



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 «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ»(ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.04 «Программная инженерия»

О Т Ч Е Т

по лабораторной работе № 1

Название: Проектирование систем на кристалле на основе ПЛИС

Дисциплина: Архитектура электронно-вычислительных машин

Студент

ИУ7-51Б

(Группа)

(Подпись, дата)

Е.А.Варламова

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

А.Ю.Попов

(И.О. Фамилия)

Москва, 2021

Цель работы: изучение основ построения микропроцессорных систем на ПЛИС. В ходе работы необходимо ознакомиться с принципами построения систем на кристалле (СНК) на основе ПЛИС, получить навыки проектирования СНК в САПР Altera Quartus II, выполнить проектирование и верификацию системы с использованием отладочного комплекта Altera DE1Board.

1. Функциональная схема разрабатываемой системы на кристалле.

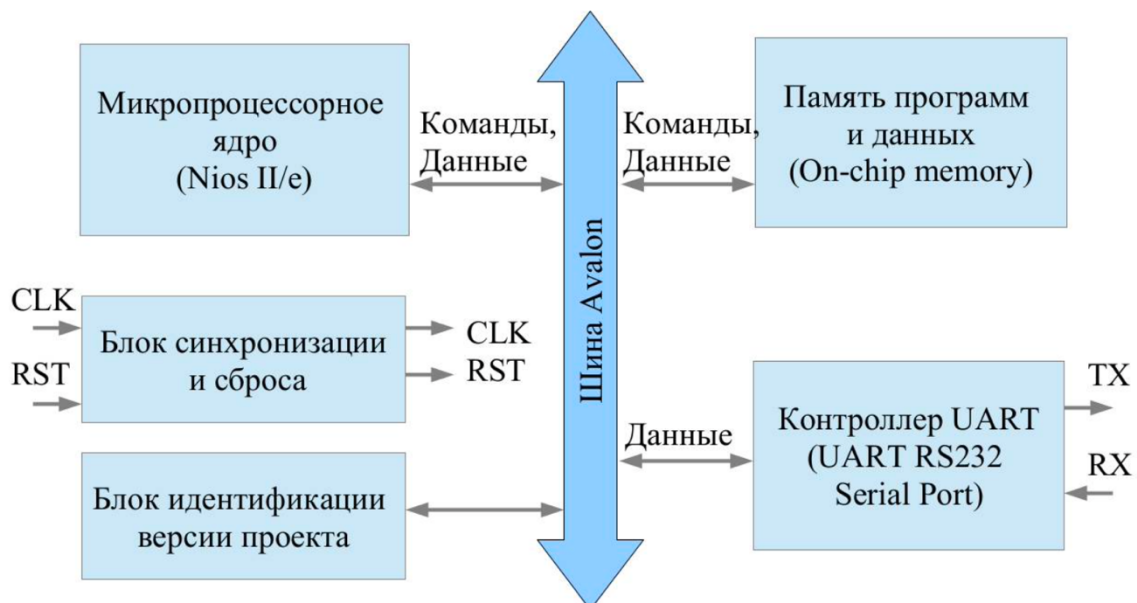


Рисунок 1 Функциональная схема разрабатываемой системы на кристалле

Система на кристалле состоит из следующих блоков:

- Микропроцессорное ядро Nios II/e выполняет функции управления системой.
- Внутренняя оперативная память СНК, используемая для хранения программы управления и данных.
- Системная шина Avalon обеспечивает связность всех компонентов системы.
- Блок синхронизации и сброса обеспечивает обработку входных сигналов сброса и синхронизации и распределение их в системе. Внутренний сигнал сброса синхронизирован и имеет необходимую для системы длительность.
- Блок идентификации версии проекта обеспечивает хранение и выдачу уникального идентификатора версии, который используется программой управления при инициализации системы.
- Контроллер UART обеспечивает прием и передачу информации по интерфейсу RS232.

