
Using MicroC/OS-II RTOS with the Nios II Processor

Tutorial

1. Выполнение учебного руководства

Это руководство ознакомит Вас с процессом разработки в NIOS II IDE и MicroC/OS-II. NIOS II IDE предлагает разработчикам богатую платформу разработки приложений для NIOS II. NIOS II IDE позволяет вам интегрировать операционную систему реального времени MicroC/OS-II, предоставляя разработчикам возможность быстро собрать приложения на MicroC/OS-II для процессора NIOS II. Это руководство обеспечивает пошаговые инструкции для создания простой базовой программы на MicroC/OS II RTOS.

Для полного описания MicroC/OS II для процессора NIOS II, обращайтесь к разделу [MicroC/OS II Real-Time Operating System](#) в [NiosII Software Developer's Handbook](#).

Программные и аппаратные требования

Это руководство требует следующие аппаратные и программные средства:

- Quartus II версия 11.0 или выше,
- Одна из списка отладочных плат Altera, показанных на сайте Altera на странице [Nios II Ethernet Standard Design Example](#).

Страница содержит информацию о примере и включает файлы аппаратного проекта для разных отладочных плат Altera.

- Altera USB-Blaster или подобный,
- Файлы аппаратного проекта для Nios II Ethernet Standard Design Example.
- Файлы программного проекта для этого обучающего руководства.

Получение аппаратных файлов проекта для примера

Для получения аппаратных файлов проекта для этого учебного руководства выполните следующие шаги:

1. Перейти на страницу Nios II Ethernet Standard Design Example на сайте Altera.
2. Загрузить .zip файл пример проекта [Nios II Ethernet Standard Design Example](#), соответствующий вашей плате.
3. Распаковать файл в директорию проекта выбранного вами. Учебное руководство предлагает путь к директории как <директория вашего аппаратного проекта>.

Получение программных файлов проекта для примера

Для получения программных файлов проекта для этого учебного руководства выполните следующие шаги:

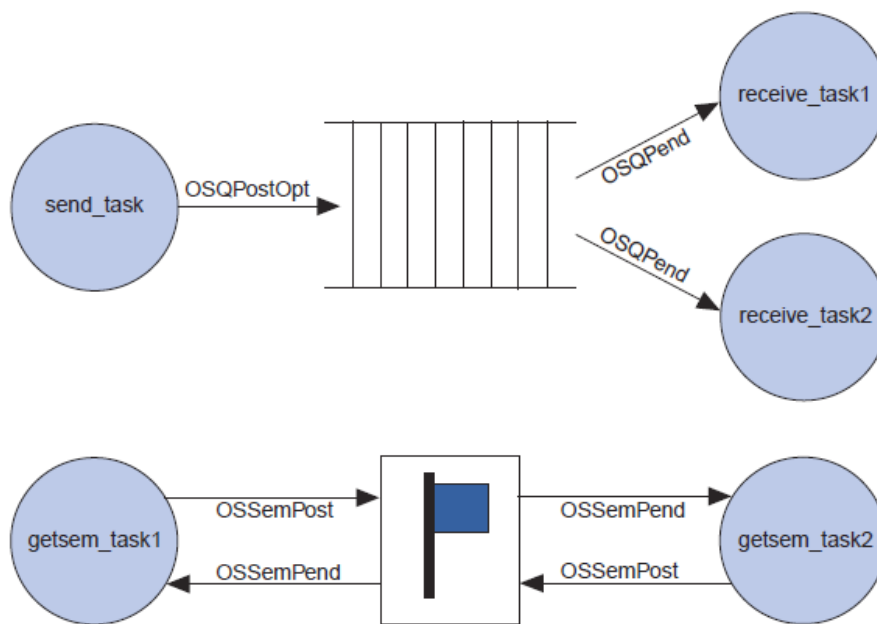
1. Загрузите файл uCOS_II_Tutorial.zip. Вы можете найти этот файл на странице [MicroC/OS-II RTOS with the Nios II Processor](#) сайта Altera или рядом с этим руководством на странице [Literature: Nios II Processor](#).
2. Распаковать файл в директорию на вашей системе. Учебное руководство предлагает путь к директории как <директория примеров ваших программ>.

Когда Вы извлекаете файлы из uCOS_II_Tutorial.zip, убедитесь, что в указании пути нет пробелов.

