

## Метод «ручной оптимизации» кода программного обеспечения блока цифровой обработки на основе процессора ADSP2185M

Т.Н. Романова<sup>1</sup>, А.В. Сидорин<sup>2</sup>

<sup>1</sup> МГТУ им. Н.Э. Баумана, Москва, 105005, Россия

<sup>2</sup> ФГУП «НПО «Орион», Москва, 111123, Россия

*Рассмотрен выбор методов и приемов разработки программного обеспечения для процессора блока цифровой обработки камеры инфракрасного диапазона, а также программного обеспечения для ПЭВМ, производящего получение видеосигнала с блока цифровой обработки, его цифровую обработку и отображение. Описаны технические решения, которые позволили провести «ручную оптимизацию» кода для удовлетворения всех требований технического задания.*

**E-mail:** [rtn@bmstu.ru](mailto:rtn@bmstu.ru)

**Ключевые слова:** цифровая обработка сигналов, инфракрасная камера, матричное фотоприемное устройство, оптический тракт, сигнальный процессор ADSP2185, микроконтроллер AVR, ПЛИС Xilinx XC2S300E.

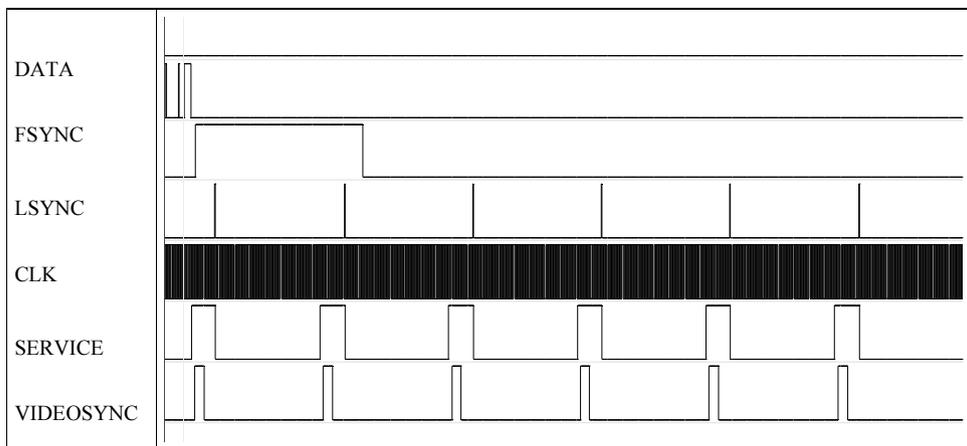
В настоящее время в России активно ведется разработка камер инфракрасного диапазона спектра. Современная камера строится на основе матричного фотоприемного устройства (МФПУ). Излучение, проходя через оптический тракт (ОТ), попадает на элементы МФПУ, аналого-цифровой преобразователь (АЦП) которого производит оцифровку сигнала. Для получения сигнала АЦП МФПУ оснащают блоком цифровой обработки (БЦО). На рис. 1 представлена структурная схема прибора на основе ФПУ из InGaAs. Первичную цифровую обработку сигнала может осуществлять БЦО, который также служит интерфейсом для подключения к другим устройствам (например, к ПЭВМ или к блоку индикации (БИ)). Как правило, БЦО представляет собой плату или набор плат с несколькими процессорами и контроллерами, осуществляющими обработку сигнала. Алгоритмы цифровой обработки при этом реализуются не аппаратной логикой, а с помощью микропрограмм, под управлением которых работают элементы БЦО [1, 2]. Далее рассмотрим, от чего зависит выбор методов и приемов разработки программного обеспечения (ПО) для процессора БЦО камеры диапазона 0,9...1,7 мкм, а также ПО для ПЭВМ, используемого для получения видеосигнала с БЦО, его цифровой обработки и отображения. К БЦО предъявляют следующие требования:

– БЦО должен осуществлять прием, обработку и передачу видеосигнала разрядностью 14 бит с МФПУ формата 320×256 с частотой не менее 50 кадров в секунду;

- подключаться к ПЭВМ по протоколу USB;
- обрабатывать набор цифровых сигналов, мультиплексированный на прерывания, данные и управляющие сигналы, как показано на рис. 2;
- платы, на которых реализуется БЦО, должны серийно или массово производиться на территории РФ.



**Рис. 1. Структурная схема прибора на основе МФПУ из InGaAs**



**Рис. 2. Тактовые диаграммы синхросигналов МФПУ**

Одной из основных проблем является выбор платы. На территории России производят в основном платы АЦП, которые имеют ряд недостатков:

- недостаточную частоту (платы с высокочастотными АЦП имеют высокую стоимость и большие размеры, применение их не оправдано);
- не способны принимать цифровой сигнал.

Единственное, что реально использовать в качестве аппаратной части БЦО, — плата USB2185 на основе процессора ADSP2185M.

