

ОРГАНИЗАЦИЯ ВЗАИМОДЕЙСТВИЯ МЕЖДУ HPS И FPGA НА БАЗЕ SoC ТЕХНОЛОГИЙ ПРИ СОЗДАНИИ ТЕЛЕКОММУНИКАЦИОННЫХ УСТРОЙСТВ

Кайдено В.М., Кайдено М.М., Крылач О.Ф., Роскошный Д.В.

Институт телекомунікаційних систем НТУУ «КПІ», Україна

E-mail: kkb610@ukrl.net

Organization of HPS and FPGA interaction based on SoC technology for telecommunication device development

This paper gives a description of HPS+FPGA technology based on SoC, its advantages for telecommunication device development, benefits of Linux OS usage in cope with this technology, and concepts for HPS and FPGA interaction organization.

В настоящее время разработка и применение систем-на-кристалле (SoC) является одним из наиболее перспективных направлений в разработке телекоммуникационных устройств и систем. В общем случае SoC являются дальнейшим развитием технологии FPGA и представляют собой интегрированные в одном кристалле аппаратное процессорное ядро (ядра), блок программируемой логической матрицы FPGA, а также аппаратно или программно реализованные модули управления периферийными устройствами.

Внутренняя шина данных с высокой пропускной способностью свыше 10 Гбит/с, соединяющая между собой процессорное ядро и логическую матрицу FPGA, позволяет управлять периферийными модулями и в некоторых случаях ускорить выполнение алгоритмов, повышая тем самым общую производительность системы. Такой уровень интеграции не только увеличивает производительность, но и снижает мощность потребления, размеры и стоимость конечного решения [1]. Упрощенная структурная схема SoC представлена на рис 1.

Принципиальная особенность SoC - это наличие программируемых блоков. Поэтому SoC – не просто интегральная схема, а комплекс, в состав которого входят как аппаратная часть (собственно FPGA и периферийные устройства), так и программная – встраиваемое программное обеспечение. Следовательно, при проектировании SoC необходимо выполнить операции по совместной верификации и отладке взаимодействия программной и аппаратной частей [2,3].

В телекоммуникационных приложениях, в частности, в системах беспроводного широкополосного доступа, включая решения Software Defined Radio (SDR), использование технологии HPS+FPGA дает возможность ускорить разработку конечных телекоммуникационных устройств, на выходе которой можно получить мощное устройство способное обеспечить большинство необходимых вычислений в режиме реального времени (благодаря возможности использования FPGA), которое тем не менее может легко

интегрироваться в любую систему в связи с возможностью использовать такие интерфейсы как Ethernet, I2C, SPI, CAN и другие, на высоком уровне операционной системы [4].