Обучающий проект MicroC/OS II

Пример простого проекта для упражнения некоторых базовых свойств операционной системы. Рис 1-1 это упрощенная диаграмма приложения.

Figure 1-1. Tutorial Design Example



Как показано на рис 1-1 проект имеет пять активных задач. Send_task помещается в очередь сообщений с увеличением данных. Receive_task1 и receive_task2 периодически достаются из очереди сообщений. Getsem_task1 и getsem_task2 совместно используют общий ресурс, защищенный семафором. Проект также имеет две задачи, не показанные в рис. 1-1: одна для начальной настройки и вторая для печати информации состояния.

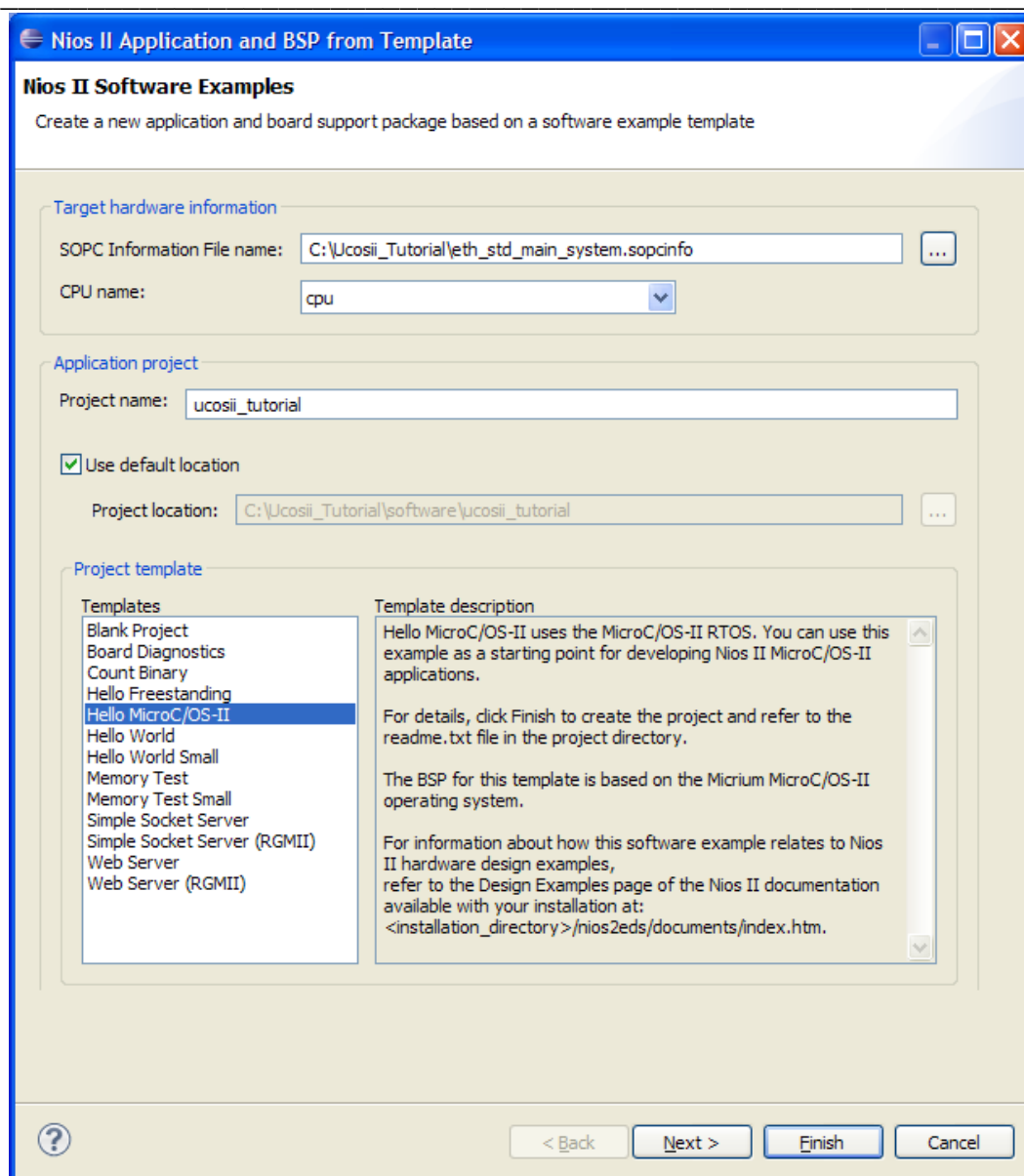
Процессы для создания программного образа MicroC/OS-II для процессора Nios II включают следующие шаги:

1. Создать новый проект в Nios II IDE.
 2. Сконфигурировать программную библиотеку Nios II.
 3. Собрать и выполнить программный проект Nios II.
- Эти шаги детально описаны в следующих разделах.