Программное обеспечение, примененное в макетном образце камеры, выполняет следующие функции:

- настройка платы ввода цифрового сигнала (USB2185) для синхронизации, приема и передачи данных;
- управление функциями камеры и реализация специального протокола общения микроконтроллера ADSP2185 с базовым управляющим блоком прибора, изготовленным на основе ПЛИС Xilinx XC2S300E;
- преобразование полученных цифровых данных в изображение в соответствии с форматом передачи данных матрицы;

- получение данных о дефектных элементах матрицы;
- проведение цифровой фильтрации одно- и многоэлементного цифрового массива в целях устранения шумов и выравнивания чувствительности;
- отображение оцифрованного с МФПУ массива данных с частотой до 50 Гц;
- управление настройками контраста и яркости, калибровками темного и освещенного уровней;
- единичная и многокадровая съемка полученных с МФПУ изображений с сохранением в файлы на носителе.

При построении программно-технического комплекса (ПТК) возникает ряд проблем.

1. *Низкая пропускная способность платы USB2185.* Загрузка контроллера AVR платы зависит от количества данных, проходящих через него в единицу времени. При минимальной загрузке, когда через него проходят только данные яркостей пикселей камеры, необходимая пропускная способность составляет  $320 \times 256 \times 50 = 4\,096\,000$  слов в секунду (машинное слово памяти данных ADSP составляет 16 бит).

При максимальной загрузке через AVR проходят все сигналы, поступающие по прерыванию пиксела, в том числе на служебных интервалах синхронизации, без предварительной выборки. При этом необходимая пропускная способность составляет 5 632 000 слов в секунду.

Согласно техническому паспорту изделия [3], максимальная пропускная способность платы USB2185 составляет 6 мегаслов в секунду.

Таким образом, при максимальной загрузке контроллера требуемая пропускная способность составит 94 % от максимально возможной, что недопустимо высоко (из практических соображений должен оставаться запас не менее 25 %). Это означает, что необходимо провести предвыборку передаваемых данных для уменьшения их количества.

2. *Высокие требования к скорости обработки платой поступающего сигнала.* Оценив необходимую скорость обработки основной действующей подпрограммы, которой является обработчик прерывания по пикселу, получаем максимально возможное количество тактов (10) с учетом дополнительных прерываний и интервалов пересылки данных. Таким образом, появляется жесткое ограничение на время обработки одного пиксела: не более 10 тактов.

3. *Высокие требования по пропускной способности ПЭВМ.* Поскольку плата имеет ограничения по пропускной способности, поток, в котором осуществляется ввод данных с помощью драйвера платы, будет периодически находиться в состоянии ожидания данных, предоставляемых драйвером. В связи с этим следует применять асинхронный метод ввода данных.

4. *Высокие требования к скорости обработки ПЭВМ.* Согласно ТЗ, ПЭВМ должна обрабатывать получаемые данные со скоростью, достаточной для отображения с частотой 50 Гц. Это означает, что

время обработки одного кадра (с учетом интервала ввода) должно составлять не более 0,02 с.

Требования к пропускной способности и скорости обработки для платы противоречат друг другу, поскольку, с одной стороны, для предвыборки данных необходимо выделение процессорного времени, а с другой — количество выделяемого времени необходимо сократить. Минимальное время интервала ввода данных ПЭВМ непосредственно зависит от коэффициента загрузки AVR платы. Сокращение времени интервала ввода данных ПЭВМ позволяет выделить больше процессорного времени на обработку полученных данных. Кроме того, можно отметить следующие проблемы, возникающие при разработке:

– хотя для процессора ADSP возможно выполнение нескольких команд в один такт [4], в рамках данной задачи распараллеливание затруднено из-за высокой связанности команд по данным. Поскольку в любой момент времени проводится обработка только одного пиксела, то при распараллеливании возникают конфликты типа «чтение после записи» и «запись после чтения». Данная проблема может быть решена только так называемой ручной оптимизацией кода, поскольку для ADSP отсутствуют оптимизирующие компиляторы;

– ограничение возможности смоделировать поведение процессора, связанное с отсутствием качественных эмуляторов ADSP, их высокой сложностью и невозможностью имитации аппаратных эффектов (например, гонок сигналов).

В отношении микропрограммы ADSP для уменьшения нагрузки на AVR можно выбрать один из двух вариантов действий:

- 1) собственно улучшение — для разгрузки AVR уменьшить поток данных с помощью предвыборки;
- 2) осуществлять ручной контроль за корректностью выходных данных платы.

С учетом того, что такой контроль проводить проблематично ввиду больших объемов данных, а наличие ошибок, связанных с некорректно переданными данными, сложно обнаружить, решено использовать первый вариант.

Наиболее быстрым методом выборки при вводе является сравнение, поэтому можно при каждом поступлении пиксела сравнивать его значение с эталонным. В этом случае возникают две проблемы: во-первых, выбор эталонного значения, которое не должно встречаться при вводе данных с матрицы, и, во-вторых, изначально электронными схемами не предусмотрены дополнительные установки для определения пропускаемых интервалов.