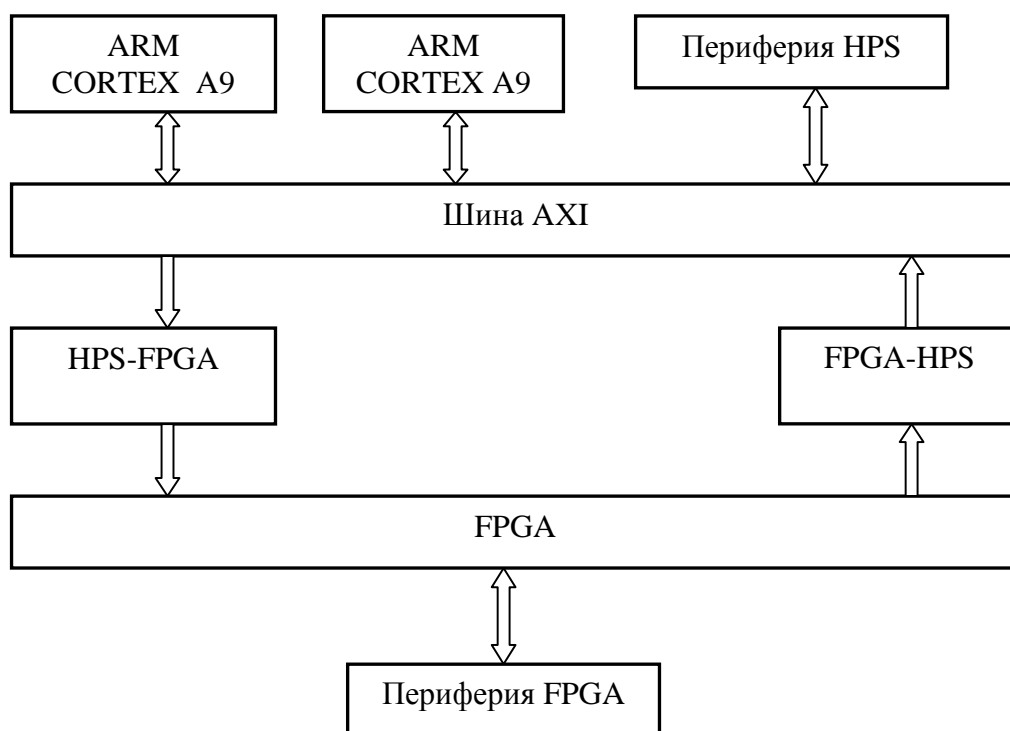


Рис. 1 Упрощенная структурная схема SoC.

Среди прочих достоинств этой технологии, применительно к SDR, особенно стоит отметить то, что она позволяет производить проектирование канала передачи данных на структурном уровне, при этом низкоуровневое проектирование будет необходимо только для реализации самых затратных с точки зрения вычислительного ресурса задач.

В разработке SoC систем существует 2 основных подхода:

- создание устройства без операционной системы (Bare Metal Application).
- использование операционной системы VxWorks/Linux

Первый подход дает возможность получения системы с максимальным уровнем производительности, но реализация его крайне сложна, так как предполагает полную разработку всех необходимых для SDR устройств, интерфейсов, и т.п. на низком уровне.

Второй подход облегчает задачу, так как на операционную систему ложится задача управления периферийными устройствами, реализации высокоуровневых протоколов обмена, таких как IP, TCP, UDP и даже протоколов доступа к устройству, например FTP, NTP, Telnet, SSH и пр.[4,5] Так же использование операционной системы позволяет создавать универсальные устройства, которые намного проще интегрировать в существующие системы, чем при использовании Bare Metal Application.

Предпочтительным является использование операционной системы Linux для SoC устройств, так как она имеет следующие преимущества [6]:

- широкая аппаратно-программная поддержка разработчика;
- огромная гибкость в приложениях готовой системы;
- простота в использовании периферийных устройств;
- наличие развитой поддержки средств разработки для встраиваемых систем;
- на уровне ядра поддержка многоядерных процессоров и процессоров ARM.

Существует три основных подхода к организации взаимодействия HPS и FPGA в ОС Linux:

1. Использование непосредственного доступа к регистрам памяти FPGA;
2. Использование пользовательского интерфейса ввода/вывода;
3. Создание полноценного драйвера устройства Linux на функциональном уровне.

1. Простейший способ обмена данных между FPGA и процессором в обход операционной системы, который тем не менее дает возможность использовать все остальные её возможности. Взаимодействие с FPGA происходит путем прямой записи/чтения по адресам памяти, которые ей присваивает операционная система. Не позволяет использовать дополнительные возможности операционной системы и самой FPGA, такие как DMA.

2. Пользовательский интерфейс ввода/вывода операционной системы Linux позволяет создавать упрощенную версию драйверов для обмена между FPGA и центральным процессором. При таком подходе имеется возможность работать по прерываниям в операционной системе, но отсутствует возможность управления кешированием, что не дает использовать технологию прямого доступа к памяти(DMA).

3. Комплексный подход, который позволяет использовать все возможности операционной системы и FPGA при работе SoC с любой периферией, в том числе и АЦП и ЦАП в SDR. Так же позволяет один драйвер для однотипных устройств, что значительно упрощает процесс разработки готовых систем.

Использование SoC устройств совместно с SDR позволяет значительно ускорить процесс разработки беспроводных телекоммуникационных устройств и систем.

Литература

1. https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/br/br-industrial-brochure.pdf.
2. https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/wp/wp-01198-fpga-software-debug-soc.pdf.
3. https://www.altera.com/content/dam/altera-www/global/en_US/pdfs/literature/ug/ug_soc_eds.pdf
4. http://www.eejournal.com/files/8713/9837/5380/WP-14033_-_Altera_SoC_Design_Advantages_1.pdf
5. <http://www.microtoolsinc.com/Why%20Use%20Embedded%20Linux%20for%20Real%20Time%20Embedded%20Systems%20Rev%20A.pdf>
6. <http://www.denx.de/en/pub/Documents/Presentations/2013-06-25-customizable-arm.pdf>