

УДК 004.9

**ОРГАНИЗАЦИЯ СИСТЕМЫ АНАЛИЗА СИГНАЛОВ В РЕЖИМЕ
РЕАЛЬНОГО ВРЕМЕНИ НА БАЗЕ ОПЕРАЦИОННОЙ СИСТЕМЫ WINDOWS**

Д.В. НОСКО

*Белорусский государственный университет информатики и радиоэлектроники
П. Бровка, 6, Минск, 220013, Беларусь**Поступила в редакцию 24 апреля 2007*

Проведен анализ причин, затрудняющих реализацию программных приложений реального времени при работе под управлением операционной системы Windows. Предложен способ реализации аппаратно-программного комплекса ввода, обработки и отображения вибрационных сигналов в режиме реального времени на базе операционной системы Windows.

Ключевые слова: система реального времени, аппаратно-программный комплекс, сигнал, вибрация.

Введение

Ввод, обработка и отображение сигналов являются необходимыми элементами различных аппаратно-программных комплексов (АПК). Функционирование таких комплексов, решающих контрольно-измерительные задачи, предполагает работу в режиме реального времени. В соответствии с [1] аппаратно-программные комплексы реального времени (АПКРВ) можно разделить на четыре класса:

а) операционная система (ОС) отсутствует, программирование осуществляется на машинно-ориентированном языке уровня микропроцессоров;

б) построенные на базе минимального ядра реального времени; ядро представляет собой набор программ, выполняющих типичные, необходимые для встроенных систем низкого уровня функции, такие как операции с плавающей запятой и минимальный сервис ввода/вывода;

в) включающие ядро реального времени и инструментальную среду; этот класс АПК обладает многими чертами АПК, построенных на базе ОС с полным сервисом; сюда включены такие средства, как дистанционный символьный отладчик, протокол ошибок и другие средства разработки; часто доступно параллельное выполнение программ; г) использующие ОС с полным сервисом; такие ОС могут быть применены для любых приложений реального времени. Разработка и исполнение прикладных программ ведутся в рамках одной и той же системы.

Основными критериями выбора ОС для решения задач реального времени, являются ее соответствие техническим характеристикам проектируемого аппаратно-программного комплекса и ценовой фактор (экономическая составляющая использования данной ОС). Технические характеристики позволяют оценить возможности ОС по таким основным параметрам как критическое время реакции на событие, наличие необходимых инструментальных средств и драйверов устройств. К экономическим критериям оценки ОС

относятся стоимость операционной системы и средств разработки, а также цена самой разработки, включая затраты на освоение системы, сопровождения и возможных рисков [2].

ОС Windows не разрабатывалась как операционная система реального времени (ОСРВ), однако ввиду наличия таких достоинств как: большой набор прикладных программ; мощный программный интерфейс WIN32; широкий выбор средств отладки, значительное количество специалистов, знающих эту систему, рассматривается возможность ее использования как базовой ОС для АПКРВ. Стоимость разработки и отладки программных систем для этой ОС значительно меньше по сравнению с системами реального времени. ОС Windows включает в себя элементы систем реального времени: двухуровневую систему обработки прерываний и процессы с приоритетами. Как следствие, для задач мягкого (квази-) реального времени на ее базе разработан ряд контрольно-измерительных систем [3].

В связи с этим представляет интерес возможность использования ОС Windows в качестве базовой ОС для построения аппаратно-программного комплекса ввода и отображения сигналов в режиме реального времени и условий, при которых возможен непрерывный прием, обработка и отображение сигналов со скоростью передачи данных до 1 Мбайт/с.

Анализ ОС Windows как ОСРВ

Согласно критериям, определяющим системы реального времени [4–7], ОС Windows не соответствует им в полной мере по следующим причинам.

1. *Неопределенность времени реакции на прерывание.* В ОС Windows нельзя точно установить время от момента генерации прерывания устройством до момента начала выполнения процедуры обработки этого прерывания. В Windows доступ к прерываниям осуществляется из драйвера ядра, а сами прерывания обрабатываются в два этапа: сначала вызывается процедура обработки прерывания (ISR, Interrupt Service Routine), осуществляющая критическую обработку, а основная обработка прерывания происходит в процедуре отложенного вызова (DPC, Differed Procedure Call). Все процедуры отложенного вызова выполняются с одинаковым уровнем приоритета в порядке поступления. Архитектура Windows подразумевает, что процедура обработки прерывания может быть вытеснена другой процедурой обработки прерывания с более высоким приоритетом. Но поскольку все процедуры отложенного вызова имеют одинаковый уровень приоритета, то даже процедура, инициированная обработчиком прерывания с более высоким приоритетом, будет вынуждена ждать завершения выполнения других, ранее поставленных в очередь процедур отложенного вызова. Таким образом, критическое ко времени обработки приложение будет зависеть от времени обработки прерываний, поступивших от устройств, способных генерировать прерывания и зарегистрированных в операционной системе.

