

**АНАЛИЗ ЭФФЕКТИВНОСТИ ПРОГРАММНОЙ РЕАЛИЗАЦИИ ТИПОВЫХ
ОПЕРАЦИЙ ОБРАБОТКИ РАДИОЛОКАЦИОННЫХ СИГНАЛОВ НА
ЦИФРОВЫХ СИГНАЛЬНЫХ ПРОЦЕССОРАХ**

Андреев Н.А., Витязев С.В., Воронков Д.В.

Представлен анализ эффективности решения задач обработки радиолокационных сигналов современными цифровыми сигнальными процессорами и модулями на их основе. В качестве примера рассматривается оптимальный прием квазинепрерывного сигнала. Анализируются архитектуры процессоров TMS320C6701, ADSP-TS101, ADSP-TS201 и 1892BM2T(MC-24) серии "Мультикор". Рассматриваемое программное обеспечение разрабатывается для модулей 1C000 и 3C001 производства Рязанского государственного приборного завода.

**THE ANALYSIS OF DIGITAL SIGNAL PROCESSORS ABILITIES TO REALIZE
SIGNAL PROCESSING IN RADAR APPLICATIONS**

N.A. Andreev., S.V. Vityazev, D.V. Voronkov

The paper presents the analysis of modern digital signal processors appliances in radar applications. The multiprocessor system solutions are also considered. One typical radar algorithm is selected to demonstrate the performance of hardware platforms. TMS320C6701, ADSP-TS101, ADSP-TS201, Multicore MC-24 DSPs and 1S000 and 3S001 multiprocessor modules are investigated.

Модули цифровой обработки сигналов (МОС) 1C000 и 3C001 производства Рязанского государственного приборного завода предназначены для решения задач цифровой обработки сигналов в составе бортовых вычислительных комплексов. Модули строятся на базе двух цифровых сигнальных процессоров (ЦСП) TMS320C6701 компании Texas Instruments (МОС 1C000) и на базе четырех ЦСП ADSP-TS101S TigerSHARC компании Analog Devices (МОС 3C001). Описание состава модуля 1C000 и его технических характеристик можно найти в [1]. Описание модуля 3C001 предполагается к публикации в том же издании. Данный доклад посвящен демонстрации эффективности модулей для решения типовых задач цифровой обработки радиолокационного (РЛ) сигнала. Дополнительно в докладе рассматриваются перспективные решения – построение модулей обработки сигналов на базе ЦСП ADSP-TS201 TigerSHARC компании Analog Devices и на базе сигнального контроллера 1892BM2T(MC-24) «Мультикор» российского научно-производственного центра «ЭЛВИС».

К типовой задаче, рассматриваемой в данной работе, отнесем оптимальный прием квазинепрерывного сигнала при работе РЛ-станции в режиме высокой частоты повторения (ВЧП). Задачу сформулируем следующим образом. Зондирующий сигнал представляется пачкой импульсов длительностью $T_{\text{и}} = 5$ мкс переменной частоты повторения $F_{\text{п}} = [56..200]$ кГц. Длительность пачки составляет $T = 8.74$ мс.

Модулированный сигнал излучается в пространство, отражается от цели (целей) и поступает на устройство цифровой обработки сигнала (исследуемый модуль) в виде цифровых отсчетов демодулированного сигнала на частоте дискретизации 7 МГц. Вся пачка импульсов представляется, таким образом, 61180 отсчетами комплексного сигнала. Такой входной сигнал разбивается на восемь стробов (без перекрытия), обуславливая работу по восьми каналам дальности. Дальнейшая обработка ведется процессором в пределах одного строба по очереди для каждого из них. При наличии N процессоров на модуле одновременно обрабатывается N стробов, и суммарное время обработки сокращается в N раз. Задача обработки заключается в выделении из принятого сигнала информации о наличии целей, их скоростях и дальностях. Алгоритм решения задачи будет включать следующие этапы.

1. Согласование с длительностью строба – суммирование в каждом периоде повторения в пределах одного строба.
2. Дополнение массива обработки нулями до размера, пригодного для выполнения БПФ.
3. Коррекция неортогональности – умножение в синфазном и квадратурном каналах на корректирующие коэффициенты.
4. Весовая обработка.
5. Быстрое преобразование Фурье (БПФ).
6. Переход к действительной форме представления сигнала.
7. Обнаружение и выдача информации.

