



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.04 «Программная инженерия»

О Т Ч Е Т

по лабораторной работе № 1

Название: Разработка СНК на ПЛИС Altera

Дисциплина: Архитектура ЭВМ

Студент

ИУ7-53Б

(Группа)

(Подпись, дата)

М. Д. Маслова

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

Е. Н. Дубровин

(И.О. Фамилия)

Москва, 2021

Цель работы

Изучение основ построения микропроцессорных систем на ПЛИС. В ходе работы студенты ознакомятся с принципами построения систем на кристалле (СНК) на основе ПЛИС, получат навыки проектирования СНК в САПР Altera Quartus II, выполнят проектирование и верификацию системы с использованием отладочного комплекта Altera DE1Board.

Схема разрабатываемой СНК

В данной лабораторной работе реализуется система на кристалле состоящая из:

- микропроцессорного ядра Nios II/e;
- внутренней оперативной памяти СНК;
- системной шины Avalon;
- блока синхронизации и сброса;
- блока идентификации версии проекта;
- контроллера UART (интерфейс RS232).

Функциональная схема разрабатываемой системы на кристалле представлена на рисунке 1.

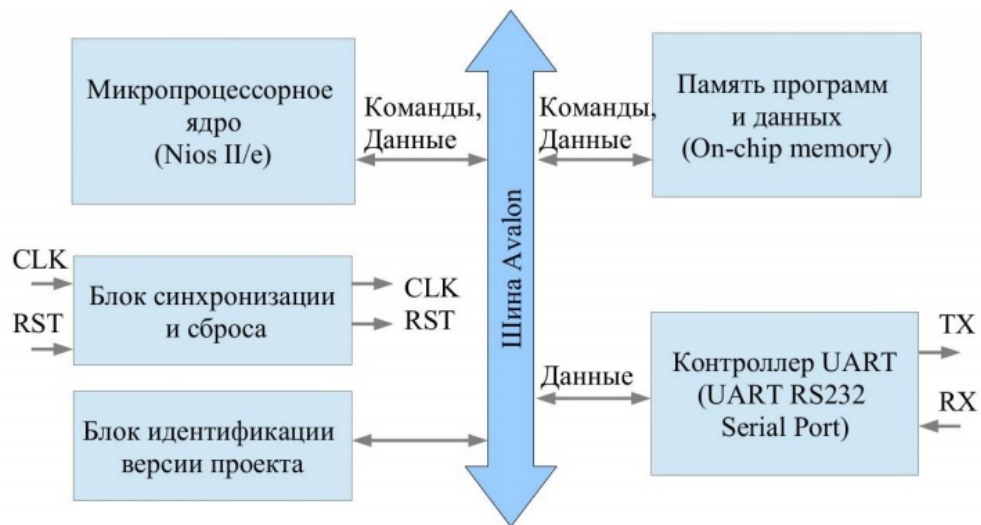


Рисунок 1 – Функциональная схема разрабатываемой системы на кристалле

Проектирование СНК в Quartus II

Модуль системы на кристалле в Quartus II создается в полном соответствии с выше приведенной схемой: добавляются необходимые компоненты системы и строятся связи между ними через шину Avalon, — в чем можно убедиться, внимательно сравнив схему, представленную на рисунке 1, и готовый модуль в системе проектирования (рисунок 2).

Use	Connections	Name	Description	Export	Clock	Base	End	IRQ	Opcode Name
<input checked="" type="checkbox"/>		clk_0	Clock Source						
		clk_in	Clock Input	clk					
		clk_in_reset	Reset Input	reset					
		clk	Clock Output	Double-click to export	clk_0				
		clk_reset	Reset Output	Double-click to export					
<input checked="" type="checkbox"/>		nios2_qsys_0	Nios II Processor						
		clk	Clock Input	Double-click to export	clk_0				
		reset_n	Reset Input	Double-click to export	[clk]				
		data_master	Avalon Memory Mapped Master	Double-click to export	[clk]		IRQ 0	IRQ 31	
		instruction_master	Avalon Memory Mapped Master	Double-click to export	[clk]				
		jtag_debug_module_re...	Reset Output	Double-click to export	[clk]	0x8800	0x8fff		
		jtag_debug_module	Avalon Memory Mapped Slave	Double-click to export	[clk]				
		custom_instruction_m...	Custom Instruction Master	Double-click to export					
<input checked="" type="checkbox"/>		onchip_memory2_0	On-Chip Memory (RAM or ROM)						
		clk1	Clock Input	Double-click to export	clk_0				
		s1	Avalon Memory Mapped Slave	Double-click to export	[clk1]	0x4000	0x6003		
		reset1	Reset Input	Double-click to export	[clk1]				
<input checked="" type="checkbox"/>		sysid_qsys_0	System ID Peripheral						
		clk	Clock Input	Double-click to export	clk_0				
		reset	Reset Input	Double-click to export	[clk]				
		control_slave	Avalon Memory Mapped Slave	Double-click to export	[clk]	0x9048	0x904f		
<input checked="" type="checkbox"/>		uart_0	UART (RS-232 Serial Port)						
		clk	Clock Input	Double-click to export	clk_0				
		reset	Reset Input	Double-click to export	[clk]				
		s1	Avalon Memory Mapped Slave	Double-click to export	[clk]	0x9020	0x903f		
		external_connection	Conduit	uart0					

