

ПОДДЕРЖКА ПРОЦЕССОРА NEUROMATRIX В СРЕДЕ МОДЕЛЬНО-ОРИЕНТИРОВАННОГО ПРОЕКТИРОВАНИЯ MATLAB/SIMULINK С РАСШИРЕНИЕМ EMBEDDED CODER

Инж. Мушкаев С.В., инж. Андрианов А.В., инж. Песельник М.Г.

ЗАО НТЦ «Модуль» mushkaev@module.ru, ООО ЦИТМ Экспонента

Рассматривается задача интеграции процессорного ядра NeuroMatrix® в среду модельно-ориентированного проектирования MATLAB/Simulink. В частности, изучается задача поддержки гетерогенного ARM/DSP (NeuroMatrix®) процессора K1879XB1Я разработки ЗАО НТЦ «Модуль» в среде моделирования Simulink. Рассматриваются принципы взаимодействия Simulink с микрокомпьютером MB77.07 на базе процессора K1879XB1Я, а также механизм RPC управления и передачи данных внутри многопроцессорной СБИС. Приводятся примеры моделирования, преимущества и возможности такого подхода проектирования.

Введение

Сложность современных задач цифровой обработки сигналов требует привлечения все более сложных систем визуального отображения и анализа информации как на этапе прототипирования, так и на этапе разработки и отладки ПО. С другой стороны, многообразие и сложность существующих DSP платформ создают определенные трудности при проектировании, выборе или переходе на новую архитектуру так как требуют знания ее архитектурных особенностей, навыков программирования, эксплуатации, а в идеале и владения языком ассемблера. Удобным средством для решения данных проблем на всех этапах цикла разработки является широко известная среда модельно-ориентированного проектирования MATLAB/Simulink [1].

В успехе технической разработки DSP аппаратно-программного комплекса можно выделить следующие составляющие: качество алгоритма, выбор аппаратной платформы, поддержка системным ПО, поддержка процессора прикладным библиотечным ПО, средства разработки, отладки, визуализации и верификации, владение навыками низкоуровневого программирования и методами оптимизации ПО.

Каждый из этих пунктов является критичным для эффективной и быстрой реализации алгоритма. Более того, процесс перехода от модели алгоритма к программе не является односторонним. В ходе адаптации к конкретной архитектуре алгоритм может претерпевать серьезные изменения. Как правило, процесс является итеративным и трудоемким, и в этой связи крайней важной является возможность быстрого прототипирования, визуализации результатов, ввода тестовых сигналов и верификации. Следует отметить, что наибольшую сложностью процесс представляет в самом начале при недостаточных навыках программирования или даже эксплуатации того или иного вычислительного устройства. Для решения данных проблем Simulink является отличным инструментом, так как предоставляет большие возможности для модельно-ориентированного проектирования алгоритма, быстрого прототипирования с автоматической генерацией С-кода, запуска на целевом устройстве либо его эмуляторе. В данной статье рассматривается Simulink как инструмент для быстрого прототипирования, программирования и запуска алгоритма на процессорах с архитектурой NeuroMatrix® разработки ЗАО НТЦ «Модуль».

Модельно-ориентированное проектирование в MATLAB/Simulink

Simulink - лидирующая графическая среда имитационного моделирования, позволяющая при помощи блок-диаграмм в виде направленных графов строить динамические системы. Simulink предназначен для симуляции и разработки динамических и встраиваемых систем. В среде можно моделировать линейные, нелинейные, дискретные, непрерывные, гибридные системы. Simulink является основой для модельно-ориентированного проектирования, где с помощью расширения - генератора кода (Embedded Coder) из созданной модели можно сформировать исходный код на языке С для заданного целевого устройства, скомпилировать и запустить в автоматическом режиме. В общем случае процесс верификации на целевом устройстве показан на рисунке 1.

