

Упражнение 2

Цель упражнения:

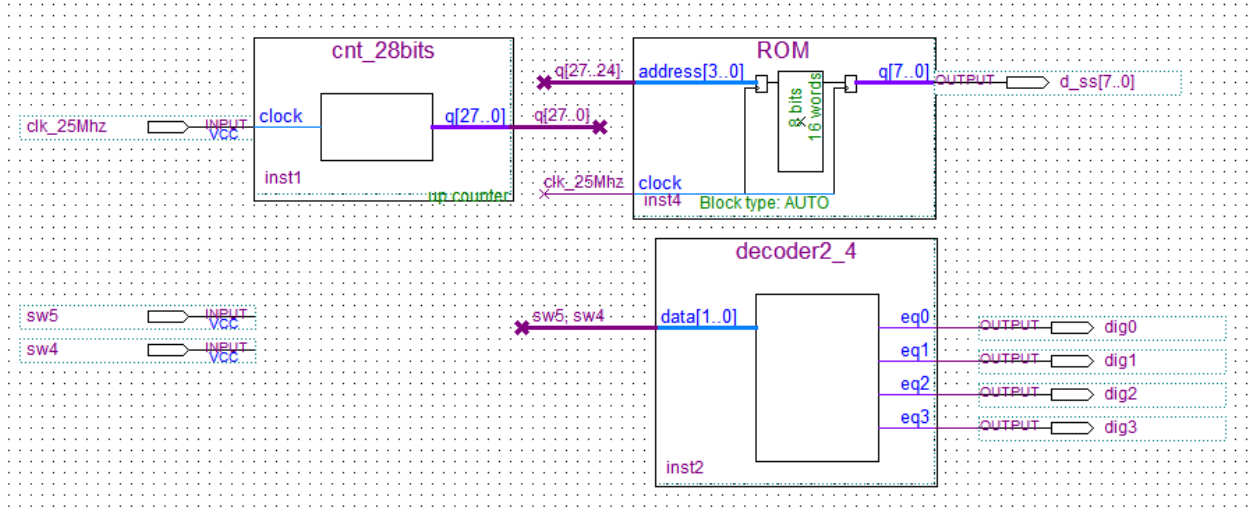
Знакомство с базовыми возможностями схемного ввода в пакете Quartus II и возможностями редактора начального содержимого модуля памяти.

В рабочей папке упражнения 2 – папке lab2, уже созданы:

- excel файл таблицей преобразования двоичного кода в 7-сегментный – bin_7seg.xls
- файл с настройками выводов СБИС для платы miniDiLab-CIV – файл lab2.qsf

Алгоритм работы проекта:

Схема проекта



Алгоритм работы:

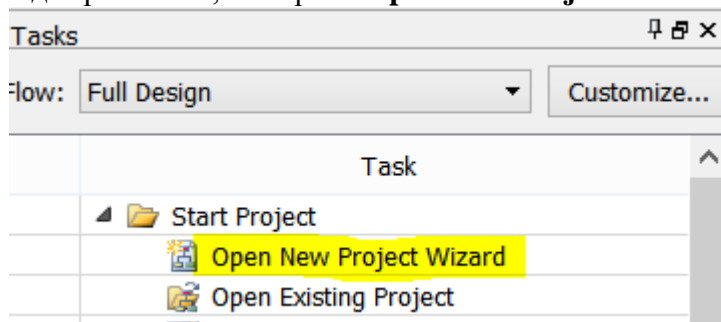
- входная частота clk_25Mhz (25 МГц) делится на 28-разрядном счетчике делителе (cnt_28bits)
- 4 старших разряда счетчика поступают на преобразователь двоичного кода в 7-сегментный (bin_7seg), реализованный на модуле памяти ROM, а с его выхода на выход (на 4разрядный 7-сегментный индикатор).
- переключатели sw5, sw4 позволяют выбрать разряд 4-разрядного 7-сегментного индикатора, на котором будут отображаться данные с выхода преобразователя двоичного кода в 7-сегментный

Задачи:

- Часть 1 – Создание проекта
- Часть 2 – Создание mif файла и экземпляра модуля памяти ROM
- Часть 3 – Создание 28 разрядного экземпляра счетчика
- Часть 4 – Создание экземпляра декодера 2=>4
- Часть 5 – Схемный ввод проекта
- Часть 6 – Компиляция проекта
- Часть 7 – Программирование СБИС и проверка работы проекта на плате.

