

*Болдырев А.В., кандидат технических наук
Доцент кафедры «Автоматизация производственных процессов»
Донской Государственный Технический Университет
Россия, г. Ростов-на-Дону
Степаненко Д.Р.,
Студент магистратуры
2 курс, факультет «Автоматизация, мехатроника и управление»
Донской Государственный Технический Университет
Россия, г. Ростов-на-Дону*

ОСОБЕННОСТИ ПРОЕКТИРОВАНИЯ ВСТРАИВАЕМЫХ СИСТЕМ НА ПЛИС

***Аннотация:** В статье рассмотрены особенности проектирования встраиваемых систем на ПЛИС применительно к стенду SDK-6.1/E. Кратко представлены специальные средства и среды проектирования. Основное внимание уделяется проектированию аппаратной части системы на языке Verilog и ее реализации на учебном стенде SDK-6.1/E отечественного производства.*

***Ключевые слова:** Программируемая логическая интегральная схема, проектирование, Altera, SDK-6.1/E, Nios II, Quartus II, System on a Programmable Chip Builder.*

FEATURES OF THE DESIGN OF EMBEDDED SYSTEMS ON THE PLD

***Annotation:** The article describes the features of the design of embedded systems on the PLD as applied to the SDK-6.1 / E stand. Special tools and design environments are briefly presented. The focus is on designing the hardware of the system in the*

Verilog language and its implementation on the SDK-6.1 / E training stand of domestic production.

Keywords: *Programmable logic device, design, Altera, SDK-6.1/E, Nios II, Quartus II, System on a Programmable Chip Builder.*

Типичная встраиваемая система на ПЛИС состоит из одного или нескольких процессоров, соединенных с памятью и другими устройствами с помощью шинной структуры [1]. Функции системы, которые должны быть реализованы аппаратно, проектируются методами аппаратного проектирования и средствами синтеза, а функции, реализуемые с помощью выполнения программного обеспечения процессора, проектируются с использованием языка С и компиляторов, а также других программных средств.

Для изучения основ проектирования электронных модулей на базе ПЛИС в российских вузах успешно применяется стенд SDK-6.1/E. Стенд также может использоваться для создания прототипов встраиваемых систем автоматического управления и сигнальной обработки, в т.ч. измерительных приборов и контроллеров [3].

Стенд оснащен основной и вспомогательной ПЛИС фирмы Altera, а также подсистемой памяти (FLASH, SRAM, EEPROM), необходимой для построения полнофункционального вычислительного ядра.

Разработка проектов и генерация конфигурационных файлов для стенда SDK-6.1/E производится в инструментальной среде Quartus II фирмы Altera. Процессором для встраиваемых проектов является процессор Nios II, представляющий собой общецелевое процессорное ядро с архитектурой RISC.

Процессорная система Nios II по сути эквивалентна микроконтроллеру или «компьютеру на кристалле». Подобно семейству микроконтроллеров, все системы процессоров Nios II используют совместимое множество команд и модель программирования. На рисунке 1 показан пример системы, включающей процессор, устройства памяти, устройства ввода-вывода, структуру шины межсоединений.

Процессор Nios II поставляется в трех версиях. Применительно к возможностям ПЛИС станда SDK-6.1/Е целесообразно использовать ядро Nios II/e – «экономичное». Это ядро является основой 32-битного RISC-компьютера и использует не более 700 логических элементов ПЛИС, а также всего два блока памяти по 4К.