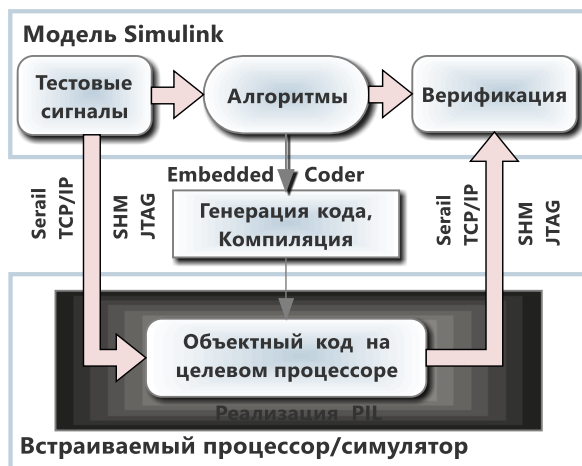


Рисунок 1 Запуск модели на целевом процессоре в режиме PIL

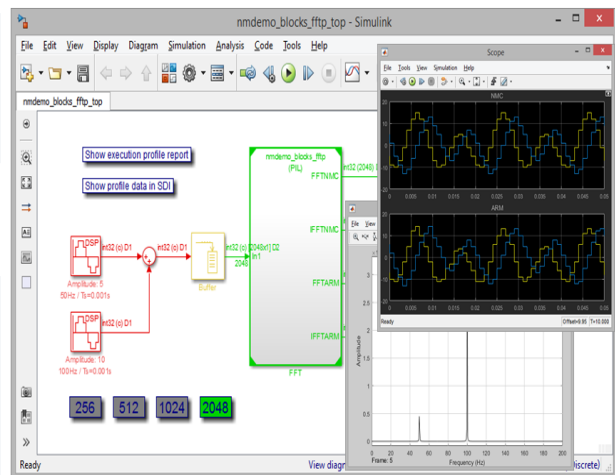


Рисунок 2 Внешний вид модели и верификации в Simulink

Запуск модели осуществляется следующим образом: по заданной модели Simulink с помощью расширения для Simulink - Embedded Coder – производит генерацию портируемого, компактного и читаемого ANSI C кода для встраиваемых систем. При этом в зависимости от настроек каждый блок модели может породить как чистый C-код, так и замениться на вызов соответствующей C-функции, реализованной и оптимизированной под данное устройство. Такой механизм называется «подстановка кода». Сгенерированный ANSI C код кросс-компилируется и загружаются на целевое устройство. Через доступные средства обмена данных входные тестовые сигналы из Simulink поступают на исполнение в вычислительный модуль, после чего результаты возвращаются в Simulink для анализа, визуализации и верификации. Такой обмен данными между Simulink и целевым устройством происходит в непрерывном цикле и называется «Процессор-в-контуре» (PIL, Processor-in-the-Loop). Как показано на рисунке 1, обсчет модели может проводиться как по нижнему контуру через целевой процессор, так и непосредственно в Simulink. Полученные напрямую результаты в Simulink в качестве эталонных, могут быть сверены с результатами от нижнего контура и таким образом осуществляется верификация модели.

Другими словами, происходит верификация того, что сгенерированный код, выполняющийся на целевом процессоре, полностью соответствует оригинальной модели Simulink. На рисунке 2 показан пример моделирования алгоритма и результата его верификации в среде Simulink. В качестве дополнительной возможности в режиме PIL может осуществляться профилирование (измерение времени выполнения) кода, чтобы убедиться, что реализация соответствует требованиям по скорости. Кроме того, имеется возможность запускать сгенерированный код не только на устройстве, но и на хост-компьютере, такой режим называется «Программа-в-контуре» (SIL, Software-in-the-Loop). Этот режим удобно использовать на ранних этапах верификации, или когда целевой вычислитель недоступен.

Поддержка NeuroMatrix в Simulink

Simulink поддерживает большое число встраиваемых аппаратных платформ. Фирмой ЗАО «НТЦ» Модуль был разработан микрокомпьютер MB77.07 (рис. 4) [2] на базе системы на кристалле K1879XB1Я [3]. Процессор представляет собой СБИС, состоящий из управляющего ядра ARM-1176 и высокопроизводительного векторного сопроцессора NeuroMatrix [4].

Задачей проекта было интегрировать вычислительный модуль MB77.07 в систему Simulink, что позволило бы разрабатывать алгоритмы с использованием модельно-ориентированного проектирования и тут же их исполнять на вычислительном модуле. С учетом архитектуры СБИС K1879XB1Я (рис. 3) общая схема запуска модели в Simulink на микрокомпьютере MB77.07, представленная на рис. 1 трансформируется к виду, показанному на рисунке 5.

Так как система на кристалле K1879XB1Я является гетерогенной системой, стоящей из управляющего процессора ARM-1176 и векторно-матричного сопроцессора с ядром NMC3 [4], то исполнение модели программного распределяется на оба процессора.