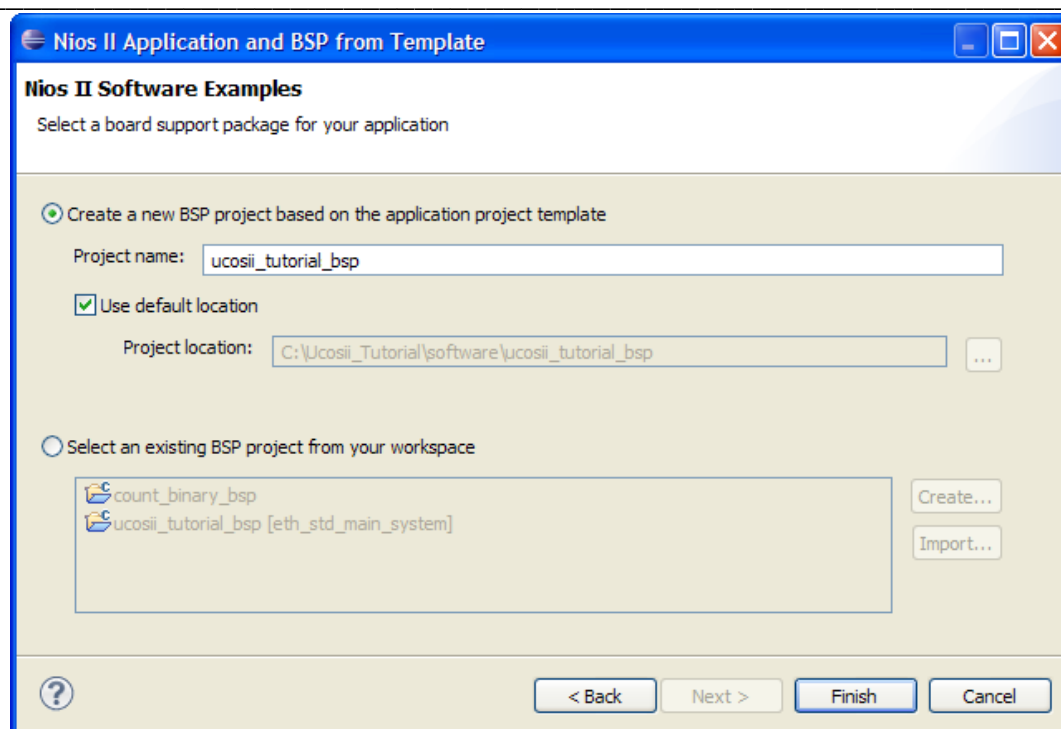
Создание программного проекта MicroC/OS-II

В этом разделе Вы создаете новое приложение Nios II и BSP проект, используя шаблон проекта “**Hello MicroC/OS-II**” доступного в Nios II SBT для Eclipse.

1. Запустите Nios II SBT для Eclipse, выполнив следующие шаги в зависимости от Вашей операционной среды:
 - В операционной среде Windows, в меню Start, пункт Programs > Altera > Nios II EDS <version>, и выбрать Nios II <version> Software Build Tools for Eclipse.
 - В операционной среде Linux, в командной строке, смените директорию на <Nios II EDS install path>, введите `./sdk_shell` для запуска командной строки Nios, затем введите `./eclipse-nios2`.
2. Если появится диалоговое окно **Workspace Launcher**, нажмите **OK** для выбора средства запуска по умолчанию.
3. В меню **File**, в разделе **New**, нажмите на **Nios II Application and BSP from Template**. Откроется первая страница мастера **Nios II Application and BSP from Template**, как показано на рис. 1-2.