Целью настоящего исследования явилась оценка эффективности реализации представленного алгоритма на указанных аппаратных платформах. Для всех процессоров в качестве базового языка программирования был выбран язык Си с последующей доработкой фрагментов программ на ассемблере при необходимости. Оценка эффективности проводилась для случая частоты повторения импульсов зондирующего сигнала 100 КГц, обуславливающей длительность сигнала в 874 периодов дискретизации (ограничиваемую 856 периодами) и размерность преобразования Фурье - 1024 точки.

Процессор TMS320C6701 и модуль МОС 1С000

Процессор TMS320C6701 компании Texas Instruments является базовым процессором семейства C6000, работающим в режиме с плавающей точкой, и, будучи представителем передовой и наиболее развиваемой платформы компании, все же не может считаться новой и наиболее высокопроизводительной моделью в своем семействе. Проведенная работа показала, что к недостаткам устройства можно отнести относительно невысокую рабочую частоту (120 МГц), недостаточный объем внутрикристалльной памяти данных (64 Кбайта), малую пропускную способность каналов обмена ядра с памятью, а также многотактное выполнение большинства ключевых команд, сильно затрудняющее процесс программирования и оптимизации. Последний недостаток компенсируется основным достоинством процессора – развитыми средствами проектирования – эффективным компилятором языка Си и оптимизатором линейного ассемблера. Отметим, что выводы о достоинствах и недостатках моделей делаются на основе сравнения их друг с другом. Так процессор TMS320C6701 сравнивался с процессором ADSP-TS101.

Наиболее трудоемкими оказываются процедуры ввода данных и согласования с длительностью строба. Это связано с высокой частотой дискретизации (7 МГц), характерной для сигнала на этих этапах, в то время как последующая обработка ведется на меньшей частоте дискретизации, равной частоте повторения импульсов (в нашем случае 100 КГц). Ввод данных заключается в загрузке входного сигнала в виде массива

из 7647 комплексных элементов из внешней памяти во внутреннюю. Поскольку сигнал такой длины не может уместиться во внутренней памяти процессора, загрузку приходится осуществлять поблочно и совмещать с процедурой согласования с длительностью строба. Данный фрагмент кода реализован на ассемблере и представлен циклом загрузки блока данных во внутреннюю память по каналу прямого доступа (ПДП) и одновременного суммирования элементов предыдущего загруженного блока. При разработке кода программы использовались средства линейного ассемблера. В полученной структуре накопление реализуется весьма эффективно. Четыре строчки цикла выполняют обработку четырех комплексных чисел, и само накопление выполняется, ориентировочно, за 18000 командных циклов процессора с учетом затрат, связанных с предложенной структурой процедуры. Однако решающую роль в определении временных затрат на процедуру и на весь алгоритм в целом играют возможности устройства по вводу данных из внешней памяти. Процессор TMS320C6701 имеет 32-разрядный интерфейс внешней памяти, способный работать на частоте ядра процессора. На модуле 1C000, для которого велась разработка, обмен с внешней памятью осуществляется на частоте, равной половине частоты ядра процессора. Таким образом, загрузка 7647 комплексных чисел требует 30588, а с учетом описанной структуры, включающей два цикла загрузки-суммирования, запуска и ожидания ПДП, вся процедура длится 52000 командных циклов процессора.

Дополнение нулями сводится к циклически повторяемой процедуре записи в память и полностью определяется возможностями канала записи. На процессоре TMS320C6701 пропускная способность этого канала составляет два 32-разрядных числа (действительных) за цикл (в отличие от канала загрузки, где возможна передача четырех таких чисел). Поэтому вся процедура для рассматриваемого случая длится 195 командных циклов (запись 2×177 нулей плюс вспомогательные команды).