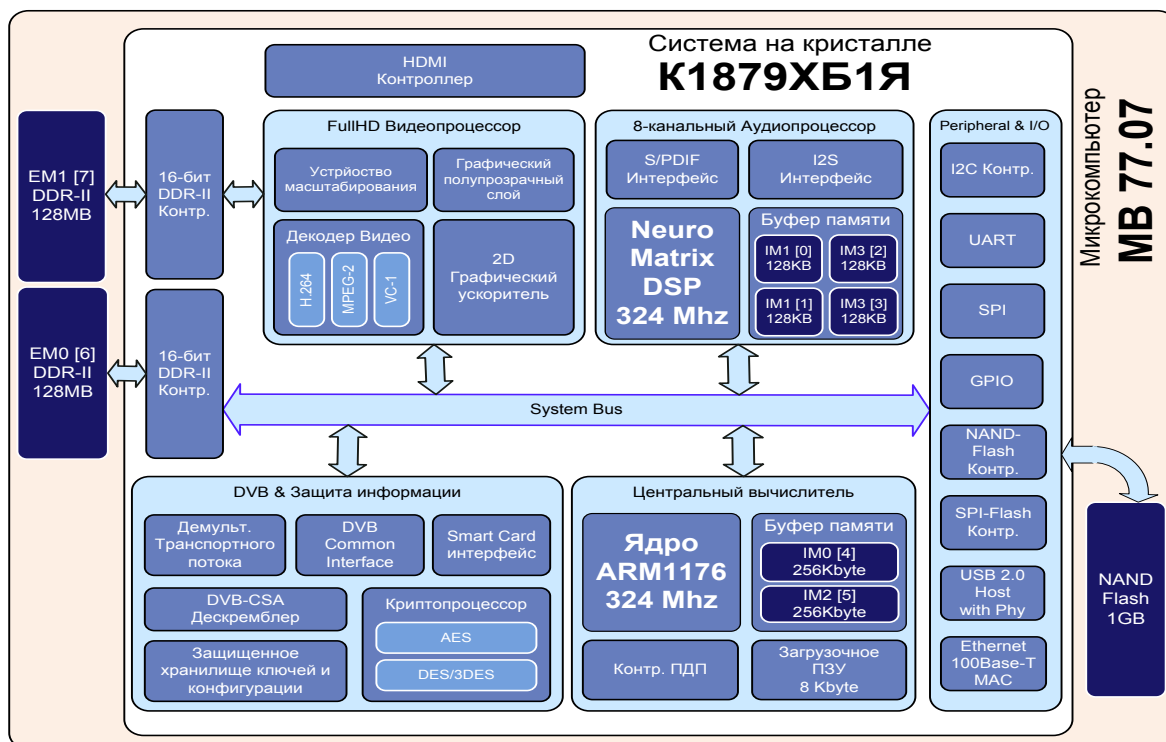


Рисунок 3 Структура СБИС в составе микрокомпьютера MB77.07

Генерация основного исполняемого кода из модели осуществляется под ARM процессор. Запуск программы и обмен данными между Simulink и микрокомпьютером так же осуществляется через ARM-процессор под управлением OS Linux. При этом для тех блоков модели, которые имеют программную реализацию под NeuroMatrix, происходит их запуск на сопроцессоре через механизм удаленного вызова процедур RPC (Remote Procedure Call).