2. Скриншоты маршрута проектирования

2.1. Модуль в QSYS

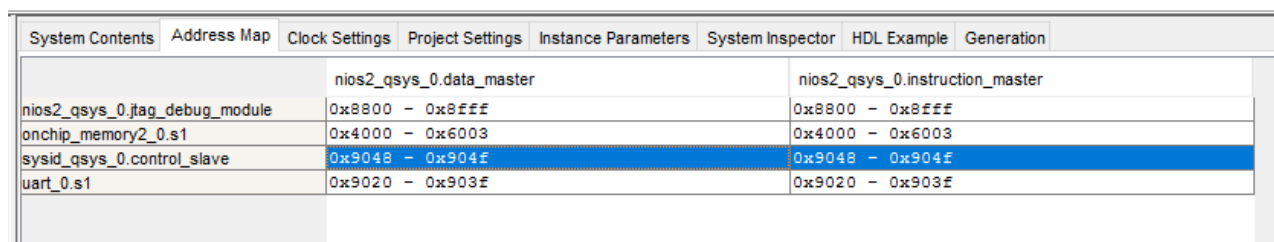
1. Был создан новый модуль Qsys.
2. Установлена частота внешнего сигнала синхронизации 50 000 000 Гц.
3. Добавлен в проект модуль синхронизируемого микропроцессорного ядра Nios2.
4. Добавлен в проект модуль ОЗУ программ и данных.
5. Добавлены компоненты Avalon System ID, Avalon UART.
6. Создана сеть синхронизации и сброса системы.
7. Сигналы TX и RX экспортированы во внешние порты.
8. Назначены базовые адреса устройств.

Итог выполненных действий показан на рисунке 2. На рисунке 3 показана таблица распределения адресов.

cuments\kate\nios.qsys)

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

Рисунок 2 Модуль Qsys



System Contents	Address Map	Clock Settings	Project Settings	Instance Parameters	System Inspector	HDL Example	Generation
				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

2.2. Назначение портам проекта контактов микросхемы

Был выполнен синтез проекта. Результат показан на рисунке 4.

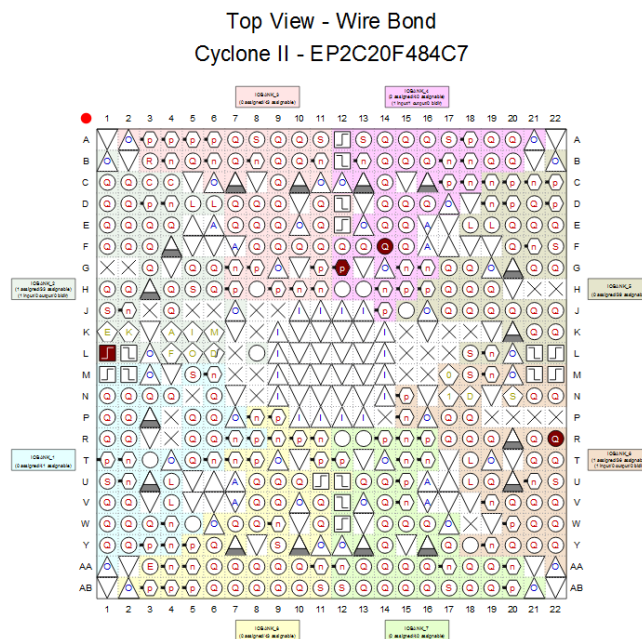


Рисунок 4 Pin Planner

Далее были назначены портам проекта контакты микросхемы в соответствии с таблицей 1, после чего был выполнен повторный синтез проекта.

Таблица 1 Назначение контактов микросхемы портам проекта

Сигнал	Контакт
clk	L1
reset	R22
uart0_rxd	F14
uart0_txd	G12

2.3. Создание проекта Nios2

В файл `hello_world_small.c` был добавлен код эхо-программы приема-передачи по интерфейсу RS232, представленный на листинге 1. Также был создан образ ОС HAL с драйверами устройств, используемых в аппаратном проекте.

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

```
#include "sys/alt_stdio.h"

int main()
{
    char ch;
    alt_putstr("Hello from System on Chip\n");
    alt_putstr("Send any character\n");
    /* Event loop never exits. */
    while (1) {
        ch=alt_getchar();
        alt_putchar(ch);
    }
    return 0;
}
```

После успешной сборки и выполнения код программы был доработан: были добавлены строки, передающие по UART значение SystemID в виде четырех байт символов в ASCII формате.. Результат доработки представлен на листинге 2.