Коррекция неортогональности усложняется наличием переменных входных параметров (также как и все другие процедуры, зависящие от выбора частоты повторения). Процедура коррекции эффективно преобразуется в ассемблерный код оптимизирующим компилятором из Си-программы. При этом автоматически применяется программная конвейеризация (software pipelining) при оптимизации основного цикла. Этот базовый цикл включает две VLIW-команды обработки одного комплексного числа. Необходимость двух команд в теле цикла обусловлена недостаточными возможностями обмена ядра с памятью. Время выполнения процедуры 1757 циклов (две длины сигнала плюс вспомогательные команды). При оптимизации на языке Си для получения эффективного ассемблерного кода строки программы несколько раз изменялись по структуре и порядку следования (без изменения выполняемого алгоритма), так чтобы подобрать вариант, правильно интерпретируемый компилятором.

Для реализации весовой обработки предварительная загрузка коэффициентов функции окна осуществляется по каналу ПДП. При этом из-за переменной длины сигнала осуществляется выбор из внешней памяти весовой функции, подходящей в каждом конкретном случае. По этой же причине существует необходимость домножения коэффициентов функции окна на масштабирующий коэффициент. Основной цикл обработки включает одну команду взвешивания одного числа (действительной или мнимой составляющей комплексного сигнала). Это определяет время выполнения процедуры 3547 командных циклов (удвоенное число коэффициентов окна плюс вспомогательные команды, в том числе цикл масштабирования). Программа реализована на Си. Для оптимизации использованы директива компилятора `MUST_ITERATE` и ключевое слово описания указателя на массив `restrict` [2], [3].

Для реализации БПФ и перестановки выходных отсчетов в прямой порядок следования были использованы библиотечные функции `DSPF_sp_cfft2_dit(x, w, N)` и `DSPF_sp_bitrev_cplx(x, ind, N)`, выполняемые в сумме за 23120 командных циклов процессора.

Переход к действительному сигналу осуществляется путем вычисления суммы квадратов действительной и мнимой частей каждого комплексного отсчета. Эта процедура свелась к циклическому повторению двух команд обработки двух комплексных чисел. Поэтому время выполнения процедуры составило 1058 командных циклов (размер сигнала после БПФ плюс вспомогательные команды).

Обнаружение и выдача информации были разделены на два этапа. Первый – сравнение всех элементов массива с фиксированным порогом и «запоминание» тех элементов массива обработки, в которых были обнаружены цели, с подсчетом их количества. Второй – подготовка массива информации о десяти целях, движущихся с наибольшей скоростью, включая расчет амплитуды сигнала и другие параметры. Такое разбиение упростило структуру циклов и позволило реализовать их более эффективно. Первый и основной этап сводится к циклу из трех команд обработки двух отсчетов уже действительного сигнала. Второй этап представляется циклом сложной и многокомандной структуры, но включает всего десять итераций. Полученное суммарное время выполнения процедуры составляет 1704 командных циклов процессора. При оптимизации программы на языке Си, кроме использованных в предыдущем случае приемов, было применено указание компилятору возможности загрузок удвоенной длины с помощью описания указателя `double` и функций работы с двойными словами `_itof()`, `_hi()`, `_lo()` [2], [3].

При реализации на модуле оба процессора работают фактически независимо – один по всем четным, другой – по всем нечетным стробам. Независимость процессоров оказывается возможной, во многом благодаря наличию собственной внешней памяти данных у каждого из них, куда входное устройство контроллера связи записывает весь поступивший сигнал. Выходной, единый для обоих процессоров, массив информации размещается в общей для них двухпортовой памяти обмена [1]. Совместная работа двух устройств организуется путем синхронизации их работы за счет выставления флагов начала и окончания выполнения задачи.

Итоги оценки времени выполнения разработанного программного обеспечения на процессоре TMS320C6701 сведены в таблицу 1.