Первую проблему удалось решить в результате консультации с разработчиками матрицы. Было выяснено, что матрица может направлять значения, близкие к нулевым, лишь в исключительных случаях (при крайне низком — ниже минимально допустимого времени накопления либо при обзоре объектов, имеющих температуру,

близкую к абсолютному нулю). Таким образом, в качестве эталонного значения выбран нуль.

Вторую проблему решили путем установки на выход мультиплексо-ра матрицы дополнительной несложной аппаратной логики, в результате чего на плату в промежуточных интервалах подается нулевое слово.

Существуют следующие способы увеличения скорости обработки:

– правильный выбор необходимых прерываний с исключением остальных;

– использование возможности параллельного выполнения нескольких команд в один такт;

– минимизация непроизводительных действий (в частности, пересылок данных).

Второй способ исключен ввиду проблем, связанных с его использованием.

В итоге располагаем шестью возможными прерываниями, из которых три (SERVICE, LSYNC и VIDEOSYNC) дублируют друг друга, следовательно, могут не обрабатываться (а значит, и не подаваться на вход). То же относится к прерыванию DATA. Прерывание LSYNC может быть исключено, поскольку не предоставляет дополнительной информации. Таким образом, обрабатываются лишь два прерывания из шести: FRAME и CLK (PIXEL).

Минимизацию пересылок проводят следующим образом:

– используют теневой набор регистров, что позволяет избежать хранения данных в памяти во избежание порчи их в основном цикле работы ADSP;

– программу строят таким образом, чтобы результаты работы одной команды сразу становились аргументами другой, без дополнительных пересылок.

**Модуль ввода ПО ПЭВМ.** При постоянном пребывании потока ввода в состоянии ожидания использовать синхронный ввод не представляется возможным. Это нецелесообразно еще и потому, что при синхронном вводе нельзя отправить команду на продолжение ввода сразу по окончании предыдущего, что означает потерю частей байтового потока. Следовательно, необходимо выбрать асинхронный режим ввода. Для этого функции, отвечающие за ввод данных через драйвер платы, выносят в отдельный поток, который обменивается с потоком обработки через буфер.

С одной стороны, время ввода данных стремится к нулю, так как обработка производится в другом потоке, а время копирования массива ничтожно мало по сравнению с временем его обработки. С другой стороны, поток ввода сам по себе требует времени — фактически, на многоядерных системах этому потоку отдают одно из ядер процессора. Работа на одном ядре хотя теоретически и возможна, но для этого требуется быстроедействие ЦП не менее 3,2 ГГц, произвести же дополнительную обработку на одноядерном ЦП невозможно, поскольку это может привести к блокировке системы.

**Модуль обработки видеоизображения.** Для повышения производительности *модуля обработки видеоизображения* были использованы следующие приемы:

1) реализация части подпрограмм на ассемблере с применением технологии SSE, позволяющей одновременно обрабатывать четыре значения типа float за проход; удобство использования SSE обеспечивается кратностью формата матриц четырем;

2) выравнивание данных, уменьшающее число кэш-промахов и обращений к памяти;

3) использование библиотеки OpenGL для отображения видеоизображения позволяет перенести часть расчетов, связанных с отображением и преобразованием готового изображения, на видеоадаптер.

В заключение отметим, что при разработке ПО для процессора блока цифровой обработки видеоизображений на ПЭВМ предъявляются взаимоисключающие требования: высокая пропускная способность ПЭВМ при высоких скоростях обработки ПЭВМ. Возникает проблема поиска компромиссного решения.

Исследование, проведенное в данной работе, позволило выбрать подходящую методику для разработки ПО, которая наиболее эффективно решает проблему противоречивых требований. Поскольку алгоритм цифровой обработки реализуется не аппаратной логикой, а с помощью микропрограмм, под управлением которых работают БЦО, то эта методика может быть применена для широкого класса задач, при решении которых следует провести «ручную оптимизацию» программного кода. С использованием описанной методики разработано ПО для процессора БЦО камеры диапазона 0,9...1,7 мкм, которое продемонстрировало достаточную скорость обработки данных. Разработанное ПО применяется как для подключения матриц формата 320×256 к ПЭВМ в целях просмотра видеоизображения, так и на специализированных стендах при обработке получаемых изображений для выявления характеристик подключаемой матрицы. При этом использованные подходы делают возможным получение результата в режиме времени, близком к реальному.

## СПИСОК ЛИТЕРАТУРЫ

1. Тарасов В.В., Якушенков Ю.Г. Двух- и многодиапазонные оптико-электронные системы с матричными приемниками излучения. М.: Логос, 2007. 192 с.
2. Ллойд Дж. Системы тепловидения / пер. с англ. под ред. А.И. Горячева. М.: Мир, 1982. 416 с.
3. Руководство пользователя и программиста USB2185 // R-Technology Ltd. URL: [http://www.r-technology.ru/docs/USB2185/USB2185ProductDataRev1\\_1.pdf](http://www.r-technology.ru/docs/USB2185/USB2185ProductDataRev1_1.pdf)
4. Комаров А.В. Цифровые сигнальные процессоры. Обнинск, 2003. 141 с.

Статья поступила в редакцию 25.10.2012