Часть 1 – Создание проекта

1. Запустите пакет QuartusII
2. В окне **File** менеджера пакета, выберите **Open New Project Wizard...**



3. На экране появится окно введения - **Introduction** (если оно небыло отключено). Нажмите кнопку **next**.
4. В появившемся окне введите следующие данные:

Раздел	Что ввести
What is the working directory for this project? Рабочая папка (с помощью браузера найдите рабочую папку проекта)	... \ lab2
What is the name of this project? Имя проекта	lab2
What is the name of the top-level design entity for this project? Имя модуля верхнего уровня в иерархии проекта.	lab2

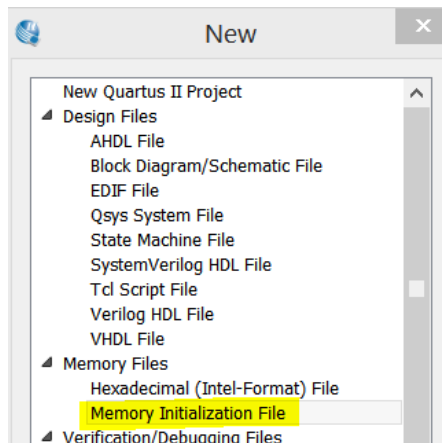
5. Нажмите кнопку **Next**.
6. В следующем окне нажмите **Next** еще раз.
7. В окне **Family & Device Setting**[page3 of 5]:
 - в разделе **Family** выберите **CycloneIV**.
 - в разделе **Available devices** выберите **СБИС EP4C6E22C8**.
8. Нажмите кнопку **Next**.
9. В окне **EDA Tool Setting** [page 4 of 5] оставьте все без изменения и нажмите кнопку **Next**.
10. Появится окно **Summary** [page 5 of 5], в котором указаны установки, заданные Вами для создаваемого проекта. Проверьте их. Если все правильно, то нажмите кнопку **Finish**. В противном случае, вернитесь назад, нажав (возможно несколько раз) кнопку **Back**.

Проект создан.

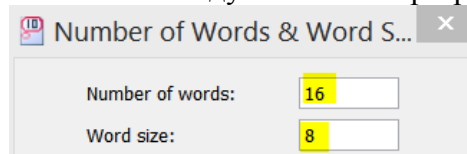
Часть 2 – создание mif файла и экземпляра модуля памяти ROM

1. Создайте новый mif файл:

- Команда File=>New



- Задайте количество слов -16- модуля памяти и разрядность -8



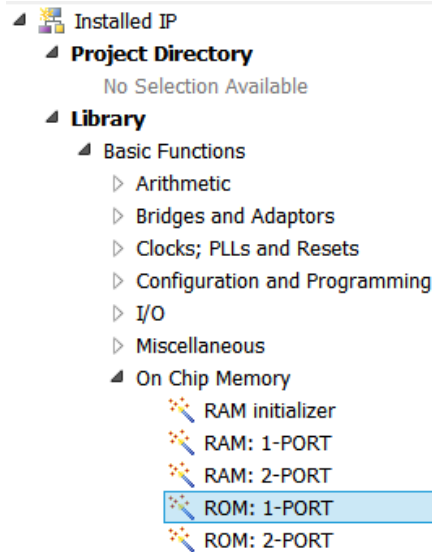
- Установите систему счисления данных – Binary; отобразите массив памяти в виде 1 колонки (1 Cell per Row).
- Откройте bin_7seg.xls файл и скопируйте содержимое колонки «7сегментный код» в созданный mif файл

адрес	7сегментный код	символ
0	0111111	0
1	0000110	1
2	1011011	2
3	1001111	3
4	1100110	4
5	1101101	5
6	1111101	6
7	0000111	7
8	1111111	8
9	1101111	9
10	1110111	A
11	1111100	B
12	0111001	C
13	1011110	D
14	1111001	E
15	1110001	F

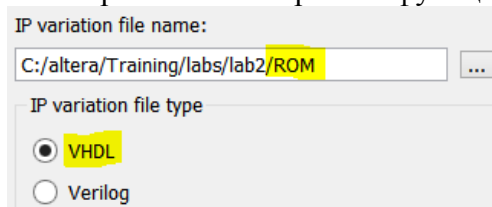
Addr	+0	ASCII
0	00111111	?
1	00000110	.
2	01011011	[
3	01001111	O
4	01100110	f
5	01101101	m
6	01111101	}
7	00000111	.
8	01111111	.
9	01101111	o
10	01110111	w
11	01111100	
12	00111001	9
13	01011110	^
14	01111001	y
15	01110001	q

- Сохраните файл под именем bin_7seg.mif .