Процессор ADSP-TS101 и модуль МОС 3С001

Процессор ADSP-TS101 компании Analog Devices является первым из семейства TigerSHARC – одного из самых передовых семейств компании. Однако в отличие от TMS320C6701 он существенно моложе по дате выпуска и может считаться новейшей моделью. Его достоинства по сравнению с TMS320C6701 – это повышенная рабочая частота – 300 МГц, большой объем внутрикристалльной памяти (6 Мбит), гибкость и эффективность каналов обмена ядра и внутренней памяти, возможность осуществлять операции сложения и вычитания пары операндов одновременно на одном вычислительном блоке, что существенно ускоряет вычисление БПФ. О недостатках процессора в ходе работы были сделаны следующие выводы. Большая глубина конвейера команд обуславливает динамические зависимости между предыдущими и последующими командами, что затрудняет процесс программирования и в некоторых случаях приводит к неэффективному использованию ресурсов и большим временным затратам. Программные средства поддержки разработки зачастую оказываются неэффективны, и Си-код приходится переписывать на ассемблере. Этот недостаток можно считать временным, поскольку инструментальные средства быстро

развиваются. Еще один предполагаемый недостаток может заключаться в неэффективности работы архитектуры в случаях, когда алгоритм не позволяет использовать SIMD-режим (одна команда – много данных). Этот недостаток носит предположительный характер, так как в ходе работы он выявлен не был.

Большой объем внутрикристалльной памяти позволяет сразу все отсчеты сигнала в одном стробе во внутреннюю память процессора с использованием двух буферов и осуществлять ввод данных по каналам ПДП на фоне вычислений. В результате процедура согласования с длительностью строба реализуется независимо от ввода данных, что по сравнению с реализацией на TMS320C6701 сильно упрощает ее структуру и сокращает время выполнения. Цикл накопления внутри блоков включает две команды обработки двух комплексных чисел. Наличие зависимостей между ресурсами вычислительных блоков вызывает простои командного конвейера, увеличивающие время выполнения цикла в два раза. Типичный для платформы TigerSHARC прием устранения этого недостатка, заключающийся в «разворачивании» цикла (Loop Unrolling) и обработке большего числа данных за то же время, не работает в данном случае. Причина этого – переменные параметры алгоритма. Число повторений как внутреннего, так и внешнего циклов обработки является заранее неизвестным. Поэтому корректная реализация цикла в случаях, когда число итераций оказывается достаточно мало, требует усложнения структуры внешнего цикла процедурами сравнения и условно выполняемыми фрагментами кода. Повышенная сложность структуры программы в этом случае приводит к затратам, большим, чем при реализации цикла без «разворачивания». Первоначально предложенный вариант реализации процедуры обуславливает время ее выполнения 19703 командных циклов процессора. При более детальном рассмотрении не исключается возможность дальнейшей оптимизации процедуры и сокращения времени ее выполнения, но для этого необходимы дополнительные исследования.

Дополнение массива обработки нулями до размерности БПФ осуществляется циклическим повторением записи сразу четырех нулей в одной команде. Время выполнения всей процедуры оказывается равным 112 циклов ($2 \times 177/4$ плюс вспомогательные команды). Процедура была написана на ассемблере.

Коррекция неортогональности сводится к циклу из четырех команд обработки четырех комплексных отсчетов и выполняется за 889 командных циклов процессора (длина сигнала плюс вспомогательные команды). Чтобы полностью использовать ресурсы архитектуры, процедуру пришлось переписать на ассемблере. При этом были использованы принципы «разворачивания» цикла и программной конвейеризации (Software Pipelining).

Для выполнения весовой обработки коэффициенты окна загружаются с использованием ПДП. Базовый цикл взвешивания включает шесть команд весовой обработки восьми комплексных чисел. Время выполнения составляет 1332 цикла (длина сигнала плюс вспомогательные команды, в том числе цикл масштабирования). Процедура реализована на ассемблере с использованием тех же принципов оптимизации, что и предыдущая.

БПФ осуществляет библиотечная функция `cfft(in, in, out, w, tw, N)`, которая включает и перестановку выходных отсчетов в прямой порядок следования. Время выполнения процедуры – 10441 цикл.

Си-программа перехода к действительному сигналу выполняется за 1057 циклов и базируется на циклическом повторении четырех команд обработки четырех комплексных чисел. Требуемый ассемблерный код удалось получить из Си-программы с использованием слова описания указателя `restrict`, директивы компилятора

loop_count(min, max, modulo) и объявления указателей на массивы типа complex_float. Принцип использования этих приемов оптимизации можно найти в [4], [5].

