УДК 004.312

Болдырев А.В.,
кандидат технических наук,
доцент кафедры «Автоматизация производственных процессов»
Донской Государственный Технический Университет
Россия, г. Ростов-на-Дону

ВСТРАИВАЕМАЯ СИСТЕМА НА ПЛИС ДЛЯ УЧЕБНОГО СТЕНДА

Аннотация: *В данной статье рассматриваются особенности создания прототипа встраиваемой системы на учебном стенде, предназначенном для обучения программированию ПЛИС. Система реализуется средствами пакета программ Quartus II и строится на основе модулей, образованных с помощью языка описания аппаратуры, и процессора с программируемым ядром Nios II, оптимизированного для работы на ПЛИС стенда. Полученные результаты подтверждают корректную работу встраиваемой системы.*

Ключевые слова: *встраиваемая система; ПЛИС; учебный стенд; Quartus II; Nios II; Cyclone; Intel/Altera.*

EMBEDDED SYSTEM BASED ON PLD FOR A TRAINING STAND

Abstract: *This article examines the specifics of creating a prototype of an embedded system on a training stand designed for PLD programming education. The system is implemented using the Quartus II software suite and is built on the basis of modules created with a hardware description language, as well as a processor with a programmable Nios II core, optimized for operation on the stand's PLD. The obtained results confirm the correct functioning of the embedded system.*

Keywords: *embedded system; PLD; training stand; Quartus II; Nios II; Cyclone; Intel/Altera.*

Типичная встраиваемая система на ПЛИС состоит из одного или нескольких процессоров, соединенных с памятью и другими устройствами с помощью шинной структуры [1]. Главной особенностью создания учебных встраиваемых систем является сочетание модулей, образованных с помощью языка описания аппаратуры, например Verilog, и процессора с программируемым ядром, оптимизированного для работы на ПЛИС учебного стенда.

Для создания прототипа встраиваемой системы удобно использовать учебный стенд SDK-6.1/E, хорошо известный в российских вузах [2]. Стенд оснащен основной и вспомогательной ПЛИС фирмы Intel/Altera, а также подсистемой памяти, необходимой для построения полнофункционального вычислительного ядра на основе процессора Nios II. Применительно к возможностям основной ПЛИС стенда (Altera Cyclone EP1C6) целесообразно использовать «экономичный» вариант ядра Nios II/e, который является основой 32-битного RISC-процессора и использует не более 700 логических элементов ПЛИС, а также всего два блока памяти по 4К.

На рисунке 1 приведен пример учебной встраиваемой системы, состоящей из клавиатуры, жидкокристаллического дисплея (ЖКД), их драйверов и вычислительного устройства с программным ядром Nios II/e.

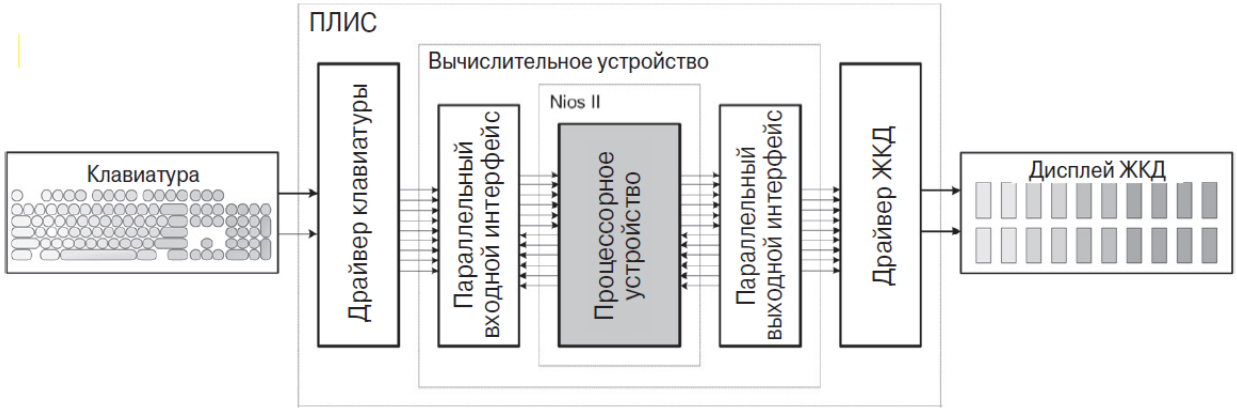

Рисунок 1. Пример встраиваемой системы

Данные и выполняемые операции вводятся с клавиатуры. Драйвер клавиатуры подготавливает параллельные данные для чтения процессором. Процессор читает эту информацию через свой интерфейс, выполняет определенные операции и передает результат на интерфейс ЖКД для отображения. В состоянии готовности драйвер ЖКД принимает данные из своего параллельного порта и отображает их на дисплее ЖКД.