Рисунок 2 – Готовый модуль в системе проектирования систем на кристалле

Для правильно работы системы необходимо корректное распределение адресного пространства между устройствами, так как неверное обращение к памяти может привести к непредсказуемому поведению системы или даже полной неработоспособности ПЛИС. В Quartus II данное распределение происходит автоматически, и, как видно из рисунка 3, каждому подключенному компоненту выделяется свое адресное пространство, при этом данные и код каждого устройства имеют одинаковое адресное пространство, что соответствует одному из принципов архитектуры фон Неймана.

	nios2_qsys_0.data_master	nios2_qsys_0.instruction_master
nios2_qsys_0.jtag_debug_module	0x8800 - 0x8fff	0x8800 - 0x8fff
onchip_memory2_0.s1	0x4000 - 0x6003	0x4000 - 0x6003
sysid_qsys_0.control_slave	0x9048 - 0x904f	0x9048 - 0x904f
uart_0.s1	0x9020 - 0x903f	0x9020 - 0x903f

Рисунок 3 – Таблица распределения адресов модулей в системе на кристалле

Программное обеспечение

После проектирования и сборки аппаратного обеспечения можно переходить к написанию программного кода. В данной лабораторной работе реализуются простейшие функциональности. Одной из них является отклик СНК на нажатие клавиши на клавиатуре. Код такой эхо-программы представлен на листинге 1. Результаты тестирования приведены на рисунке 4.

Листинг 1 – Код эхо-программы приема-передачи по интерфейсу RS232

```
1 #include "sys/alt_stdio.h"
2
3 int main()
4 {
5     char ch;
6     alt_putstr("Hello from System on Chip\n");
7     alt_putstr("Send any character\n");
8
9     while(1)
10    {
11        ch = alt_getchar();
12        alt_putchar(ch);
13    }
14
15    return 0;
16 }
```

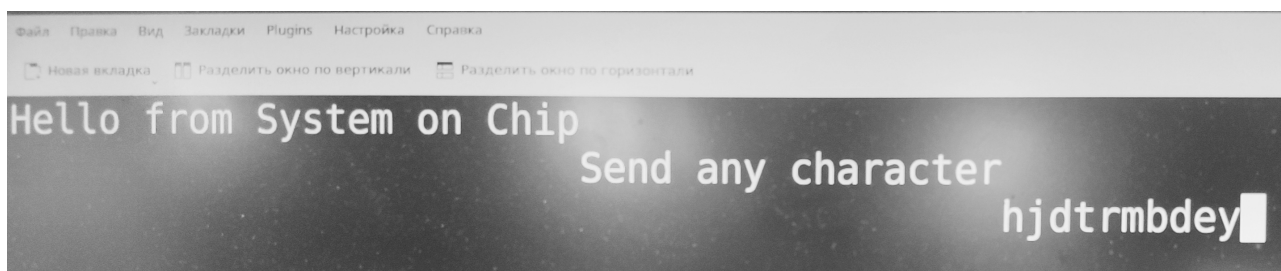


Рисунок 4 – Результаты тестирования эхо-программы

В виде более сложность функциональности нам предлагалось реализовать передачу по UART значения SystemID в виде четырех байт символов в формате ASCII.

Для проверки работоспособности программы в поле SystemID (рисунок 5) было записано значение в шестнадцатеричной системе счисления, соответствующее десятичному значению, полученному конкатенацией номера группы и варианта ($5302_{10} = 14B6_{16}$).

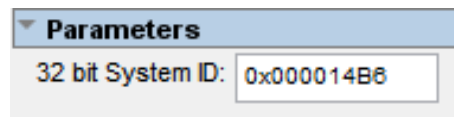


Рисунок 5 – Запись необходимого значения в SystemID

Реализованная нами программа представлена на листинге 2.

Листинг 2 – Код программы, передающей по UART значение SystemID

```
1 #include "system.h"
2 #include "altera_avalon_sysid_qsys.h"
3 #include "altera_avalon_sysid_qsys_regs.h"
4
5 int main()
6 {
7     alt_putstr("Hello from System on Chip\n");
8
9     int num = IORD_ALTERA_AVALON_SYSID_QSYS_ID(SYSID_QSYS_0_BASE);
10    char a[5] = "0000\0";
11
12    int i = 1;
13    while(num > 0)
14    {
15        a[4 - i] = num % 10 + '0';
16        i++;
17        num /= 10;
18    }
19
20    alt_putstr(a);
21
22    return 0;
23 }
```

Вывод

В данной лабораторной работе нами была реализована СНК на ПЛИС в системе проектирования Quartus II. Работоспособность системы на кристалле была проверена загрузкой в нее и выполнением простейших программ.