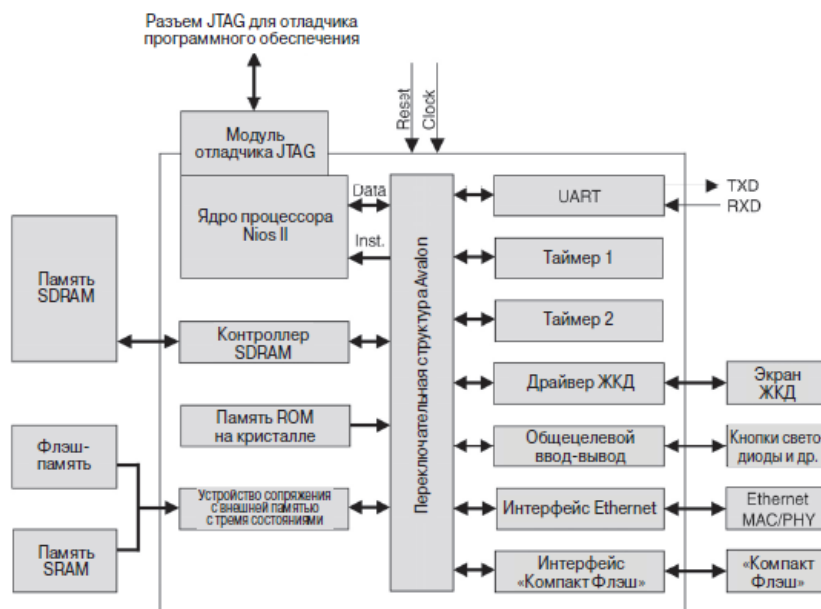


Рисунок 1 – Пример системы Nios II

Средством автоматизации в пакете Quartus II задачи объединения элементов аппаратуры в большую систему является программа SOPC Builder.

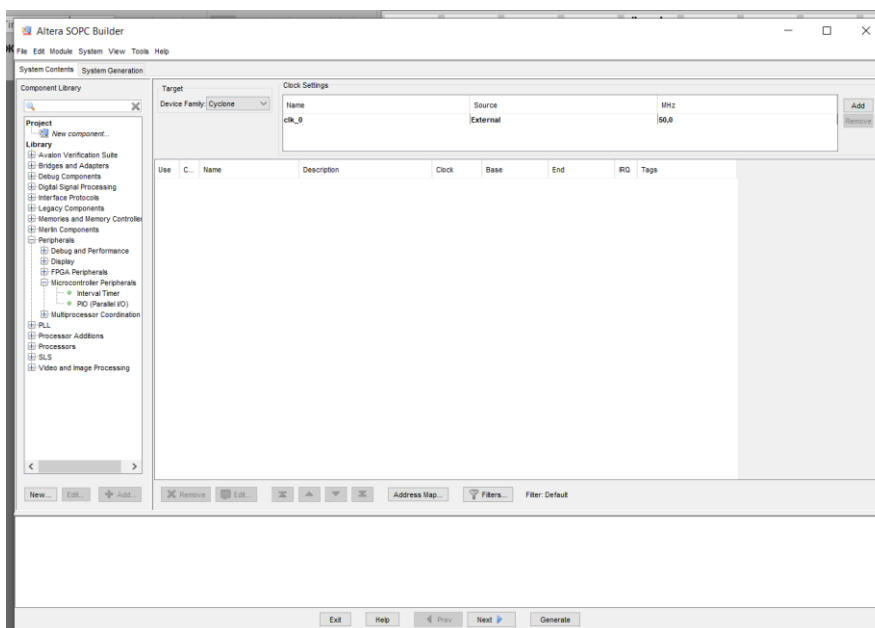


Рисунок 2 – Интерфейс SOPC Builder

Программа SOPC Builder является общецелевым средством для создания произвольных проектов «систем на кристалле», которые могут содержать (а могут и не содержать) процессор.

Еще один программный продукт от Altera – интегрированная среда проектирования IDE предназначен для разработки на языке C программы, которая выполняется на процессоре Nios II. IDE включает компилятор языка C, а также необходимые утилиты для ввода и тестирования программы.