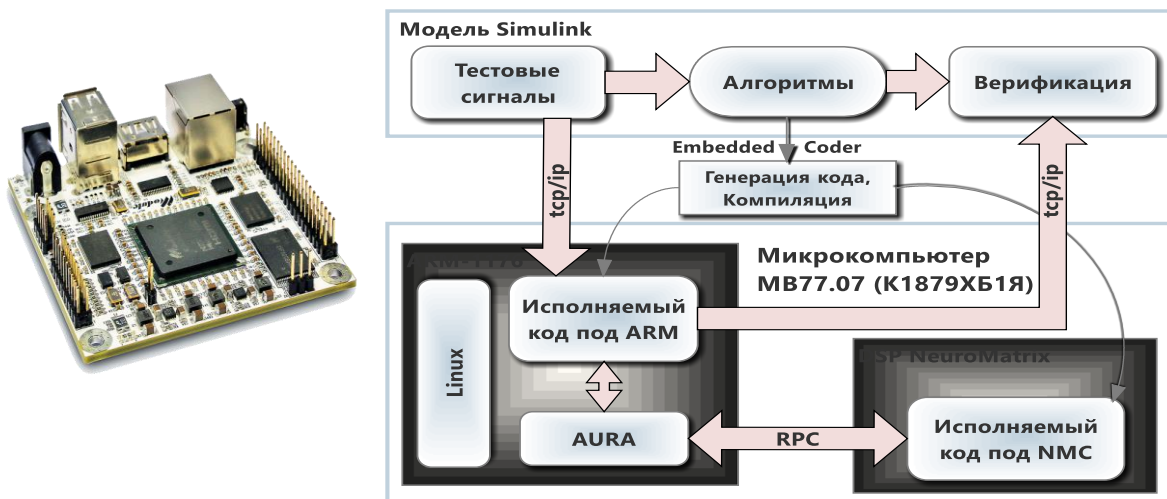


Рисунок 4 Микрокомпьютер MB77.07

Рисунок 5 Запуск модели в режиме PIL на СБИС K1879XB1A

Механизм удаленного вызова процедур

Для реализации механизма удаленного вызова процедур на NeuroMatrix со стороны ARM была адаптирована универсальная библиотека AURA (Another Universal RPC, Actually) [6].

Aura – универсальная библиотека высокопроизводительного RPC, ориентированная на обмен данными между хост-системой под управлением OS Linux и удаленным узлом, которым могут являться микроконтроллеры и DSP процессоры без операционной системы или под управлением RTOS.

Ряд ее особенностей делает ее особенно привлекательной при использовании совместно с DSP Neuromatrix:

- Наличие асинхронного API. (Результат выполнения удаленной процедуры можно получить в функции обратного вызова, callback)
- Наличие механизма событий, позволяющего организовать симметричный обмен данными при использовании асинхронного API
- Аллокация буфера для передаваемых параметров и возвращаемых значений производится в модуле транспорта. Это позволяет записывать передаваемые параметры сразу в буфер, выделенный, например, через ION Memory Manager
- Наличие встроенного пула буферов, позволяющего существенно снизить количество вызовов аллокатора памяти, которые могут занимать продолжительное время
- Интеграция с библиотекой асинхронного ввода-вывода libevent
- Привязки к языку lua, позволяющие существенно ускорить разработку хост-части на стадии прототипирования приложения
- Возможность передачи буферов, выделенных через ION в виде аргументов вызываемой удаленной процедуры

В отличие от традиционных библиотек удаленного вызова процедур библиотека AURA устраняет такие характерные недостатки как: асимметричность - когда только одна из взаимодействующих сторон является инициатором и синхронность - когда выполнение вызываемой процедуры приостанавливается до момента возврата из вызываемой процедуры.

Основным преимуществом данной библиотеки является возможность использовать специальный аллокатор памяти, специфичный для системы с которой происходит взаимодействие.

При построении взаимодействия между процессорным ядром общего назначения, работающим под управлением OS Linux и вычислительными DSP ядрами необходимо учитывать ряд особенностей аппаратуры, оказывающих серьезное влияние на итоговую производительность. В частности, основной проблемой при выделении памяти в программе стандартным аллокатором (malloc) является ее фрагментация в физическом пространстве и, вследствие этого, невозможность передачи DSP процессору как непрерывного сегмента. Кроме того, страницы памяти буфера, выделенного в пространстве пользователя, могут быть выгружены из памяти ядром операционной системы (если используется файл/раздел подкачки). В системах с распределенной системой памяти время доступа также может сильно отличаться в зависимости от типа памяти или переходе от одной страницы к другой.

В случае с Neuromatrix DSP использование ION Memory Manager и библиотеки RPC aura позволило полностью избежать копирования данных при передаче буфера от ARM к Neuromatrix и обратно (zero-copy).

Интеграция AURA в пакет поддержки NeuroMatrix в Simulink

В общем случае из модели Simulink генерируются ANSI-C код, который целиком выполняется на ARM процессоре. Для тех блоков в модели, у которых существует специальная целевая реализация (сумматор и вычитатель на рис. 6) происходит подстановка функций (ADD,SUB). Каждая такая подстановка представляет собой RPC (remote procedure call) удаленный вызов процедуры, через который осуществляется передача дескриптора функции и ее аргументов на NMC. На стороне NMC работает монитор, который отслеживает RPC события со стороны ARM и вызывает соответствующие grc_xxx функции. Данные функции являются интерфейсной оберткой для вызова конечных целевых функций (add, sub, mul,fft ...). На рис.6 показана цепочка вызовов в RPC – механизме.