Листинг 2. Код эхо-программы приема-передачи по интерфейсу RS232

```
#include "sys/alt_stdio.h"
#include "system.h"
#include "altera_avalon_sysid_qsys.h"
#include "altera_avalon_sysid_qsys_regs.h"

int main()
{
    char ch;
    alt_putstr("Hello from System on Chip\n");
    alt_putstr("Send any character\n");
    int id = IORD_ALTERA_AVALON_SYSID_QSYS_ID(SYSID_QSYS_0_BASE);
    char a[6];
    int i = 1;
    while (id)
    {
        a[4 - i] = '0' + id % 10;
        id /= 10;
        i++;
    }
    a[4] = '\n';
    a[5] = '\0';
    for (int i = 0; i < 5; i++)
        alt_putchar(a[i]);
    /* Event loop never exits. */
    while (1) {
        ch=alt_getchar();
        alt_putchar(ch);

    }
    return 0;
}
```

Результат выполнения программы показан на рисунке 5.

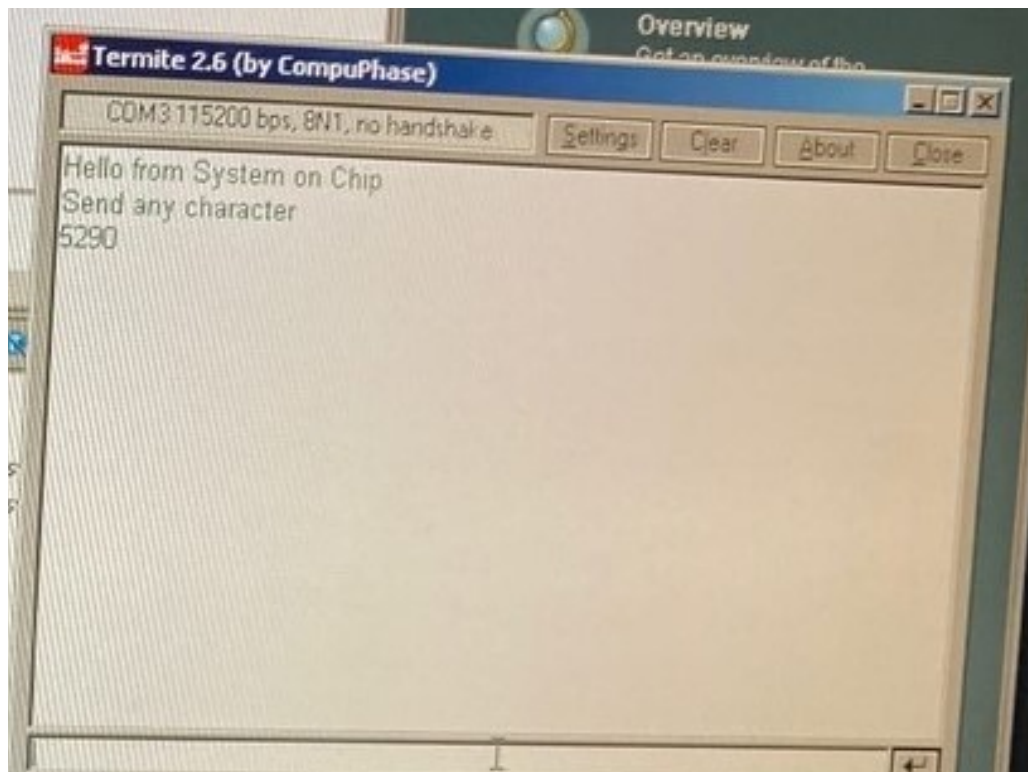


Рисунок 5 Результат выполнения доработанного кода

Вывод

В ходе данной лабораторной работы были изучены основы построения микропроцессорных систем на ПЛИС, получены навыки проектирования СНК в САПР Altera Quartus II, также были выполнены проектирование и верификация системы с использованием отладочного комплекта Altera DE1Board.

Поставленная **цель** достигнута.