4. Под **Target hardware information**, просмотрите <your hardware design directory> и выберите **eth_std_main_system.sopcinfo**. Nios II SBT под Eclipse запишет в CPU name имя процессора найденное в **eth_std_main_system.sopcinfo**, как показано на рис.1-2.
5. Ввести в раздел **Project name** имя **ucoseii_tutorial**, как имя Вашего проекта.
6. Установить **Use default location**.
7. В списке **Templates**, выбрать Hello MicroC/OS-II.
8. Выбрать **Next**. Мастер откроет второе окно Nios II Application and BSP from Template, как показано на Рис. 1-3.

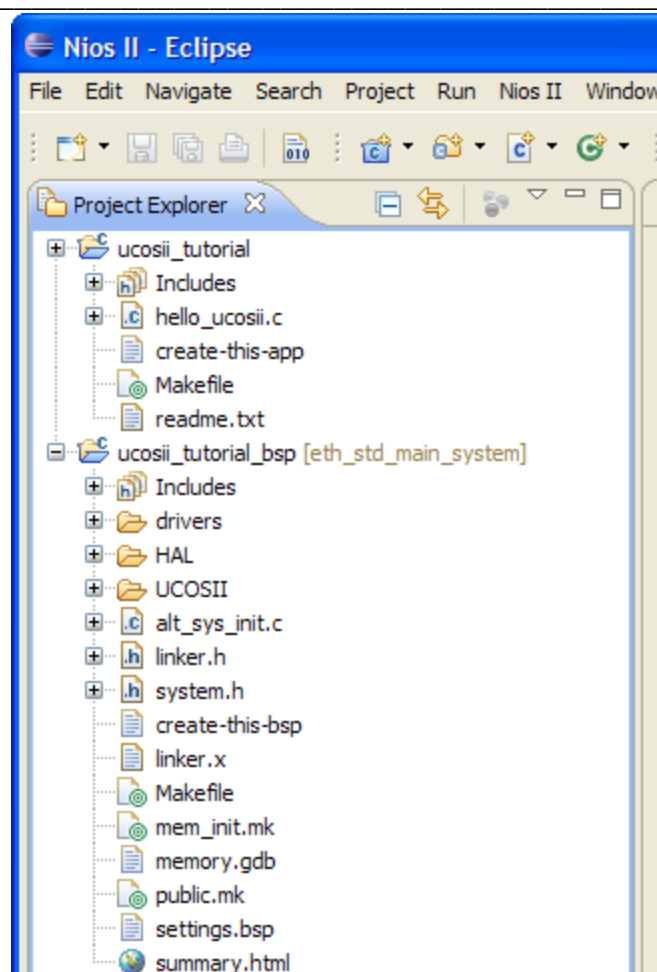


9. Убедитесь, что **Create a new BSP project based on the application project template** выбрано и выделен **Use default location**.

В этом обучающем руководстве, Вы используете шаблон проекта **Hello MicroC/OS-II**, с простым и быстро созданным BSP, основанным на операционной системе MicroC/OS-II. Поэтому, если Вы хотите использовать свой BSP, выберите **Select an existing BSP project from your workspace** и используйте нижнюю часть из **Nios II Application and BSP from Template** мастера для выбора, создания и импорта вашего BSP. Убедитесь, что Ваш BSP основан на операционной системе MicroC/OS-II.

10. Выберите **Finish** для создания нового приложения и DSP.

11. Nios II SBT под Eclipse создается в двух папках, доступных в виде Project Explorer: `ucosii_tutorial` и `ucosii_tutorial_bsp`. Вид файлов созданных из проекта Hello MicroC/OS-II, в развернутом виде, отображен на рис. 1-4.