2. *Малое количество приоритетов.* Особенностью задач, решаемых в режиме реального времени, является использование потоков с приоритетом. В ОС Windows уровней приоритета реального времени всего 16. Соответственно при большом количестве потоков, работающих в операционной системе, возможна ситуация, когда поток критической задачи, выполняемой в реальном времени, будет разделять приоритет с несколькими другими потоками. Следовательно, время выполнения критической процедуры становится непредсказуемым.

3. *Возможность инверсии приоритетов.* При функционировании ОС Windows возможна ситуация, когда планировщик задач выделяет процессорное время процессу, имеющему не самый высокий приоритет. Такая ситуация может возникнуть, если процесс с более высоким приоритетом не может получить доступ к необходимому ему ресурсу в связи с тем, что этот ресурс занят вытесненным низкоприоритетным процессом. В такой ситуации нарушается слаженность работы алгоритма планирования задач и методов управления доступа к ресурсам, т.е. нарушается работа подсистемы вытесняющего планирования на основе уровней приоритета, в которой всегда должен выполняться поток с наибольшим приоритетом.

4. *Неопределенность времени реакции на системные вызовы Win32 API.* Время срабатывания системных вызовов Win32 API зависит от общей загруженности системы. Например, мьютекс-запросы [8], которые используются для синхронизации потоков и решения

задачи взаимного исключения, в ОС Windows обрабатываются в порядке поступления, а не в соответствии с приоритетами.

5. *Особенности управления памятью.* В Windows все процессы функционируют каждый в своем адресном пространстве. Это достигается за счет использования страничной виртуальной памяти. Диспетчер памяти ОС Windows решает задачи освобождения и управления виртуальной памятью. Поддержка виртуальной памяти основана на механизме подкачки страниц. Это означает, что в любой момент времени диспетчер памяти может инициировать процедуру выгрузки страницы на диск. Такая организация порождает неопределенность в момент, когда критическая задача должна получить страницу памяти [9].

Условия использования ОС Windows в качестве ОСРВ для задач ввода и обработки данных

При построении аппаратно-программного комплекса ввода и обработки данных в режиме реального времени необходимо создать условия функционирования ОС Windows, при которых ее недостатки как системы реального времени, не будут влиять на работу АПКРВ.

Критически важным для задачи ввода сигналов является своевременное выполнение процедуры передачи данных между аппаратной и программной подсистемами.

ОС Windows позволяет реализовать обмен данными следующими способами [10]:

- а) программно-управляемый обмен;
- б) обмен данными по прерыванию;
- в) обмен по каналу прямого доступа к памяти (ПДП);
- г) обмен в режиме прямого управления шиной.

Первые два метода позволяют программе непосредственно обмениваться данными с устройством. Все данные во время обмена с устройством проходят через процессор. Использование в процессе передачи данных процессора делает зависимым передачу данных от его загруженности. Возникновение задержек при считывании данных вследствие занятости процессора либо при работе механизма прерываний может привести к тому, что данные будут потеряны либо повреждены. Для АПКРВ такая ситуация недопустима.

Организация передачи данных по каналу ПДП либо в режиме управления шиной проходит асинхронно по отношению к программной части системы, которая получает и передает данные только через буферы, расположенные в оперативной памяти компьютера. Такой способ передачи данных позволяет задействовать процессор только для инициализации канала передачи данных, но не в процедуре передачи данных. Исключение из процедуры передачи данных процессора устраняет влияние на передачу данных механизма прерываний. Управление передачей данных осуществляет контроллер ПДП. Такая организация возможна для устройств, построенных на базе шин PCI, USB, IEEE 1394 и других, поддерживающих запись в оперативную память.

При использовании режима ПДП успех функционирования системы реального времени зависит только от подсистемы диспетчеризации процессов ОС Windows. Для обеспечения непрерывного ввода и обработки сигналов в режиме реального времени необходимо минимизировать количество процессов, работающих в системе и разделяющих процессор. Достигнуть этого можно путем запрета запуска приложений в операционной системе до начала либо во время работы АПКРВ. Так же необходимым условием является отсутствие сторонних устройств в компьютере, на котором работает программная часть комплекса, помимо необходимых для функционирования АПКРВ. Особенно важно, чтобы отсутствовали устройства, имеющие драйвера, не сертифицированные Microsoft, так как для таких устройств не гарантируется своевременное освобождение ресурсов после обработки процедуры прерывания. Выполнение этих условий в сочетании с установкой для потоков клиентского приложения высокого уровня приоритета позволяет получить необходимый объем процессорного времени для программной части АПКРВ. Таким образом, нивелируется требование наличия у базовой ОС большого количества приоритетов реального времени [11].

Для предотвращения выгрузки страниц памяти на жесткий диск механизмом виртуальной памяти необходимо обеспечить наличие достаточного количества физической оперативной памяти для функционирования ОС Windows и программной части АПКРВ.