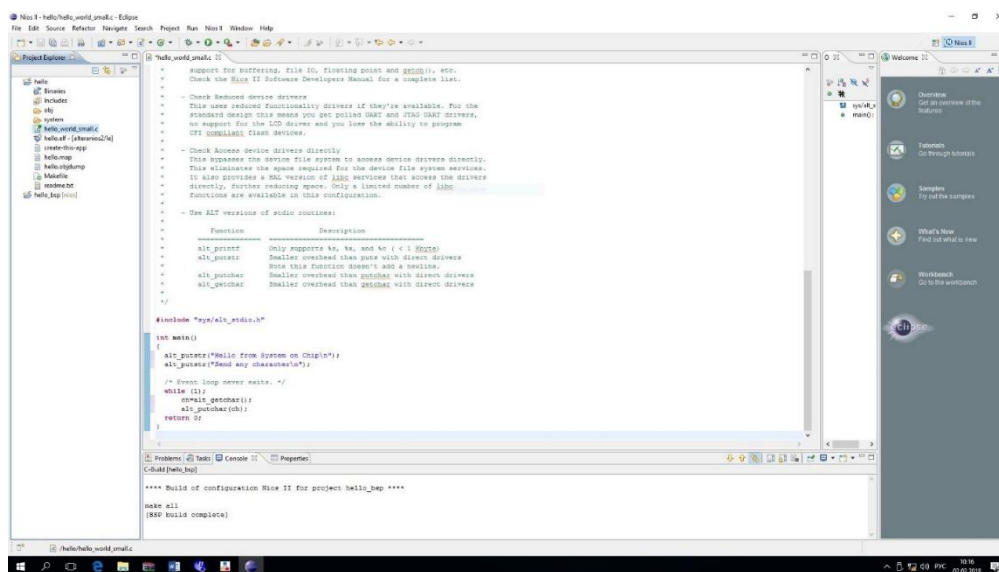


Рисунок 3 – Интерфейс Nios II IDE

Неотъемлемой частью встраиваемой системы являются устройства ввода-вывода. Как правило, данные и выполняемые операции в любом проекте вводятся с помощью клавиатуры. Драйвер клавиатуры подготавливает параллельные данные для чтения процессором. Процессор читает эту информацию через свой интерфейс, выполняет определенные операции и передает результат на интерфейс жидкокристаллического дисплея (ЖКД). В состоянии готовности драйвер ЖКД принимает данные из своего параллельного порта и отображает их на ЖКД.

Клавиатура и ЖКД являются физическими устройствами, которые подсоединяются к выводам ПЛИС. Применительно к учебному стенду SDK-6.1/Е для подключения полноразмерной клавиатуры (рисунок 4) с интерфейсом

PS/2 могут быть использованы внешние разъемы стенда «EXTERNAL PIO» и «JTAG» [3]. Двухстрочный 16-символьный ЖКД, отображающий 8-битный код ASC II символов, располагается на самом стенде, однако подключается к ПЛИС по 4-битной шине.

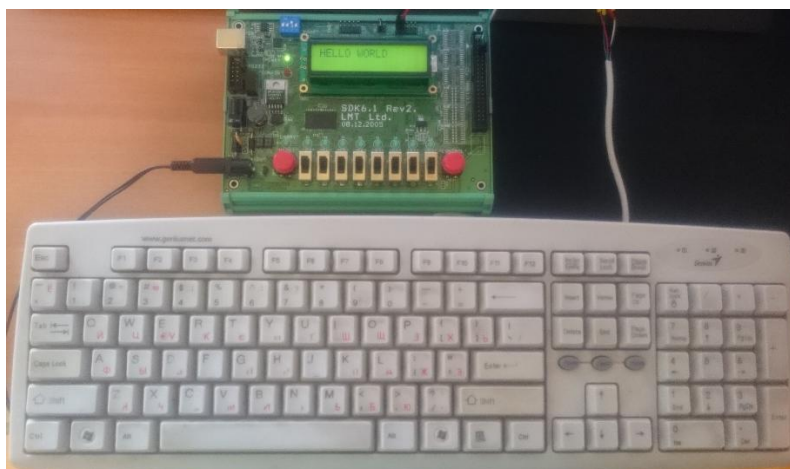


Рисунок 4 – Стенд SDK-6.1/E с подключенной клавиатурой

Драйверы клавиатуры и ЖКД реализуются на ПЛИС стенда в среде Quartus II с использованием языка описания аппаратуры Verilog.