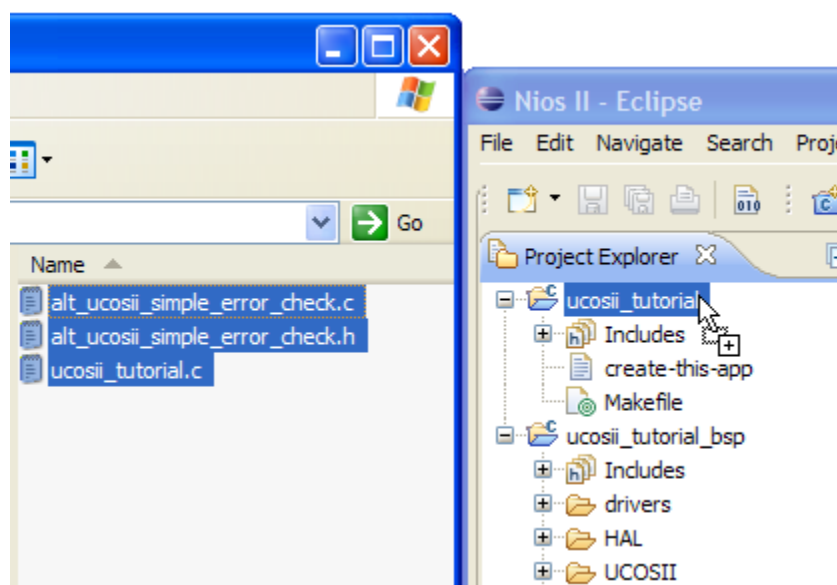
12. В проекте приложения, удалите **hello_ucosii.c** (выделить файл и выбрать **Delete**).

13. Добавить следующие файлы в проект:

- ucosii_tutorial.c
- alt_ucosii_simple_error_check.c
- alt_ucosii_simple_error_check.h

Выделить файлы в папке <your software examples directory>/software_source_files/ и переместить их в папку ucosii_tutorial, как показано на рис. 1-5.

Если предлагается выбрать копировать или связаться с файлами, скопируйте их.

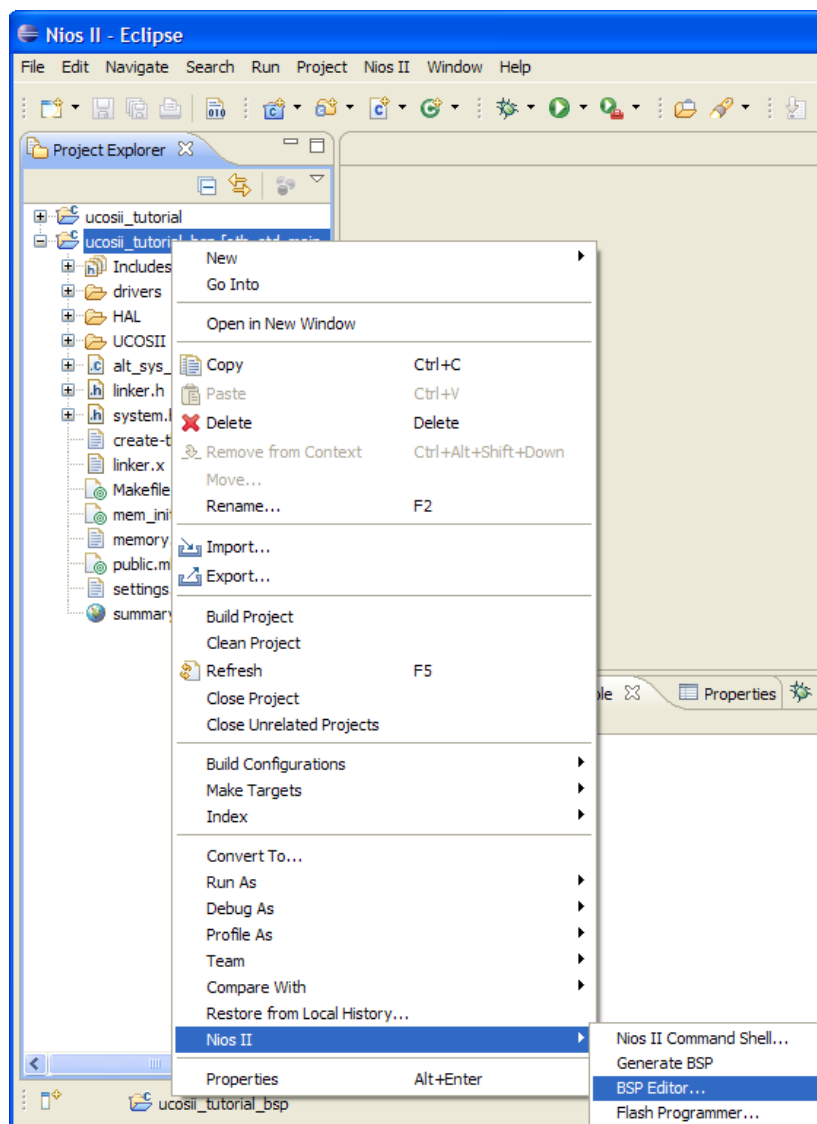


Вы создали приложение MicroC/OS-II. Конфигурация BSP в следующем разделе.