Обнаружение и подготовка выходной информации выполняются за 4458 циклов. Из них 4095 циклов затрачивается на поэлементное сравнение с порогом, сводящееся к повторению команд выборки, сравнения и принятия решения по двум действительным числам за восемь командных циклов в одной итерации. Процедура была реализована на ассемблере.

Реализация на модуле предполагает независимую обработку на четырех процессорах восьми стробов – по два на каждом процессоре поочередно. В отличие от модуля 1C000 модуль 3C001 построен по кластерной архитектуре. Все процессоры модуля по единственной кластерной шине, работающей на частоте 100 МГц, обмениваются данными с единственным устройством внешней памяти данных. Очевидно, что вероятность простоев процессоров из-за недостаточной скорости загрузки велика. Однако все зависит от конкретной задачи. В представляемой работе при выбранной длительности пачки зондирующих импульсов загрузка отсчетов входного сигнала в память всех четырех процессоров не вызывает затруднений.

Результаты оптимизации программного обеспечения для процессора ADSP-TS101 сведены в таблицу 1.

Процессор ADSP-TS201

Процессор ADSP-TS201 компании Analog Devices является одним из новейших устройств цифровой обработки сигналов. Перспектива его использования обусловлена повышенной рабочей частотой (600 МГц), расширенной внутрикристалльной памятью (24 Мбита), увеличенной пропускной способностью связанных портов (link-ports) и другими причинами [6].

Основной целью при разработке нового семейства процессоров (TS-20x) было размещение на кристалле памяти типа DRAM объемом до 24 Мбит, работающей в то же время на частоте 100 МГц, существенно меньшей частоты ядра. Эта цель стала причиной организации оригинальной структуры обмена ядра с внутренней памятью через систему промежуточных буферов, а также обусловила увеличение длины конвейера команд.

При переложении программного обеспечения обработки радиолокационного сигнала с процессора ADSP-TS101 на ADSP-TS201 потребовалась дополнительная процедура предварительной конфигурации системы внутренней памяти, иницирующей работу через кэш. В остальном программное обеспечение осталось практически без изменений.

Результаты оценки времени выполнения процедур рассматриваемого алгоритма на процессоре ADSP-TS201 сведены в таблицу 1. Небольшое увеличение по сравнению с процессором ADSP-TS101 числа командных циклов для всех процедур обусловлено большей длиной конвейера, а следовательно, большими потерями при переходах.

Процессор 1892BM2T(МС-24) «Мультикор»

Процессор 1892BM2T(МС-24) производства НПЦ «ЭЛВИС» является отечественным процессором, что, несомненно, можно считать основой перспективности его применения. Вместе с тем, данный процессор на равных способен конкурировать с передовыми моделями ведущих мировых производителей ЦСП, о которых мы писали выше, как по производительности, так и по гибкости архитектуры и другим параметрам [7].

Исследование эффективности процессора для решения рассматриваемой задачи позволило выявить следующие его особенности.

Наличие двух ядер – RISC (микроконтроллера) и DSP (сигнального процессора) – позволяет разграничить задачи собственно обработки и управления системой. Такое разделение задач делает эффективнее решение каждой из них, и повышает производительность всей системы в целом. В данном исследовании разработчики сконцентрировали свое внимание на использовании ресурсов только DSP-части. Пакет программ, тем не менее, строился на базе RISC-ядра, осуществляющего контроль за ходом выполнения обработки сигнала и организующего загрузки данных по DMA-каналам в память DSP.

Процессор имеет большой объем внутренней памяти (160 Кбайт) и характеризуется высокоэффективной SIMD-архитектурой и «настроенностью» на выполнение БПФ. Все команды процессора, за исключением команд перехода, выполняются за один такт, а программный конвейер состоит из трех ступеней. Это делает относительно простым процесс разработки программного обеспечения и позволяет получать «быстрый» и компактный код. Для программирования DSP, в настоящее время, применим только язык ассемблера. Однако в ближайшем будущем компания-производитель устройства планирует выпуск компиляторов языка C++, позволяющих разрабатывать программное обеспечение на языке высокого уровня для обоих процессорных ядер в едином синтаксисе языка и в рамках единого проекта. Недостатком устройства можно считать относительно невысокую тактовую частоту (100 МГц), меньшую частоты как процессора TigerSHARC, так и TMS320C6701.