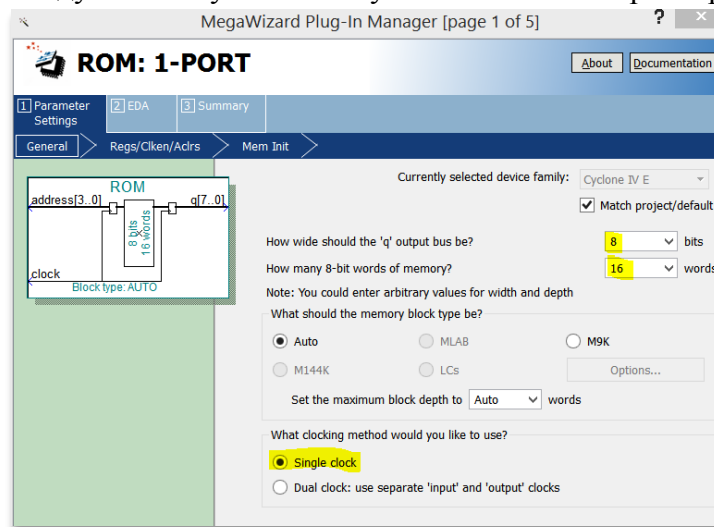
- В окне IP Catalog выберите мегафункцию ROM:1-PORT и нажмите кнопку ADD.



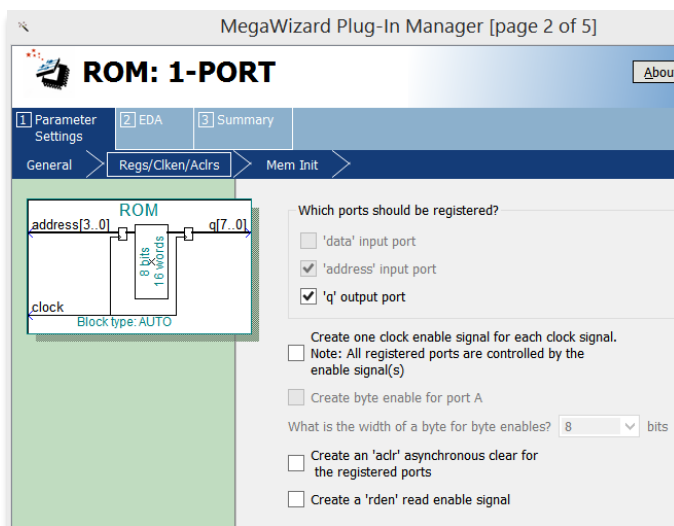
- В появившемся окне задайте имя создаваемого экземпляра мегафункции (ROM) и укажите язык для описания настроек экземпляра мегафункции (VHDL).



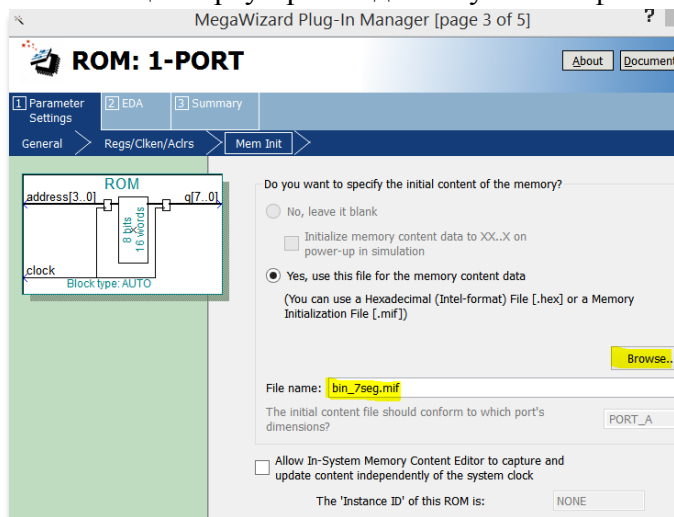
- Нажмите кнопку – ОК, запустится помощник MegaWizard Plug-in Manager.
- В окне настроек модуля ROM установите указанные ниже параметры