Конфигурация BSP

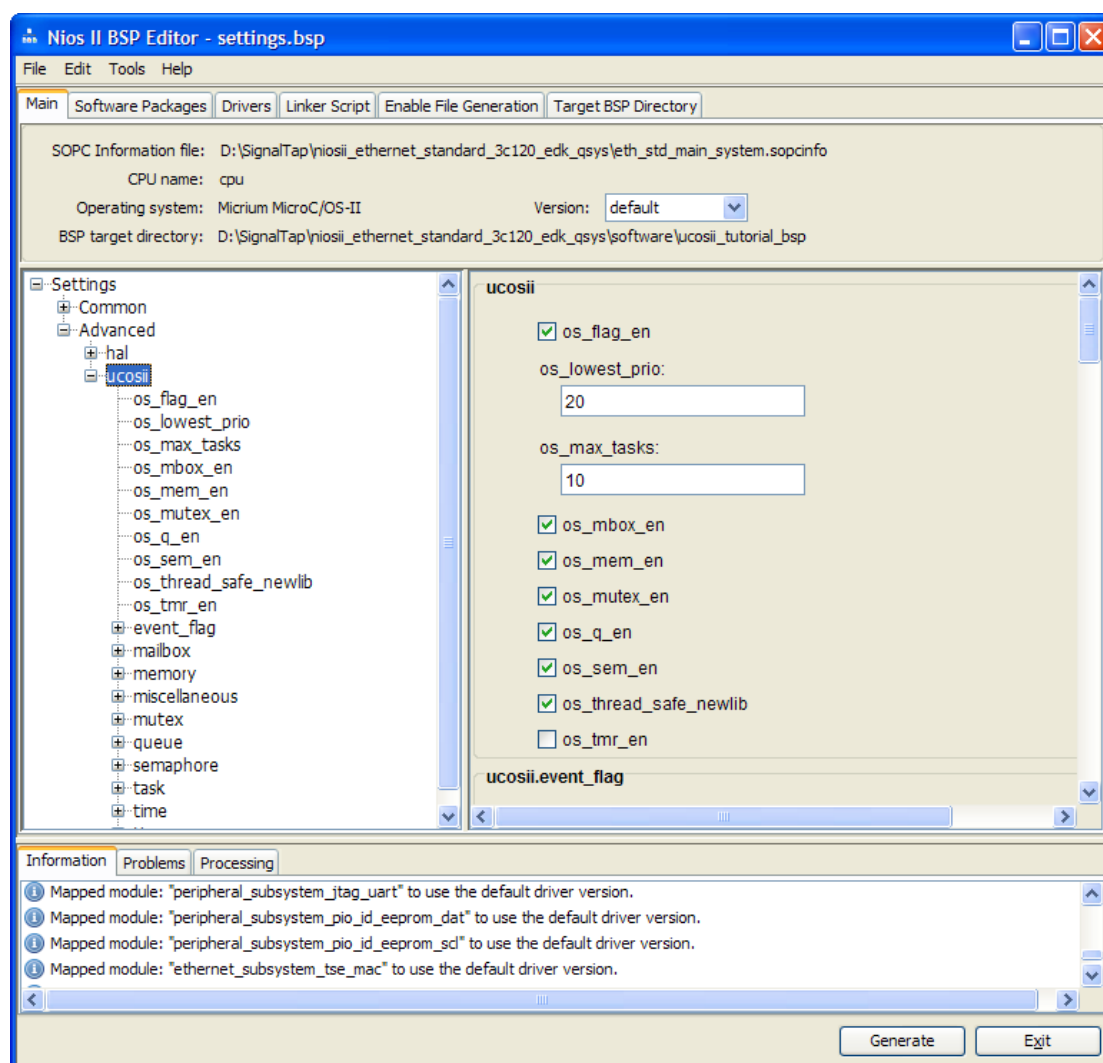
Затем Вы создаете новый BSP, используйте BSP Editor для настройки конфигурации BSP. Используйте настройки по умолчанию, если не меняете их.

