

Хлуденев Александр Владимирович

PSPICE МОДЕЛИ БИС ДЛЯ ВИРТУАЛЬНОГО ПРОТОТИПИРОВАНИЯ ПЕЧАТНЫХ УЗЛОВ

В работе рассматривается задача формирования моделей цифровых БИС системного уровня, предназначенных для виртуального прототипирования схем печатных узлов. Используются инструменты симулятора схем PSpice A/D. Предлагаются решения по реализации PSpice SystemC моделей специализированных БИС, в которых реализован конвейер обработки данных. Рассмотрены способы обеспечения точности и учета временных ограничений на границе интерфейса БИС.

Печатный узел, виртуальное прототипирование, интегральная схема (ИС), большая интегральная схема (БИС), модель системного уровня, симулятор схем, задержка распространения, временное ограничение.

Khludenev Alexander Vladimirovich

PSPICE LSI MODELS FOR VIRTUAL PROTOTYPING OF PCB ASSEMBLIES

The paper examines the problem of forming system-level LSI models intended for virtual prototyping of printed circuit board (PCB) assemblies. PSpice A/D circuit simulator tools are used. Solutions are proposed for the implementation of PSpice Sys-

temC models of application-specific standard products, in which a data processing pipeline is implemented. The means of ensuring accuracy and allowing for time constraints at the LSI interface are considered.

PCB assembly, virtual prototyping, integrated circuit (IC), large scale integration (LSI), system-level model, circuit simulator, propagation delay, time constraint.

Введение

Основной элементной базой современной электроники являются БИС. В конечных изделиях все комплектующие, включая БИС, другие ИС и дискретные компоненты, устанавливаются на печатные платы, разработку которых выполняют, используя инструменты САПР печатных узлов [1]. Виртуальное прототипирование выполняют в процессе разработки новых изделий с целью проверки полученных проектных решений перед созданием физического образца, используя их компьютерные модели. Для верификации схем печатных узлов обычно используют SPICE симуляторы. Рекомендации по разработке и применению SPICE моделей устанавливает комплекс национальных стандартов [2].

В состав САПР печатных узлов OrCAD PCB Designer входит симулятор PSpice A/D, способный моделировать работу схем, построенных из аналоговых и цифровых ИС и дискретных элементов. PSpice макромодели цифровых ИС реализованы на уровне логических вентилях. Использовать аналогичные модели для БИС неприемлемо по затратам машинного времени. Начиная с версии 17.2, симулятор PSpice A/D способен работать с моделями системного уровня, реализованными на языках C/C++, SystemC и Verilog-A. В библиотеках OrCAD PCB Designer отсутствуют готовые модели БИС. Инструменты системного моделирования PSpice A/D ориентированы на выполнение ускоренного анализа схем печатных узлов, включающих до нескольких БИС. Проектно-ориентированные модели БИС должны быть экономичными и обеспечивать требуемую точность на границе их внешнего интерфейса. Разработку таких моделей для своих проектов должны выполнять пользователи САПР печатных узлов. Исследование путей реализации PSpice C/C++/SystemC моделей цифровых БИС является актуальной задачей. В работе исследуются пути построения PSpice SystemC моделей БИС с аппаратной реализацией алгоритма.

Основная часть

К моделям системного уровня относят модели, построенные на уровнях абстракции выше уровня регистровых передач (Register transfer level

(RTL)) [3]. Средства системного моделирования появились и развивались как инструмент верификации проектов сложных цифровых БИС [4]. SystemC модели БИС, в которых алгоритмы обработки данных реализованы аппаратно, могут быть выполнены на различных уровнях абстракции [5]:

- функциональных (поведенческих) моделей, не учитывающих временные параметры;
- функциональных моделей, учитывающих время протекающих процессов с точностью до такта синхронизации на границе внешних выводов;
- моделей на уровне регистровых передач.

Основными задачами виртуального прототипирования схем цифровых устройств в PSpice A/D являются обнаружение временного рассогласования сигналов, рисков сбоя, нарушений условий синхронизации. С этой целью в макромоделе ИС включены временные модели, в которых учитываются разброс задержек распространения сигналов и временные ограничения для поступающих на входы сигналов с точностью до такта моделирования.