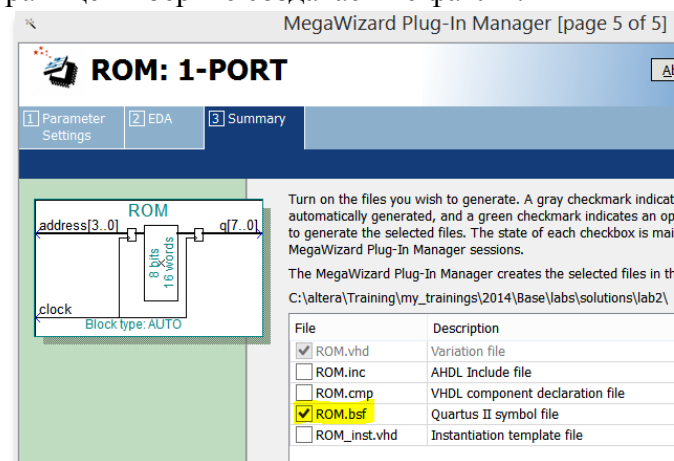
- Нажмите кнопку Next
- В появившемся окне оставьте все без изменений



8. Нажмите кнопку Next
9. В появившемся окне с помощью браузера найдите и укажите файл bin_7seg.mif



10. Нажмите кнопку Next
11. Нажмите кнопку Next
12. На появившейся странице выберите создаваемые файлы:

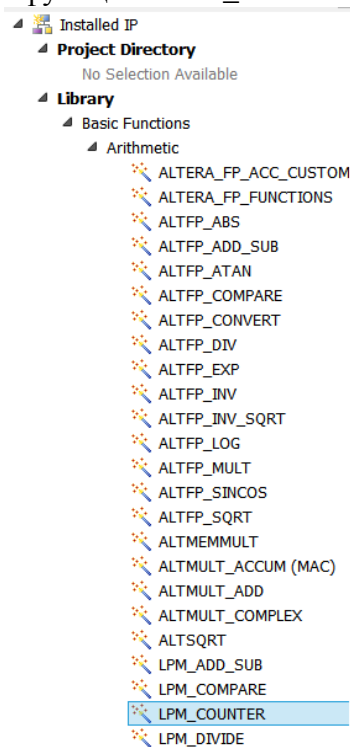


13. Нажмите кнопку Finish.

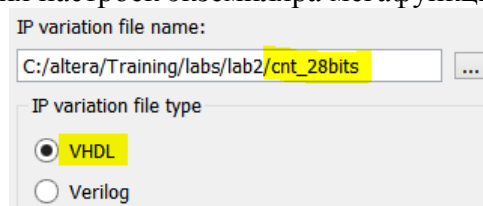
Экземпляр модуля ROM создан и с ним связан файл начального содержимого.

Часть 3 - создание 28 разрядного экземпляра счетчика

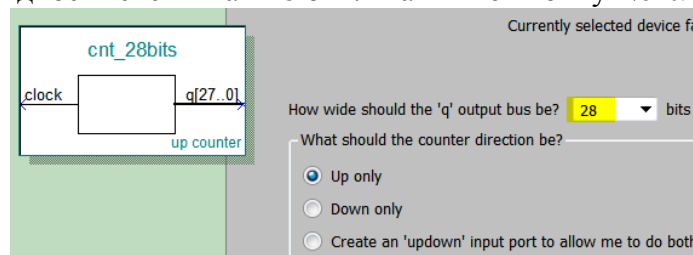
1. В окне IP Catalog выберите мегафункцию LPM_COUNTER и нажмите кнопку ADD.



2. В появившемся окне задайте имя создаваемого экземпляра мегафункции (cnt_28bits) и укажите язык для описания настроек экземпляра мегафункции (VHDL).



3. Нажмите кнопку – ОК, запустится помощник MegaWizard Plug-in Manager.
4. Установите разрядность счетчика - 28 бит. Нажмите кнопку **Next**.



5. В следующем окне нажмите кнопку **Next**.
6. В следующем окне нажмите кнопку **Next**.
7. Появится окно **Simulation Libraries**. Нажмите кнопку **Next**.
8. В появившемся окне укажите файлы, которые **MegaWizard** должен создать:

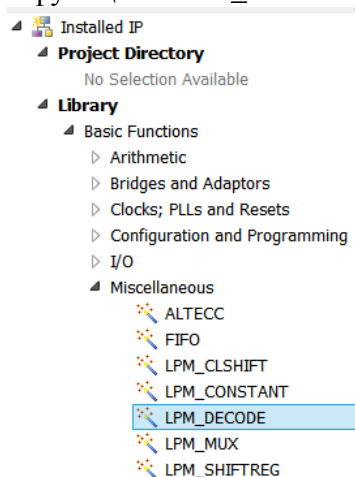
<input checked="" type="checkbox"/>	cnt_28bits.vhd	Variation file
<input type="checkbox"/>	cnt_28bits.inc	AHDL Include file
<input type="checkbox"/>	cnt_28bits.cmp	VHDL component declaration file
<input checked="" type="checkbox"/>	cnt_28bits.bsf	Quartus II symbol file
<input type="checkbox"/>	cnt_28bits_inst....	Instantiation template file

Нажмите кнопку **Finish**.