Клавиатура и дисплей на рисунке 1 являются физическими устройствами, которые присоединяются к выводам ПЛИС.

Полный проект, включающий драйверы клавиатуры [3] и ЖКД [4], определяется как проект пакета Quartus II компании Intel/Altera [5], а вычислительное устройство становится файлом проекта в программе SOPC Builder, являющейся составной частью пакета.

Создание и конфигурирование проекта встраиваемой системы выполняется через меню New в ряде страниц Quartus II, в которых задаются название и место расположения проекта, выбирается семейство ПЛИС (Cyclone) и конкретная микросхема (EP1C6T144C8), используемая в стенде SDK-6.1/E. Так как для стенда требуется конфигурационный файл с расширением .rbf [6], необходимо отметить опцию включения его создания при компиляции проекта. Для этого в меню Quartus II следует открыть командой Assignments/Device окно настроек, в котором нажать кнопку Device and Pin Options, а затем выбрать в открывшемся окне вкладку Programming Files и проверить, отмечен ли галочкой пункт Raw Binary File (.rbf). Во избежание выхода из строя внешних устройств или БИС на плате стенда в связи с тем, что по умолчанию неиспользованные выводы ПЛИС соединяются с общей шиной, в этом же окне необходимо выбрать вкладку Unused Pins и установить As input tri-stated. Для создания файла, содержащего описание структурной схемы выбирается File/New/Design Files, а в таблице – Block Diagram/Schematic File. В результате открывается окно графического

редактора с загруженным в него bdf-файлом, который необходимо сохранить, задав имя и установив флажок Add file to current project.

Конфигурирование вычислительного устройства осуществляется в программе SOPC Builder, в которой устройство определяется как файл системы на программируемом кристалле (СнПК). Программа, вызываемая из меню Tools, автоматизирует задачу объединения элементов аппаратуры в большую систему, в том числе за счет создания переключательной структуры Avalon.

При создании файла СнПК задается имя системы, выбирается язык HDL (Verilog), определяются семейство ПЛИС (Cyclone) и частота тактового генератора (40 МГц для стенда SDK-6.1/E), после чего выбираются и добавляются в систему необходимые модули из библиотечного списка. Процедура проводится в окне бланка программы SOPC Builder и сопровождается серией окон, позволяющих выбрать желаемые опции.

Первым в окно компонентов бланка устанавливается процессор Nios II/e с отказом от опции отладчика No Debugger для упрощения проекта. Размещение процессора автоматически вызывает размещение главной шины структуры Avalon для памяти команд ROM и главной шины для памяти данных RAM.

Добавление на следующих двух шагах обоих типов памяти On-Chip Memory, размещаемой внутри ПЛИС, сопровождается окном спецификации, в котором выбираются тип памяти (ROM и RAM), ее размер (32 bits) и объем (4К). После появления новых модулей в бланке СнПК необходимо дифференцировать имена типов памяти, подсоединение шин структуры Avalon и установить адреса компонентов так как это показано на рисунке 2.

При добавлении параллельных интерфейсов ввода-вывода PIO в окне спецификации на каждый из них в зависимости от назначения указывается тип (входной или выходной) и разрядность (1-битный или 8-битный). Интерфейс PIO, используемый для входа KeyReleased от клавиатуры, является 1-битным

входным портом, опрашиваемым процессором. Поэтому в одном из конфигурационных окон он определяется как входной порт прерывания IRQ. Аппаратура всех интерфейсов уже является частью переключательной структуры Avalon.

Дополнительные настройки процессора на последнем шаге конфигурирования СнПК сохраняются без изменений. Таким образом, в систему на программируемом кристалле добавляются процессорное ядро Nios II/e, два модуля памяти On-Chip Memory, два модуля PIO для клавиатуры и три модуля PIO для дисплея. В результате полное содержимое вычислительного устройства приобретает вид, показанный на рисунке 2.