Конечные целевые функции входят в состав самостоятельной векторной библиотека NMPP (Neuro Matrix Performance Primitives) [6] для семейства процессоров NeuroMatrix. Библиотека включает оптимизированные векторно-матричные операции, функции обработки сигналов и изображений. Для наиболее частых операций (базовые логические, арифметические операции, векторно-матричные умножения, поиски максимума, минимума, преобразования фурье и др.)

на основе AURA была создана соответствующая двухкомпонентная (ARM+NMC) библиотека, с помощью которой осуществлен механизм RPC вызова функций на NeuroMatrix.

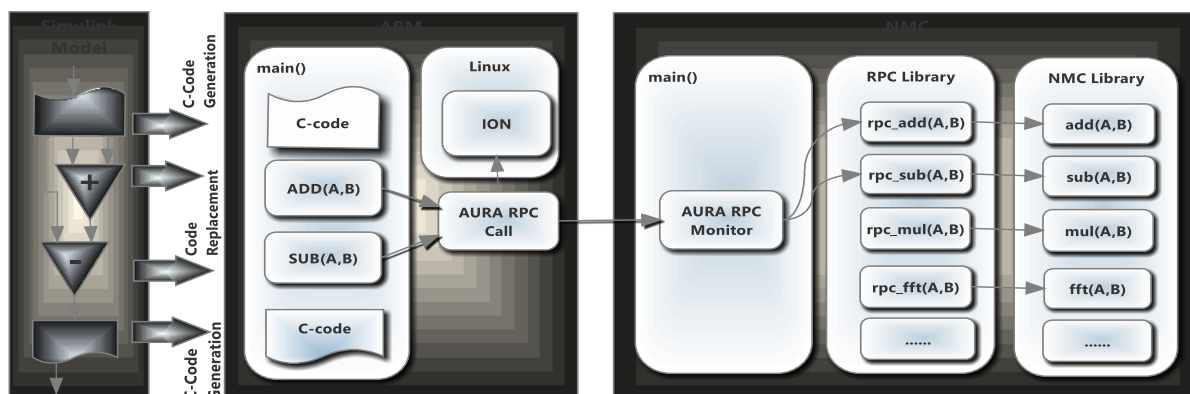


Рисунок 6 Реализация механизма удаленного вызова процедур в сгенерированном С-коде модели Simulink

Заключение

Реализованная поддержка микропроцессоров NeuroMatrix в Simulink является достаточно унифицированной и может легко расширяться на другие процессоры НТЦ Модуль.

Дополнительно, для пользователей, которые используют только MATLAB (без Simulink), существует возможность генерировать С-код напрямую из функций MATLAB при помощи MATLAB Coder. При этом сохраняются возможности подстановки специфических функций NeuroMatrix и верификации в режиме PIL так же, как для пользователей Simulink.

Набор функциональных блоков Simulink достаточно обширен, в то время как состав поддерживаемых в NeuroMatrix функций относительно ограничен. Несмотря на то, что библиотека NMPP непрерывно пополняется, покрыть все множество функций MATLAB/Simulink представляется довольно трудоемкой задачей. В виду этого, в целях поддержки пользователей допускается возможность реализации требуемого функционала по запросу.

Литература

1. URL: <http://matlab.ru/products/simulink>
2. «Универсальный вычислительный модуль на основе СБИС К1879ХБ1Я отечественной разработки», Словик А. Ю., Залетов В. Ю., Шевченко П. А., Лысенко Т. Л., Сборник докладов Международной конференции "Микроэлектроника 2015"
URL: http://www.module.ru/interesting/page-1/dokladi_mikroelektronika_215/
3. «Отечественная СБИС декодера цифрового телевизионного сигнала К1879ХБ1Я (PDF)», Шевченко П.А., Цифровая обработка сигналов и ее применение - DSPA'2011, Москва 2011
URL: http://www.module.ru/upload/images/1354523271art_dspa2011_1.pdf
4. «СБИС на базе ядра NMC3 для программного приемника навигационных сигналов», Косоруков Д.Е., Эйсымонт А.Л., Осипов В.Г., Панфилов А.П., Черников В.М., Вискне П.Е., Шелухин А.М., Насонов И.И., Сборник докладов Международной конференции "Микроэлектроника 2015"
URL: http://www.module.ru/interesting/page-1/dokladi_mikroelektronika_215/
5. URL: <https://github.com/nekromant/aura>
6. URL: <https://github.com/RC-MODULE/nmpp>