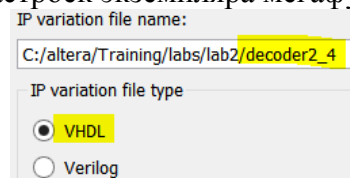
Экземпляр счетчика создан.

Часть 4 - создание экземпляра декодера 2=>4

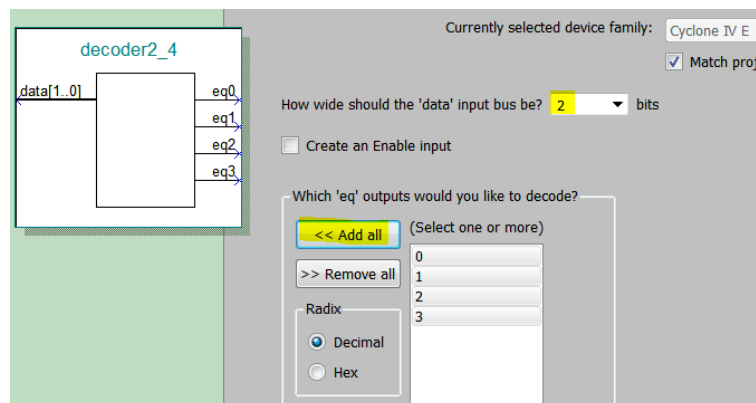
1. В окне IP Catalog выберите мегафункцию LPM_COUNTER и нажмите кнопку ADD.



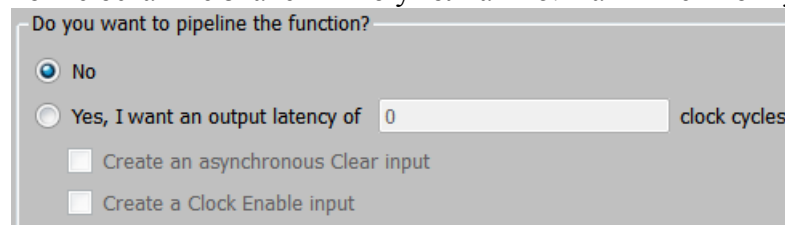
2. В появившемся окне задайте имя создаваемого экземпляра мегафункции (decoder2_4) и укажите язык для описания настроек экземпляра мегафункции (VHDL).



3. Нажмите кнопку – ОК, запустится помощник MegaWizard Plug-in Manager.
4. Установите разрядность входа - 2 бит. Нажмите кнопку **Add all**. Нажмите кнопку **Next**.



5. В следующем окне оставьте значения по умолчанию. Нажмите кнопку **Next**.



6. Появится окно **Simulation Libraries**. Нажмите кнопку **Next**.
7. В появившемся окне укажите файлы, которые **MegaWizard** должен создать:

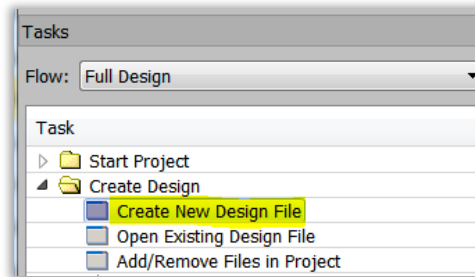
File	Description
<input checked="" type="checkbox"/> decoder2_4.vhd	Variation file
<input type="checkbox"/> decoder2_4.inc	AHDL Include file
<input type="checkbox"/> decoder2_4.cmp	VHDL component declaration file
<input checked="" type="checkbox"/> decoder2_4.bsf	Quartus II symbol file
<input type="checkbox"/> decoder2_4_ins...	Instantiation template file

Нажмите кнопку **Finish**.

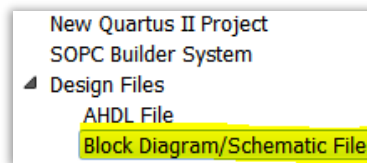
Экземпляр декодера создан.