Для тестирования драйверов клавиатуры и ЖКД была спроектирована структурная схема тестера, приведенная на рисунке 5. Схема позволяет не только подтвердить работоспособность устройств, но и демонстрирует возможность использования разработанных драйверов в других проектах встраиваемых систем на основе ПЛИС стенда SDK-6.1/E.

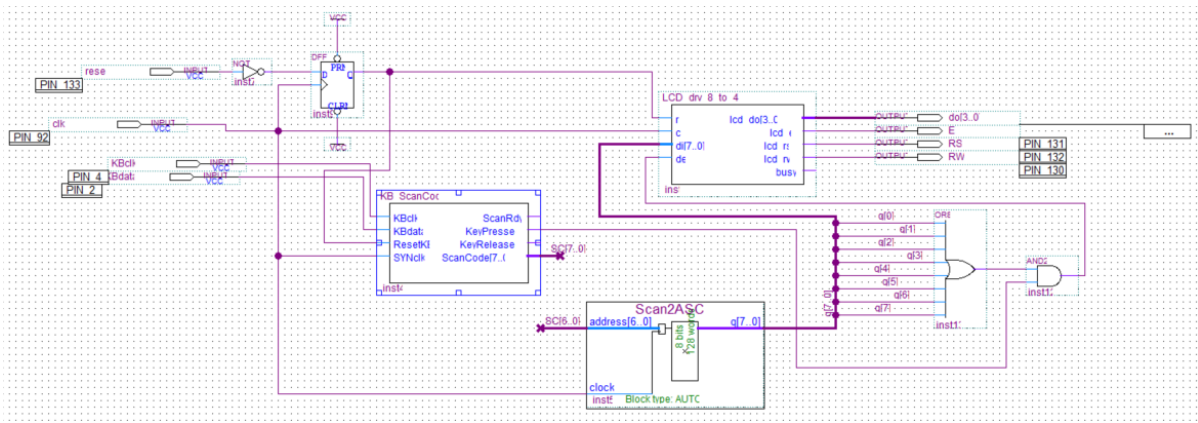


Рисунок 5 – Структурная схема тестера

Драйвер клавиатуры, принимающий сигналы от нажатой клавиши и формирующий соответствующий код ASCII, состоит из модуля KB_ScanCode и модуля памяти типа ПЗУ Scan2ASCII для поиска кода ASCII выбранного символа.

Драйвер ЖКД реализуется на модуле LCD_drv_8_to_4, в котором осуществляется инициализация экрана, а также преобразование 8-битных данных в последовательность, позволяющую подключиться к ЖКД по 4-битной шине.

Логическая схема на вентилях OR8 и AND2 блокирует нулевые коды на выходе модуля Scan2ASCII. Элементы DFF и NOT исключают помехи в цепи сброса клавиатуры при нажатии кнопки Reset. Вход KBdata является входом данных от клавиатуры, а на вход KBclk подаются сигналы для ее синхронизации от тактового генератора стенда SDK-6.1/E.

Драйвер клавиатуры (рисунок 6) читает из клавиатуры коды клавиш и формирует коды ASCII. Когда клавиша отпускается, драйвер помещает код ASCII символа, который соответствует верхнему регистру символа нажатой клавиши, на свой выход и формирует сигнал *KeyPress* на длительность одного такта синхросигнала.