Макромоделю цифровых ИС в PSpice A/D имеют вложенную организацию. Внешний контейнер выполнен как подсхема, элементами которой могут быть пятизначные логические примитивы (вентили, триггеры) или функциональный примитив LOGICEXP, а также примитивы матрицы задержек PINDLY и временных ограничений CONSTRAINT. Динамические параметры логических вентилях и функциональных примитивов могут быть заданы временной моделью примитива типа ugate, трехстабильных буферов - моделью типа utgate. Динамические параметры и временные ограничения для триггеров задаются временной моделью типа ueff. В модели типа ugate задаются одинаковые значения задержек для всех путей распространения сигналов. Если в примитиве LOGICEXP необходимо учитывать разные значения задержек для различных путей, то вместо модели ugate используют примитив матрицы задержек PINDLY. Временные ограничения для сигналов на входах синхронных автоматов в составе LOGICEXP можно задать, используя примитив CONSTRAINT [6].

Если к выводам цифровых ИС подключены дискретные элементы или аналоговые ИС, то PSpice A/D автоматически вставляет между ними подсхемы аналого-цифрового AtoD или цифро-аналогового DtoA интерфейса для преобразования между уровнями напряжения и логическими состоя-

ниями цифровых входов и выходов. Тип подсхем DtoA и AtoD задается моделью вход - выход.

Для PSpice макромоделей системного уровня сохранена стандартная организация, при этом функциональной моделью является примитив LOGICEXP с квалификатором C_MODEL и ссылкой на dll файл исполняемого модуля модели. Взаимодействие вычислительного ядра PSpice A/D с моделями устройств системного уровня обеспечивает Device Modeling Interface (DMI). Прототип PSpice C/C++/SystemC модели БИС можно сформировать, используя инструмент DMI Template code generator утилиты PSpice Model Editor [7]. Автоматически формируются:

- схемный символ для размещения на схеме устройства;
- макромодель на входном языке PSpice A/D;
- набор файлов проекта MS Visual Studio для формирования исполняемого модуля C/C++/SystemC модели.

DMI Template code generator формирует шаблон SystemC модели для конвейерного фильтра с конечной импульсной характеристикой (КИХ-фильтра) [7–9]. Для примитива LOGICEXP формируется временная модель типа ugate. Файл проекта PSpiceDigApiDefs.h содержит объявления класса PSpiceConstraint и методов для формирования ограничений времени пред-установки и удержания mSetupHold, ширины импульсов mWidth, тактовой частоты mFreq [10]. Таким образом, пользователю доступны два способа задания временных ограничений:

- на уровне PSpice макромодели, используя примитив CONSTRAINT;
- на уровне C/C++/SystemC модели, используя методы PSpiceConstraint. Проверка применения этих способов была выполнена на примере построения SystemC модели БИС "Процессор Пифагора" PDSP16330 [11]. БИС PDSP16330 реализована в виде конвейера данных и предназначена для преобразования данных из декартовой в полярную систему координат. Входные и выходные сигналы PDSP16330, их идентификаторы в PSpice макромодели и SystemC модели приведены в табл. 1. Все входы, кроме входов разрешения nOEM и nOEP для трехстабильных выходов **M** и **P**, синхронные. Данные входных каналов **X** и **Y** могут быть представлены в прямом (при FORM = 1) или дополнительном (при FORM = 0) коде. Комбинации значений S1S0 от 00 до 11 задают масштабный коэффициент от 1 до 8 для модуля **M**. Значения параметров и временные ограничения для PDSP16330A приведены в табл. 2.

В файле SysCPithPU.h объявлен класс SC_MODULE SysCPithPU с портами внешнего интерфейса, конструктором SC_CTOR(SysCPithPU) модуля, выполняющего синхронизируемый фронтом CLK поток SC_THREAD(entry, CLK.pos()). Метод entry SystemC модели сохранен в файле SysCPithPU.cpp (текст был отредактирован). При моделировании в каждом такте CLK выполняется чтение входных сигналов nCEX, nCEY, FORM, S и сохранение их значений в локальных переменных ENX, ENY, FORMFF, SFF. При ENX=0 и ENY=0 данные X и Y сохраняются во входных ячейках линий задержки tapsX и tapsY для учета задержки латентности. Все операции обработки выполняются в одном такте CLK для значений из последних ячеек линий задержки.