1. Нажмите правую кнопку мыши на `ucosii_tutorial_bsp`, выберите пункт `Nios II`, и нажмите на `BSP Editor`, как показано на рис. 1-6. Откроется редактор Nios II BSP.



2. В редакторе BSP в закладке Main, выберите раскрывающееся дерево Settings, затем Advanced и ucosii.

Вы видите, что настройки BSP для MicroC/OS-II, имеют много конфигураций, как показано на рис. 1-7. Настройки решают, какие опции MicroC/OS-II будут включены в бинарный образ. Nios II SBT сохраняет значения настроек в файле system.h.



3. Чтобы видеть описания параметров настройки, выберите каждую настройку.

Для получения дополнительной информации о настройках MicroC/OS-II, смотрите в разделе MicroC/OS-II Real-Time Operating System в Nios II Software Developer's Handbook.

4. Развернуть настройки **Common**, развернуть и выбрать **hal**, и установите все stderr, stdin и stdout в peripheral_subsystem_jtag_uart.

5. Если Вы сделали любое изменение, сохраните BSP и выберите **Generate** для воссоздания файлов BSP.

6. Выберите **Exit** для закрытия редактора BSP.

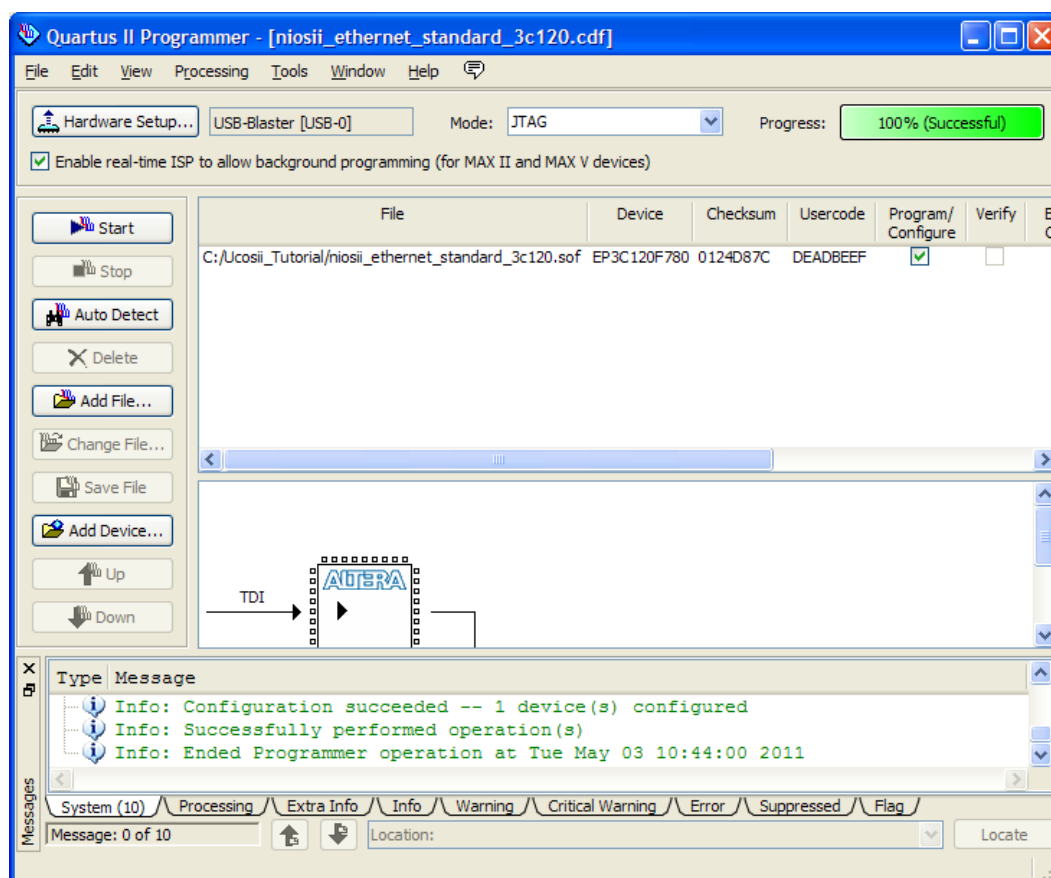
Вы закончили конфигурацию BSP. Вы готовы собрать и запустить пример программы, как описано в следующем разделе.