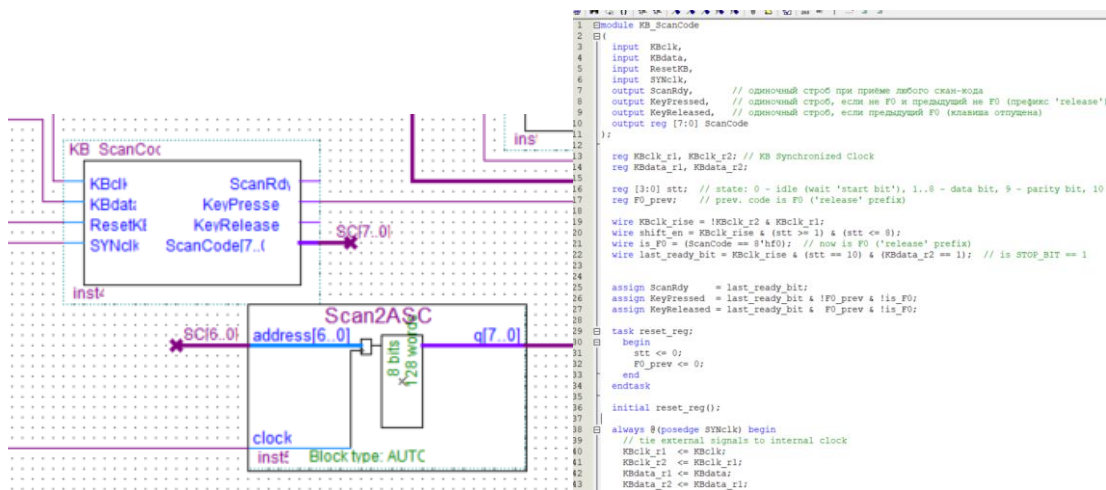


Рисунок 6 – Реализация драйвера клавиатуры

Дисплей имеет внутреннюю память для определения матрицы символов, а также стандартного множества символов. Для того чтобы использовать этот дисплей, он должен быть проинициализирован, а затем выполняется запись в его порты, которые подсоединяются к выводам ПЛИС станда SDK-6.1/Е. Пример реализации драйвера ЖКД приведен на рисунке 7.

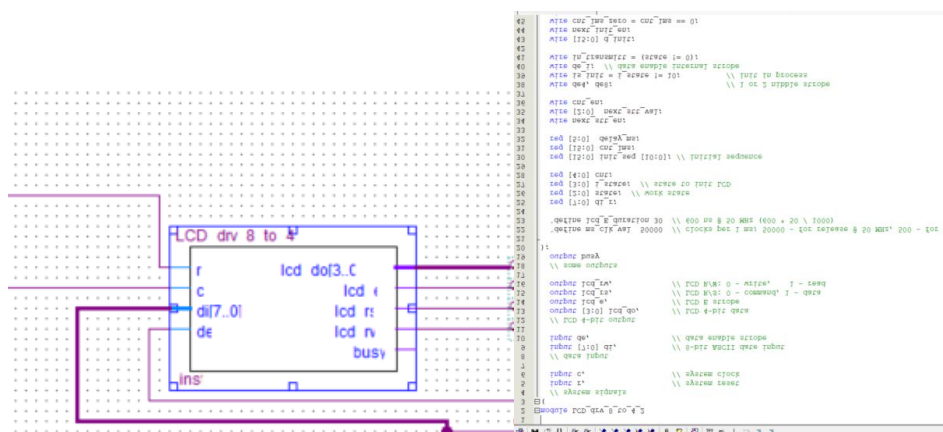


Рисунок 7 – Реализация драйвера ЖКД

Разработка интерфейса ввода-вывода встраиваемой системы на основе ПЛИС станда SDK-6.1/Е предваряет полный проект, в котором создается собственно вычислительное устройство системы на процессоре Nios II с помощью программы SOPC Builder. Рассмотрение среды проектирования фирмы Altera продиктовано ограничениями конструкции станда SDK-6.1/Е. Однако

полученные результаты могут быть использованы и в других пакетах проектирования встраиваемых систем на основе ПЛИС.

ИСПОЛЬЗУЕМЫЕ ИСТОЧНИКИ

1. Наваби З. Проектирование встраиваемых систем на ПЛИС / пер. с англ. Соловьева В.В. – М.: ДМК Пресс, 2016.
2. Болдырев А.В., Степаненко Д.Р. Обзор учебных стендов для обучения программированию ПЛИС // Инновации и инжиниринг в формировании инвестиционной привлекательности региона: Сб. научн. трудов II открытого международного научно-практического форума. – Ростов н/Д: Изд-во «ДГТУ-Принт», 2017.
3. SDK-6.1/E [Электронный ресурс]
<http://lmt.ifmo.ru/index.php/production/productboards/productsdk61>