Часть 5 - Создание схемы

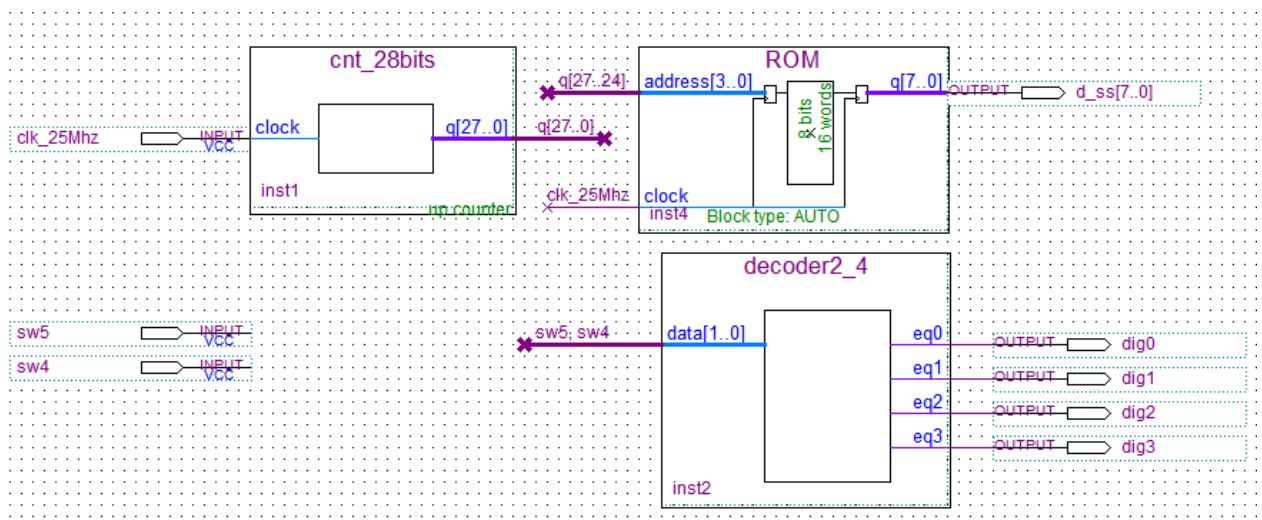
1. Запустите задачу Create New Design File в окне задач



2. Укажите тип создаваемого файла **Block Diagram/Schematic File**. Нажмите OK.



3. Выполните команду: меню **File->Save As** и сохраните файл как **lab2.bdf**
4. Схема, которая должна быть создана, изображена ниже.



5. Сохраните схему под именем lab2.bdf.

Схема создана.

Часть 6 – Компиляция проекта

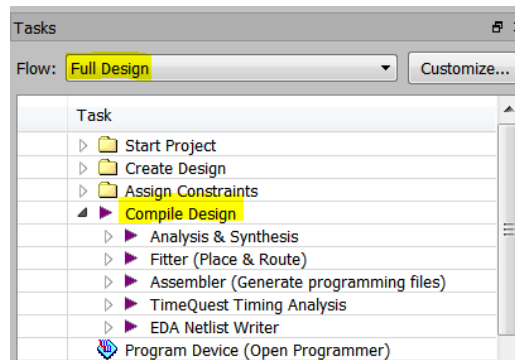
1. Назначение опций компилятора (практически все опции имеют значения по умолчанию) и номеров выводов СБИС для платы miniDiLaB-CIV выполнены для Вас и хранятся в файле **lab2.qsf**. Откройте этот файл в любом текстовом редакторе.
2. Раздел Pin & Location Assignments файла, содержащий привязку выводов проекта к выводам микросхемы EP4C6E22C8, приведен ниже (сравните с открытым файлом):

```
# Pin & Location Assignments
# =====
set_location_assignment PIN_64 -to pba
set_location_assignment PIN_58 -to pbb
set_location_assignment PIN_88 -to sw7
set_location_assignment PIN_89 -to sw6
set_location_assignment PIN_90 -to sw5
set_location_assignment PIN_91 -to sw4
set_location_assignment PIN_87 -to d_ss[0]
set_location_assignment PIN_133 -to d_ss[1]
set_location_assignment PIN_86 -to d_ss[2]
set_location_assignment PIN_77 -to d_ss[3]
set_location_assignment PIN_85 -to d_ss[4]
set_location_assignment PIN_76 -to d_ss[5]
set_location_assignment PIN_84 -to d_ss[6]
set_location_assignment PIN_75 -to d_ss[7]
set_location_assignment PIN_83 -to dig0
set_location_assignment PIN_74 -to dig1
set_location_assignment PIN_80 -to dig2
set_location_assignment PIN_73 -to dig3
set_location_assignment PIN_23 -to clk_25Mhz
```