Таблица 1

Входные и выходные сигналы PDSP16330

Сигналы	Имя в макромодели	SystemC тип и идентификатор
Тактовые импульсы	CLK	sc_in<bool> CLK
Данные X	X_0 .. X_15	sc_in<sc_int<16>> X
Данные Y	Y_0 .. Y_15	sc_in<sc_int<16>> Y
Разрешение тактирования X	nCEX	sc_in<sc_int<1>> nCEX
Разрешение тактирования Y	nCEY	sc_in<sc_int<1>> nCEY
Формат данных X и Y	FORM	sc_in<sc_int<1>> FORM
Управление масштабированием M	S_0 .. S_1	sc_in<sc_int<2>> S
Разрешение выхода M	nOEM	sc_in<sc_int<1>> nOEM
Разрешение выхода P	nOEP	sc_in<sc_int<1>> nOEP
Данные модуля M	M_0 .. M_15	sc_out<sc_int<16>> M
Данные фазы P	P_0 .. P_11	sc_out<sc_int<12>> P
Флаг переполнения M	OVR	sc_out<sc_int<1>> OVR

Таблица 2

Значения параметров и временные ограничения

Параметр	Минимальное значение	Максимальное значение
Время предустановки X , Y , S , нс	12	
Время удержания X , Y , S , нс	2	
Время задержки от фронта CLK до уста-	5	25

новления данных, нс		
Период CLK, нс	50	
Длительность импульса CLK, нс	15	
Длительность паузы CLK, нс	15	
Задержка латентности, тактов	24	24

Шаблоны пользовательских функций приведены в файле `pspSysCPithPU_user.cpp`. Функция `pspSysCPithPU::initialize()` вызывается однократно при запуске Transient анализа и выполняет инициализацию портов и переменных модели. Функция `pspSysCPithPU::evaluate(double pTicks, PSpiceState* pVectorStates, int pSize)` вызывается PSpice A/D в каждом такте моделирования цифровой части схемы и выполняет операции:

- получает текущее время `pTicks`;
- получает PSpice пятизначные уровни входных сигналов из `pVectorStates`;
- детектирует фронт синхросигнала CLK;
- преобразует PSpice уровни в значения сигналов SystemC модели;
- вызывает SystemC поток;
- преобразует SystemC данные в PSpice уровни выходных сигналов;
- обновляет состояния выходов в `pVectorStates`.

Для формирования dll модуля модели использовались библиотека SystemC-2.3.3 и MS Visual Studio Community 2019. Попытки формировать временные ограничения в C++ коде модели были неудачными. В ходе Transient анализа возникали ошибки выполнения операций с плавающей точкой. При использовании примитива CONSTRAINT, таких проблем не возникало. Параметры задержки сигналов задавались в модели ugate PSpice макромодели. На рис. 1 приведены временные диаграммы входных сигналов в одном из модельных экспериментов. Выполнялась проверка адекватности функциональной модели, были определены задержка латентности и задержки распространения. На рис. 2 приведена реакция на выходах при типовых значениях задержки распространения и предельно допустимой тактовой частоте 20 МГц.

