OPMODE(1) => GND_OPMODE,
    OPMODE(0) => VCC_OPMODE,
    PCIN(47 downto 0) => PCIN_IN(47 downto 0),
    RSTA => GND_OPMODE,
    RSTALLCARRYIN => RSTALLCARRYIN_IN,
    RSTALUMODE => RSTALUMODE_IN,
    RSTB => GND_OPMODE,
    RSTC => GND_OPMODE,
    RSTCTRL => RSTCTRL_IN,
    RSTM => RSTM_IN,
    RSTP => RSTP_IN,
    ACOUT => open,
    BCOUT(17 downto 0) => BCOUT_OUT(17 downto 0),
    CARRYASCOUT => open,
    CARRYOUT => open,
    MULTSIGNOUT => open,
    OVERFLOW => open,
    P(47 downto 0) => P_OUT(47 downto 0),
    PATTERNBDETECT => open,
    PATTERNDETECT => open,
    PCOUT(47 downto 0) => PCOUT_OUT(47 downto 0),
    UNDERFLOW => open
  );
end BEHAVIORAL;
```

Система условных обозначений интерфейсных портов элемента *add\_sub\_18\_48\_48* помимо идентификаторов, представленных в предыдущих двух разделах, включает еще четыре наименования входов:

- **ALUMODE\_IN [3:0]** — 4-разрядная входная шина управления выбором выполняемой операции;
- **CEALUMODE\_IN** — вход сигнала разрешения синхронизации для регистра управления, в который заносится код осуществляемой операции;

- RSTALUMODE\_IN — вход сигнала сброса для регистра управления выбором выполняемой функции;
- RSTCTRL\_IN — вход сигнала сброса буферных регистров, установленных в цепях управления загрузкой входных данных и входного переноса.

В случае применения элемента *add\_sub\_18\_48\_48* в качестве одного из компонентов в составе описания разрабатываемого устройства ЦОС нужно добавить в файл временных и топологических ограничений проекта САПР серии Xilinx ISE приведенную далее совокупность выражений:

```
# Generated by Xilinx Architecture Wizard
# --- UCF Template Only ---
# Cut and paste these attributes into the project's UCF file, if desired
INST DSP48E_INST ACASCREG = 1;
INST DSP48E_INST ALUMODEREG = 1;
INST DSP48E_INST AREG = 0;
INST DSP48E_INST AUTORESET_PATTERN_DETECT = FALSE;
INST DSP48E_INST AUTORESET_PATTERN_DETECT_OPTINV = MATCH;
INST DSP48E_INST A_INPUT = DIRECT;
INST DSP48E_INST BCASCREG = 1;
```

```
INST DSP48E_INST BREG = 0;
INST DSP48E_INST B_INPUT = CASCADE;
INST DSP48E_INST CARRYINREG = 1;
INST DSP48E_INST CARRYINSELREG = 0;
INST DSP48E_INST CREG = 0;
INST DSP48E_INST MASK = 3FFFFFFFFF;
INST DSP48E_INST MREG = 1;
INST DSP48E_INST MULTCARRYINREG = 1;
INST DSP48E_INST OPMODEREG = 0;
INST DSP48E_INST PATTERN = 000000000000;
INST DSP48E_INST PREG = 1;
INST DSP48E_INST SEL_MASK = MASK;
INST DSP48E_INST SEL_PATTERN = PATTERN;
INST DSP48E_INST SEL_ROUNDING_MASK = SEL_MASK;
INST DSP48E_INST USE_MULT = MULT_S;
INST DSP48E_INST USE_PATTERN_DETECT = NO_PATDET;
INST DSP48E_INST USE_SIMD = ONE48;
```

*Продолжение следует*

## Литература

1. Кузелин М. О., Кнышев Д. А., Зотов В. Ю. Современные семейства ПЛИС фирмы Xilinx. Справочное пособие. М.: Горячая линия – Телеком, 2004.
2. Зотов В. Инструментальный модуль компании Avnet для отладки проектов встраиваемых сис-

тем, разрабатываемых на базе нового семейства ПЛИС FPGA фирмы Xilinx Virtex-5 FXT // Компоненты и технологии. 2008. № 9.

3. Зотов В. Ю. Проектирование цифровых устройств на основе ПЛИС фирмы Xilinx в САПР WebPack ISE. М.: Горячая линия – Телеком, 2003.
4. Зотов В. Ю. Проектирование встраиваемых микропроцессорных систем на основе ПЛИС фирмы Xilinx. М.: Горячая линия – Телеком, 2006.
5. Virtex-5 FPGA XtremeDSP Design Considerations. User Guide.
6. Зотов В. Разработка компонентов устройств ЦОС, реализуемых на базе аппаратных модулей DSP48E в ПЛИС FPGA серии Virtex-5, с помощью «мастера» Architecture Wizard САПР серии Xilinx ISE // Компоненты и технологии. 2008. № 12.
7. Зотов В. Разработка компонентов устройств ЦОС, реализуемых на базе аппаратных модулей DSP48E в ПЛИС FPGA серии Virtex-5, с помощью «мастера» Architecture Wizard САПР серии Xilinx ISE // Компоненты и технологии. 2009. № 1.