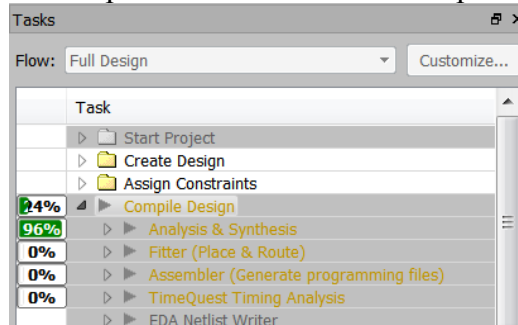
3. Часть раздела Fitter Assignments, в которой задаются:
 - режим работы не использованных выводов СБИС: как входы с pull-up резистором
 - стандарт сигнала для каждого входа/выхода
 приведена ниже (сравните с открытым файлом):

```
# Fitter Assignments
# =====
set_global_assignment -name DEVICE EP4CE6E22C8
set_global_assignment -name ERROR_CHECK_FREQUENCY_DIVISOR 1
set_global_assignment -name RESERVE_ALL_UNUSED_PINS_WEAK_PULLUP "AS INPUT TRI-STATED"
set_instance_assignment -name IO_STANDARD "2.5 V" -to pba
set_instance_assignment -name IO_STANDARD "2.5 V" -to pbb
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sw7
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sw6
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sw5
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to sw4
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to dig3
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to dig2
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to dig1
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to dig0
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[7]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[6]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[5]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[4]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[3]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[2]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[1]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to d_ss[0]
set_instance_assignment -name IO_STANDARD "3.3-V LVCMOS" -to clk_25Mhz
set_global_assignment -name STRATIX_DEVICE_IO_STANDARD "2.5 V"
```

4. В окне задач (Tasks) выберите процедуру Full Design и двойным щелчком левой клавиши мыши по команде Compile Design запустите полную компиляцию проекта.



5. Окно задач (Tasks) будет отображать ход выполнения процедуры компиляции.

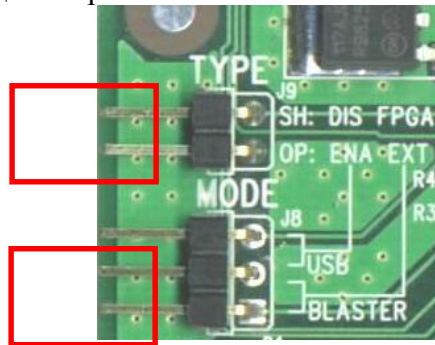


В процессе полной компиляции проекта осуществляется:

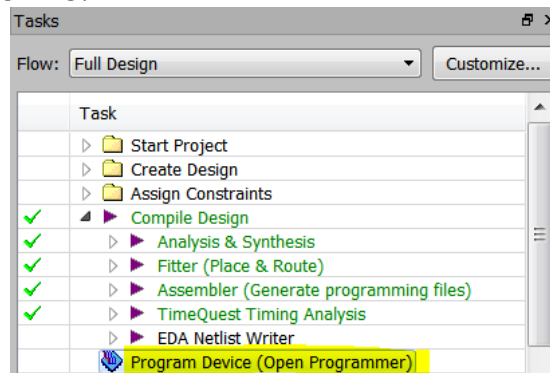
- ✓ проверка синтаксиса,
- ✓ синтез с оптимизацией занимаемой площади и быстродействия проекта,
- ✓ трассировка и СБИС с оптимизацией занимаемой площади и быстродействия проекта,
- ✓ получение файла для конфигурирования СБИС – *prof(sof)* файл,
- ✓ получение модели с временными параметрами реализованной СБИС,
- ✓ временной анализ,
- ✓ Формирования файла с детальным отчетом о всех этапах компиляции проекта.

Часть 7 – Конфигурирование СБИС и проверка проекта на плате

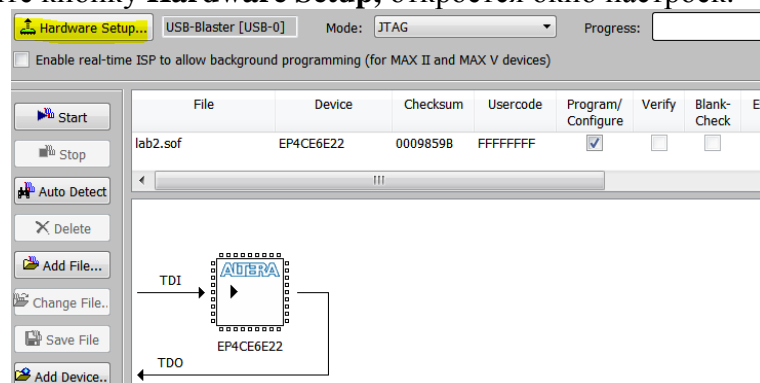
1. На плате miniDiLaB-CIV установите джамперы следующим образом:
 - a. Соедините выводы разъема “TYPE”
 - b. Соедините выводы 1-2 разъема “MODE”