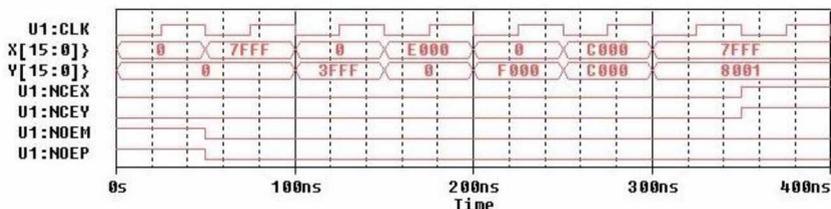


Рис. 1. Временные диаграммы входных сигналов

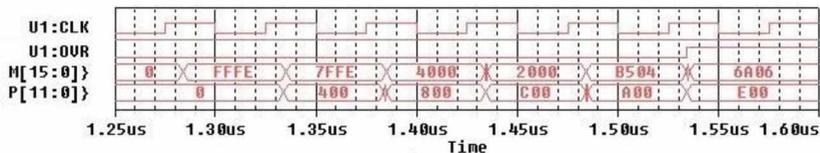


Рис.2. Временные диаграммы выходных сигналов

Выводы

В упрощенных PSpice SystemC моделях БИС конвейерного типа все операции обработки можно выполнять без распределения по ступеням конвейера. Задержку латентности с точностью до такта можно реализовать с помощью линий задержки. При равенстве задержек сигналов для всех путей распространения они могут быть учтены во временной модели примитива LOGICEXP типа ugate. Иначе следует использовать примитив матрицы задержек PINDLY. Учет разброса задержек распространения позволяет выполнять анализ рисков сбоя для наихудшего случая. Попытки использовать встроенные в SystemC модель средства формирования временных ограничений не дали удовлетворительного результата. При использовании примитива CONSTRAINT симулятор PSpice A/D формировал предупредительные сообщения при нарушениях заданных по таблице 2 условий предустановки, удержания, для ширины тактовых импульсов, а также при превышении максимальной тактовой частоты.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. ГОСТ Р 70607–2022. Системы автоматизированного проектирования электроники. Состав и структура системы автоматизированного проектирования печатных узлов. – Введ. 2023–02–01. – М.: Рос. ин-т стандартизации, 2023. – 8 с.
2. ГОСТ Р 70884–2023. Системы автоматизированного проектирования электроники. Информационное обеспечение. Модели SPICE. Общие положения. – Введ. 2022–10–01. – М.: Рос. ин-т стандартизации, 2022. – 11 с.

3. Panda P. R. SystemC – a modeling platform supporting multiple design abstractions // *International Symposium on System Synthesis (IEEE Cat. No.01EX526)*. – IEEE, 2001. – P. 75–80. – doi: 10.1145/500001.500018.

4. Алексин В.А. Проектирование электронных систем с использованием SystemC и SystemC–AMS // *Российский технологический журнал*, 2020. – Т. 8, № 4. – С. 79–95.

5. Алексин В.А. SystemC. Моделирование электронных систем: учебное пособие для вузов. – М. : Горячая линия – Телеком, 2018. – 320 с.

6. PSpice User Guide. Product Version 17.2–2016. – Cadence, April 2016. – 900 p.

7. PSpice Device and System Modeling with C/C++ and SystemC. Product Version 17.2–2016. – Cadence, April 2016. – 34 p.

8. Virtual Prototyping in PSpice: Application Note, V1. – FlowCAD. – 72 p. – URL: https://www.flowcad.de/AN/FlowCAD-AN_Device-Modeling-Interface.pdf (дата обращения 20.05.2024).

9. Хлуднев А.В. Моделирование цифровых устройств на системном уровне // *Актуальные проблемы и перспективы в сфере инженерной подготовки*. – Оренбург: Изд-во ОФ ПГУТИ, 2022 – С. 127–132.

10. PSpice Device Modeling Interface API Reference. Product Version 17.2–2016. – Cadence, April 2016. – 93 p.

11. PDSP16330/A/B Pythagoras Processor: datasheet DS3884. – Issue 2.1. – Zarlink Semiconductor, November 1998. – 6 p. – URL: https://datasheetpdf.com/download_new.php?id=576650 (дата обращения 20.05.2024).

Хлуднев Александр Владимирович, кандидат технических наук, доцент кафедры промышленной электроники и информационно-измерительной техники Оренбургского государственного университета, Россия, город Оренбург, проспект Победы 13, 460018, телефон: +7 (3532) 37-28-74, email: avhludenev@yandex.ru.

Khudnev Alexander Vladimirovich, Candidate of Technical Sciences (Ph.D.), associate professor, Industrial Electronics and Information Measuring Engineering Department, Orenburg State University, 460018, Russia, Orenburg, 13 Victory Avenue, phone: +7 (3532) 37-28-74, email: avhludenev@yandex.ru.