

Интегрирование драйвера устройства в HAL

Nios II SBT может интегрировать драйверы устройств и пакеты программ, поставляемые Altera, сторонними разработчиками или непосредственно вами. В этой секции описывается, как подготовить драйверы устройств и пакеты программ так, чтобы генератор BSP распознал и добавил их в генерируемый BSP.

Вы можете использовать преимущества этого сервиса, когда создаёте драйвер устройства для одного из групповых моделей устройств HAL, или когда создаёте драйвер для специфического периферийного устройства.

Процесс, необходимый для интегрирования драйвера устройства очень похож на тот, что требуется для разработки пакета программ. В следующих секциях описан процесс для обоих направлений. Определённые пункты, которые не требуются для пакета программ, выделены в тексте.

Общее представление

Для издания драйвера устройства или пакета программ, вам необходимо предоставить следующее:

- Заголовочный файл, определяющий интерфейс пакета или драйвера
- Скрипт Tcl, задающий то, как добавляется пакет или драйвер в BSP

Заголовочный файл и Tcl скрипт описаны в следующих секциях.

Допущения и требования

В этой секции подразумевается, что вы уже разработали драйвер устройства или пакет программ для возможного интегрирования в BSP. Драйвер или пакет, который будет интегрирован в BSP, конечным пользователем, который имеет ограниченное представление о внутренней реализации драйвера или пакета. Для добавления вашего драйвера или пакета в BSP, конечный пользователь должен полагаться только на настройки драйвера или пакета, которые вы создаёте с помощью инструментов, описанных в этой секции.

Чтобы драйвер устройства или пакет программ работал с Nios II SBT, он должен соответствовать следующим критериям:

- Он должен иметь определённый Tcl скрипт. Скрипт Tcl для каждого драйвера или пакета программ предоставляется Nios II SBT с полным описанием драйвера или программы. Такое описание должно содержать следующую информацию:
 - Имя – Уникальное идентификационное имя для драйвера или пакета программ.
 - Исходные файлы – Место, имя и тип каждого исходного или заголовочного файлов на языке C++/ Си или ассемблера.
 - Ассоциированный класс устройства (только для драйвера устройства) – Название класса аппаратной периферии, поддерживаемого драйвером.
 - Информация о версии и совместимости – Версия драйвера или пакета, и (для драйверов) информация о том, какие версия ядер устройств он поддерживает.
 - Типы BSP – Поддерживаемые операционные системы.
 - Настройки – Видимые параметры, контролирующие сборку программы и конфигурацию во время прогона.
- Скрипт Tcl, размещённый в корневой директории драйвера или пакета программ.
- Имя файла скрипта Tcl заканчивается на **_sw.tcl**. Например: **custom_ip_block_sw.tcl**.
- Корневая директория драйвера или пакета программ может находиться в одном из следующих мест:
 - В любой директории, находящейся в переменной SOPC_BUILDER_PATH среды, или в любой директории, находящейся ниже на один уровень от указанной директории. Это рекомендовано, когда ваш драйвер или пакет программ инсталлирован в созданное вами место.
 - В директории, называемой **ip**, на один уровень ниже директории проекта Quartus II, содержащий ваш проект BSP. Это рекомендовано, когда ваш

драйвер или пакет программ используется только в специфическом проекте.

- Имена файлов и структура директории согласовываются с определёнными условными обозначениями, описанными в секции "Имена файлов и размещение" на стр. 7-20.
- Если ваш драйвер или пакет программ использует механизм авторизации HAL (`alt_sys_init()`), определённые макросы должны быть указаны в заголовочном файле. За подробной информацией о заголовочном файле обратитесь к секции "Заголовочные файлы и `alt_sys_init.c`" на стр. 7-16.

За подробной информацией об интегрировании драйверов устройств в HAL, обратитесь к [AN 459](#): "Руководство по разработке драйверов устройств под Nios II HAL". За подробной информацией о командах Tcl скрипта, которые вы можете использовать в драйвере, обратитесь к главе "[Справка по Nios II SBT](#)" в настольной книге программиста под Nios II.

Генератор Nios II BSP

В этой секции описываются процессы при которых Nios II BSP генератор добавляет драйверы устройств и пакеты программ в ваш BSP. Nios II BSP генератор – подкласс Nios II SBT – это комбинация из командных утилит и скриптов, которая позволяет вам создавать и управлять BSP и его настройками.