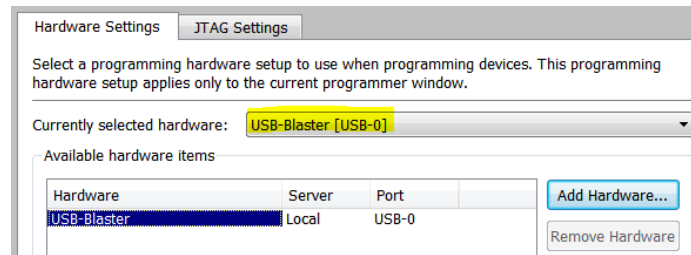
2. Подсоедините входящий в комплект поставки USB кабель A-miniB к USB 2.0 порту компьютера (должен обеспечивать ток до 500мА), а затем к плате miniDiLaB-CIV.
3. Подсоедините к плате (разъем PR_A) модуль 7-сегментного индикатора
4. Включите плату miniDiLaB-CIV : переключатель Power
5. В окне задач (Tasks) выберите процедуру Full Design и двойным щелчком левой клавиши мыши по команде Program Device запустите приложение, управляющее конфигурированием СБИС.



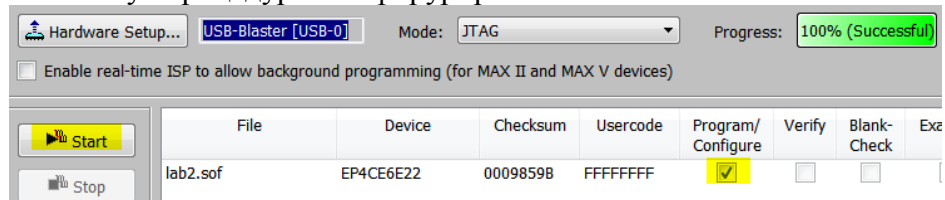
6. Откроется окно управления конфигурированием СБИС.
7. Для установки интегрированного на плату miniDiLaB-CIV средства конфигурирования СБИС нажмите кнопку **Hardware Setup**, откроется окно настроек.



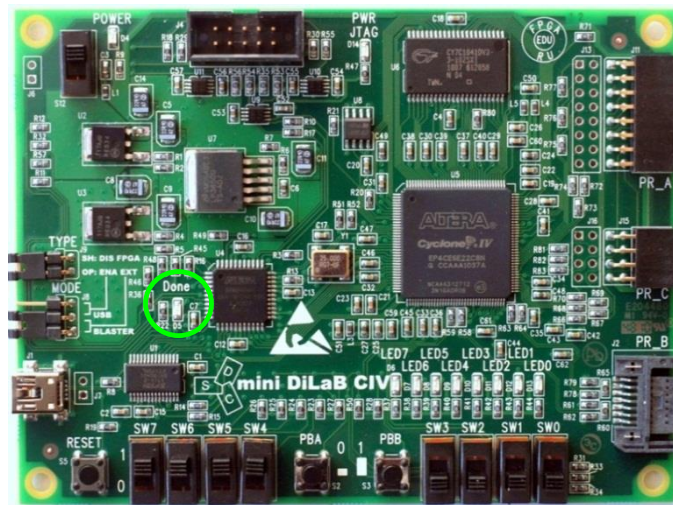
8. В разделе **Available hardware items** выберите (двойным щелчком левой клавиши мыши) USB-Blaster. Нажмите кнопку **Close**.



9. Включите опцию **Program/Configure** и нажмите кнопку **Start**. В окне Progress будет отображаться статус процедуры конфигурирования СБИС.



10. Когда СБИС будет запрограммирована на плате miniDiLaB-CIV загорится зеленый светодиод – “Done”.



11. Проверьте работу проекта:
- Переключатели sw5, sw4 установите в положение 00: на первый (самый правый) разряд 7-сегментного индикатора должны последовательно выводиться 0,1,2,3,...9, a,b,c,d,e,f,0...
 - Повторите для всех возможных комбинаций переключателей sw5, sw4: 00 – активный первый разряд; 01 – активный второй разряд; 10 – активный третий разряд; 11 – активный четвертый разряд.

Лабораторная работа завершена.