По окончании конфигурирования для создания кода описания аппаратуры осуществляется генерирование системы (кнопка Generate на рисунке 2). При отсутствии ошибок работа с программой SOPC Builder завершается, в противном случае вносятся исправления и генерация системы повторяется. С успешным завершением создания вычислительного устройства формируется символ для данной СнПК – NIOS (рисунок 3).

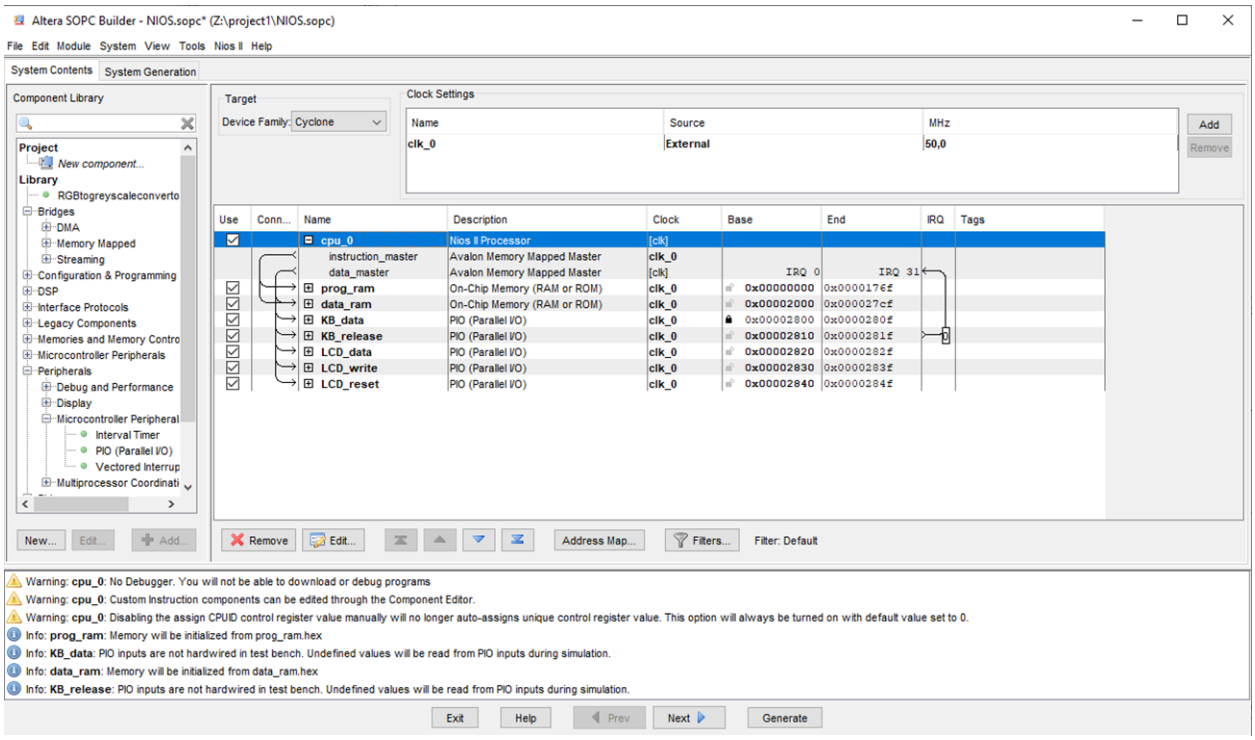

Рисунок 2. Конфигурация вычислительного устройства

Полученные файлы вручную добавляются в проект встраиваемой системы как и другие файлы, описывающие логику работы системы. Размещение их в окне графического редактора наряду с драйверами клавиатуры и ЖКД, другими компонентами, а также выполнение соединений ничем не отличается от известных методик [6]. Для выполнения назначений входных и выходных сигналов проекта выводам корпуса реальной ПЛИС Cyclone используется карта ее выводов, задействованных в работе стенда SDK-6.1/E [6], а сама процедура назначений осуществляется с помощью графического редактора Pin Planner.

Полная структурная схема встраиваемой системы, соответствующая примеру на рисунке 1, приведена на рисунке 3.