Общее представление о Nios II SBT находится в главе "Общее представление и начало работы в командной строке" в настольной книге программиста Nios II.

Обнаружение компонента

Когда вы запускаете утилиту генератора BSP, заполняется библиотека доступных драйверов и пакетов программ.

Генератор BSP обнаруживает пакеты программ и драйверы, проверяя список известного размещения, определённый в Altera Nios II EDS, программе Quartus II и инсталляторах библиотеки MegaCore® IP, а заодно и обнаруживает размещение, заданное в определённых переменных системного окружения.

Инструменты Nios II BSP идентифицируют драйверы и пакеты программ, отыскивая Tcl скрипты с файлами, заканчивающимися на `_sw.tcl`, в месте поиска.

Для эффективной работы на стадии прогона, генератор BSP просматривает только файлы драйвера, которые соответствуют критерию, описанному в этой секции.

После локализации каждого драйвера и пакета программ, Nios II SBT ищет подходящий драйвер для каждого аппаратного модуля системы SOPC Builder (контролируется процессором Nios II, для которого сгенерирован BSP), а заодно и пакет программ, который запрашивается создателем BSP.

Версии драйвера устройства

В случае с драйверами устройств, используется старшая версия драйвера для ассоциированной аппаратной периферии, она добавляется в BSP, если, конечно, не выбран другой с помощью команд менеджера драйвера устройства.

За дополнительной информацией обратитесь к главе "[Справка по Nios II SBT](#)" в настольной книге программиста под Nios II.

Добавление драйвера устройства и пакета программ

Генератор BSP добавляет пакет программ в BSP, если он специально запрашивается во время генерации BSP, с помощью команды *enable_sw_package*.

За дополнительной информацией обратитесь к секции "Tcl команды" в главе ["Справка по Nios II SBT"](#) в настольной книге программиста под Nios II.

Если не запрашивается специальный драйвер устройства, и не найден совместимый драйвер устройства для определённого аппаратного модуля, генератор BSP выдаёт информативное сообщение в выходе генерации либо в debug (отладка), либо в verbose (многословный). Такое поведение нормально для большинства типов устройств в системе SOPC Builder, таких как устройства памяти, не имеющих драйверов устройств. Если пакет программ или определённый драйвер запрашивается и не может быть обнаружен, генерируется ошибка, а генерация BSP или обновление настроек приостанавливается.

Создание скриптов Tcl позволяет вам добавлять специальные определения в файл **system.h**, разрешать автоматическую инициализацию драйвера в структуре **alt_sys_init.c** и разрешать Nios II SBT контролировать любые специальные параметры, которые должны существовать.

Имея Tcl файлы программных определений, SBT читает Tcl файл и заполняет соответственно makefiles и другие поддерживаемые файлы.

Когда Nios II SBT добавляет каждый драйвер или пакет программ в систему, он использует данные из Tcl скрипта, определяющие драйвер или пакет программ для контроля каждого файла, копируемого в BSP. Это правило распространяется на генерируемые BSP файлы, такие как BSP **Makefile**, **public.mk**, **system.h**, на настройки BSP и итоговые HTML файлы.

Когда вы создаёте новый программный проект, Nios II SBT генерирует содержимое **alt_sys_init.c**, чтобы создать определённое аппаратное содержимое системы SOPC Builder.

Имена файлов и размещение

Как было описано в секции "Генератор Nios II BSP" на стр. 7-19 инструменты создания Nios II находят драйвер устройства или пакет программ, локализуя Tcl скрипт с именем файла, оканчивающимся на **_sw.tcl**, и используют его в качестве исходника.

Каждая периферия в системе Nios II ассоциирована с определённой директорией компонентов SOPC Builder. Эта директория содержит файлы, определяющие программный интерфейс с периферией. Обратитесь к секции "Аппаратный доступ" на стр. 7-3.

Чтобы разрешить SBT находить драйвер устройства вашего компонента, разместите Tcl скрипт в директории с именем в директории **ip** вашего аппаратного проекта.

На рис. 7-1 показана иерархия файлов, совместимая с Nios II SBT. Эта иерархия файлов находится в директории *<Altera installation>/ip/altera/sopc_builder_ip*. В этом примере считается, что драйвер устройства поддерживает компонент *custom_component*.

Обнаружение исходного кода

Вы используете Tcl скрипты для задания положения исходных файлов драйвера. За подробной информацией обратитесь к секции "Генератор Nios II BSP" на стр. 7-19.