Таблица 1

	Время выполнения процедур алгоритма							
	TMS320C6701		ADSP-TS101		ADSP-TS201		1892BM2T MC-24	
	циклы	мкс	циклы	мкс	циклы	мкс	циклы	мкс
Согласование с длительностью строба	52000	433,0	19703	65,7	20517	34,2	9416	94,2
Дополнение нулями до размера БПФ	195	1,6	112	0,4	127	0,2	170	1,7
Коррекция неортогональности	1757	14,6	889	3,0	903	1,5	1712	17,1
Весовая обработка	3547	29,6	1332	4,4	1376	2,29		
БПФ	23120	192,7	10441	34,8	10682	17,8	10300	103,0
Переход к действительному сигналу	1058	8,8	1057	3,5	1073	1,8		
Обнаружение	1704	14,2	4458	14,9	5489	9,1	1536	15,4
Всего:	83381	694,9	37992	126,6	40167	66,9	23134	231,3

Также как и на процессорах TigerSHARC достаточная внутренняя память процессора MC-24 позволяет разделить процедуры ввода данных и согласования с длительностью строка, что определяет выигрыш процессора по сравнению с устройством TMS320C6701. В то же время отсутствие динамических ограничений на совместное использование ресурсов ядра позволяет проще и эффективнее, чем на устройствах ADSP-TS101 и ADSP-TS201, реализовать согласование с длительностью строка.

При выполнении остальных процедур алгоритма процессор MC-24 также продолжает выигрывать по числу тактов у процессора TMS320C6701 за счет высокопроизводительной SIMD-архитектуры и одноктактных команд и у процессоров TigerSHARC за счет отсутствия динамических ограничений и изначальной разработки кодов программ на языке низкого уровня, позволяющей более эффективно использовать ресурсы кристалла. При измерении времени выполнения алгоритма в секундах устройство MC-24 продолжает уверенно опережать процессор TMS320C6701 и уступает процессорам ADSP-TS101 и ADSP-TS201 из-за меньшей рабочей частоты.

Данные, характеризующие время выполнения процедур рассматриваемого алгоритма на процессоре 1892BM2T - MC-24 Мультикор, отражены в таблице 1.

Литература :

1. Андреев Н.А., Рыбаков В.Ю., Марочкин М.В. Модуль цифровой обработки радиолокационных сигналов МОС 1С000. – Цифровая обработка сигналов, 2004, № 3, с. 47 – 50.
2. TMS320C6000 Optimizing Compiler User's Guide, SPRU187I, Texas Instruments, 2001.
3. TMS320C6000 CPU and Instruction Set Reference Guide, SPRU189F, Texas Instruments, 2000.
4. VisualDSP++ 3.5 C/C++ Compiler and Library Manual for TigerSHARC® Processors, Part Number 82-000336-03, Analog Devices, 2004.
5. ADSP-TS101 TigerSHARC® Processor Hardware Reference, Part Number 82-001996-01, Analog Devices, 2003.
6. ADSP-TS201 TigerSHARC® Processor Hardware Reference, Part Number 82-000815-01, Analog Devices, 2003.
7. Микросхема интегральная 1892BM2T. Руководство пользователя, ГУП НПП «ЭЛВИС», 2004.

Сведения об авторах :

Витязев Сергей Владимирович,
год рождения 1979,
окончил Рязанскую государственную
радиотехническую академию в 2001 году, где и
работает по настоящий момент. В 2004 году
поступил в очную аспирантуру. Направление
исследований: применение цифровых сигнальных
процессоров в системах связи, радиолокационных
системах, других приложениях.
e-mail: tor@rgta.ryazan.ru

Воронков Дмитрий Вадимович,
год рождения 1982,
окончил Рязанскую государственную
радиотехническую академию в 2004 году, где и
работает в настоящий момент.
Направление исследований: синтез адаптивных
фазированных решеток и применение цифровых
сигнальных процессоров для
телекоммуникационных и радиолокационных
систем