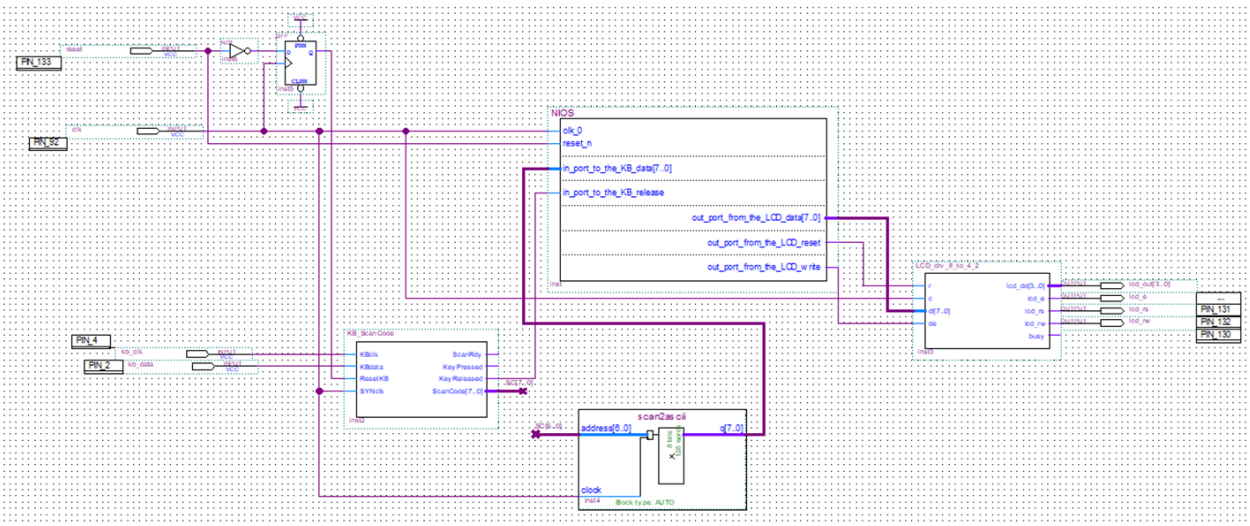

Рисунок 3. Схема встраиваемой системы

Построение программной части является последним этапом проектирования встраиваемой системы. Эта фаза компилирует программу на языке C/C++, помещаемую в память ROM проекта СнПК. Для создания программы используются инструменты Nios II Software Build Tools for Eclipse из меню Tools. В мастере создания приложения для конфигурирования пакета поддержки платы указывается файл описания СнПК, выбирается процессор для которого создаётся приложение, а также название проекта. В результате

формируется исполнимый elf-файл, из которого перед загрузкой в ПЛИС выделяется машинный код.

Для верификации проекта, после завершения его создания и компиляции выполняется синтез rbf-файла, содержащий в том числе машинный код для процессора Nios II. С помощью утилиты sdk61fw.exe [2] файл проекта загружается в любую страницу Flash-памяти стенда, не используемую для хранения конфигурационных файлов, данных или программ (согласно инструкции стенда SDK-6.1/E). По окончании загрузки к стенду подключается клавиатура по модифицированному интерфейсу PS/2.

На рисунке 4 в качестве примера функционирования учебной встраиваемой системы на стенде SDK-6.1/E показана работа приложения «Калькулятор».

Рисунок 4. Демонстрация работы встраиваемой системы

Используемые источники:

1. Наваби З. Проектирование встраиваемых систем на ПЛИС / пер. с англ. Соловьева В.В. – М.: ДМК Пресс, 2016.

2. Комплекс учебный лабораторный SDK-6.1. Руководство пользователя. [Электронный ресурс]: URL: http://embedded.ifmo.ru/sdk/sdk61/doc/sdk61_usermanual.pdf (дата обращения 12.11.2025).

3. Болдырев А.В. Логический интерфейс клавиатуры для стенда на основе ПЛИС [Электронный ресурс]: URL: https://alley-science.ru/domains_data/files/Dec25/LOGICHESKII-INTERFEIS-KLaVIaTURI-DLYa-STENDa-Na--OSNOVE-PLIS.pdf?x67998 (дата обращения 5.01.2026).

4. Болдырев А.В. Интерфейс дисплея для учебного стенда на основе программируемой логической интегральной схемы // Научное обозрение. Технические науки. 2025. № 5. С. 12-18. URL: https://science-engineering.ru/ru/article/view?id=1521 (дата обращения: 09.02.2026). DOI: https://doi.org/10.17513/srts.1521

5. Программа Quartus II Web Edition (Win) [Электронный ресурс]: URL: http://www.altera.com.

6. Болдырев А.В. Основы проектирования ПЛИС и БМК: практикум. – Ростов-на-Дону: Донской гос. техн. ун-т, 2019. – 50 с.