Запуск программного проекта MicroC/OS-II.

В этом разделе, Вы запустите пример на отладочной плате. Используя **Quartus II Programmer**, Вы конфигурируете плату с SRAM Object File (.sof). Используя Nios II SBT для Eclipse, Вы собираете приложение, загружаете Executable and Linking Format File (.elf), стартуете, и наблюдаете за выходными данными.

Выполнение обучающего программного проекта, делается следующими шагами:

1. В меню Nios II SBT для Eclipse, выберите Quartus II Programmer для запуска Quartus II Programmer, как показано на рис. 1-8.



2. Подайте напряжение на отладочную плату, правильно подсоедините USB кабель и откройте Quartus II Programmer.

3. Выберите **Hardware Setup** в верхнем левом углу Quartus II Programmer, для проверки настроек загрузочного кабеля. Появится диалоговое окно Hardware Setup.

4. В закладке **Hardware Settings**, выберите соответствующий кабель в Currently selected hardware (выбор текущего оборудования) списке. Если соответствующего кабеля нет в списке, Вы должны сначала установить драйвер для кабеля.

Для детальной информации о кабелях и драйверах, смотрите на сайте Altera страницу [Download Cables](#).

5. Выберите **Close**.

6. В меню File в Quartus II Programmer, выберите **Open**.

7. Перейдите на <your hardware design directory> и выберите niosii_ethernet_standard_<board>.sof.

8. Выберите **Open**, выберите .sof файл и вернитесь в Quartus II Programmer.

9. В строке таблицы для открытого .sof файла установите Program/Configure.

10. Выберите **Start** для конфигурации FPGA. При успешном завершении, Quartus II Programmer выведет на экран примерно следующие сообщения:

Info: Configuration succeeded -- 1 device(s) configured

Info: Successfully performed operation(s)

Info: Ended Programmer operation at Tue May 03 10:44:00 2011

11. Выйдите из Quartus II Programmer. Вы вернетесь в Nios II SBT для Eclipse. Если Quartus II Programmer спросит, желаете ли Вы сохранить изменения в chain1.cdf, выберите **No**.

12. Нажмите правой кнопкой мыши на ucosii_tutorial и выберите **Build Project**, для сборки программы. Сборка программы может занять несколько минут.

13. Нажмите правой кнопкой мыши на ucosii_tutorial, пункт **Run As**, и выберите Nios II Hardware, загрузите программу в плату и запустите её.

Если появится диалоговое окно **Run Configurations**, выберите закладку **Target Connection**. Затем, выберите **Refresh Connections** и **Apply**, пока соединение не будет установлено. Когда соединение установлено, выберите Run.

Загрузка завершена, окно **Nios II Console**, периодически обновляется функцией программы `print_status_task()`, как показано на примере 1-1.

Цифры отображаемые на экране могут быть другие.

```
*****
```

```
Hello From MicroC/OS-II Running on NIOS II. Here is the status:  
The number of messages sent by the send_task: 123  
The number of messages received by the receive_task1: 73  
The number of messages received by the receive_task2: 24  
The shared resource is owned by: getsem_task2  
The Number of times getsem_task1 acquired the semaphore 240  
The Number of times getsem_task2 acquired the semaphore 185
```

```
*****
```

Если появляющиеся данные в окне Nios II Console не корректны, Вам необходимо перезапустить Reset the selected target system в закладке Target Connection диалогового окна Run Configuration. Вы должны перезапустить Ваше приложение после изменения конфигурации запуска.

Поздравляем! Вы успешно сконфигурировали, собрали и запустили программу MicroC/OS-II.

Для большей информации об этом примере, смотрите файл `ucosii_tutorial.c`.

Для большей информации об операционной системе MicroC/OS-II смотрите раздел [MicroC/OS II Real-Time Operating System](#) в [NiosII Software Developer's Handbook](#).

1. Выполнение учебного руководства	1-1
Программные и аппаратные требования	1-1
Получение аппаратных файлов проекта для примера.....	1-1
Получение программных файлов проекта для примера	1-2
Обучающий проект MicroC/OS II.....	1-2
Создание программного проекта MicroC/OS-II.....	1-3
Конфигурация BSP	1-7
Запуск программного проекта MicroC/OS-II.....	1-10