Дополнительно необходимо установить запрет выгрузки страниц для критических ко времени выполнения процессов, а также обеспечить их периодическую активацию, так как в противном случае механизм виртуальной памяти ОС может выгрузить даже заблокированные страницы. Архитектуру программной части АПКРВ необходимо строить таким образом, чтобы минимизировать хранение данных в оперативной памяти, а главное, обеспечить неизменность количества используемой памяти в процессе функционирования. Одним из подходов к решению данной проблемы является динамическое сохранение принятых данных в файле на жестком диске. Современные жесткие диски обеспечивают скорость передачи данных в самом тяжелом для них старт/стоповом режиме порядка 8 Мбайт/с со скоростью доступа порядка 20 мс. Это значительно превышает пиковое значение скорости ввода данных в 1 Мбайт/с для аналогово-цифровых преобразователей (АЦП) с частотой дискретизации до 500 кГц. Таким образом, программная часть АПКРВ может хранить в оперативной памяти только текущую, обрабатываемую и отображаемую часть данных, которая имеет небольшие размеры (до 32 кбайт). В результате даже длительный ввод данных не потребует увеличения используемой оперативной памяти и, как следствие, выгрузки страниц памяти на жесткий диск.

Организация АПКРВ ввода, обработки и отображения вибрационных сигналов

АПКРВ ввода, обработки и отображения вибрационных сигналов строится на базе ОС Windows с использованием всех описанных выше механизмов обеспечения работы в реальном режиме времени. Программная часть АПКРВ решает две основные задачи:

- а) ввод данных;
- б) обработка и отображение.

Организация процедуры ввода данных. Аналогово-цифровое преобразование сигналов, отражающих вибрационный процесс, осуществляется специальным модулем, подключаемым к компьютеру по USB-шине. Схема ввода данных в АПКРВ представлена на рис. 1.

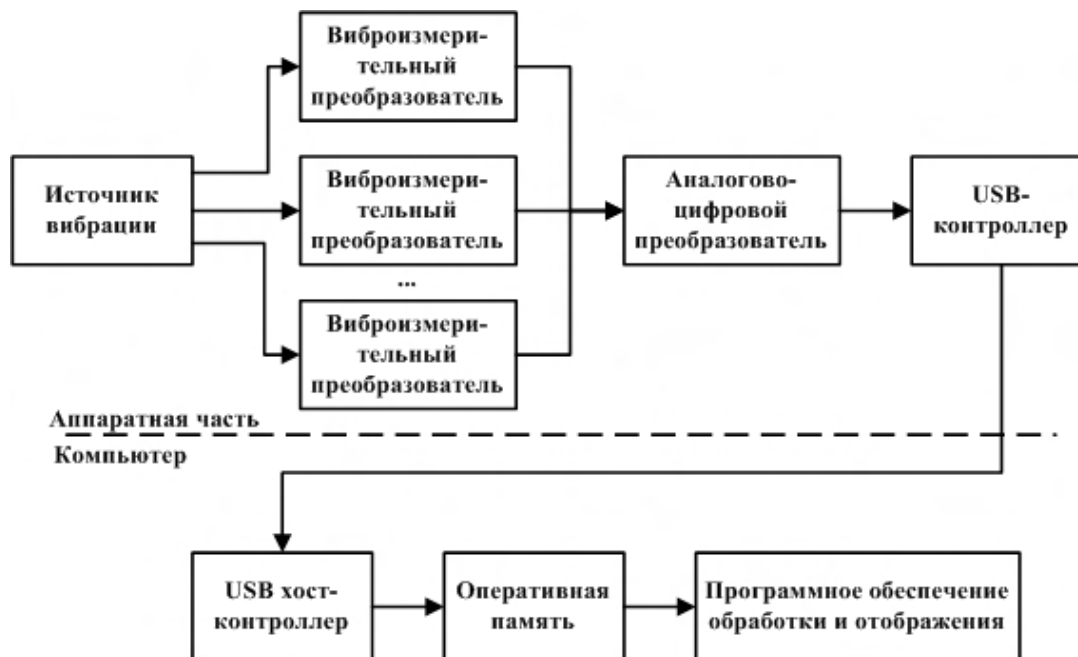


Рис. 1. Схема ввода данных в АПКРВ

Применение шины USB позволяет записать вибрационные данные в оперативную память, для промежуточного хранения которых выделяется буфер размером 128 Кб, организованный по кольцевой схеме. При заполнении кольцевого буфера данные вновь записываются с начала буфера, перезаписывая ранее сохраненные. Для передачи полученных цифровых кодов в функцию обработки-отображения необходимо реализовать механизм чтения данных из буфера. Для этого используются два указателя: указатель, содержащий адрес начала буфера и указатель текущей позиции записи данных в буфере. Буфер делится на две равные

части. При достижении указателем текущей позиции начала второй части кольцевого буфера инициируется копирование данных из первой части буфера и соответственно при его достижении начала первой части считываются данные второй части буфера (рис. 2). Такой подход позволяет избежать ситуации, когда указатель текущей позиции записи изменился, а данные из АЦП в кольцевой буфер еще записываются.

Организация процедуры чтения и обработки-отображения данных. При организации клиентского приложения с последовательной работой процедур чтения данных из кольцевого буфера и обработки-отображения данных возможна ситуация, когда буфер заполнен и порция готова к чтению, однако дескриптор управления ходом программы в данный момент находится в процедуре отображения данных.

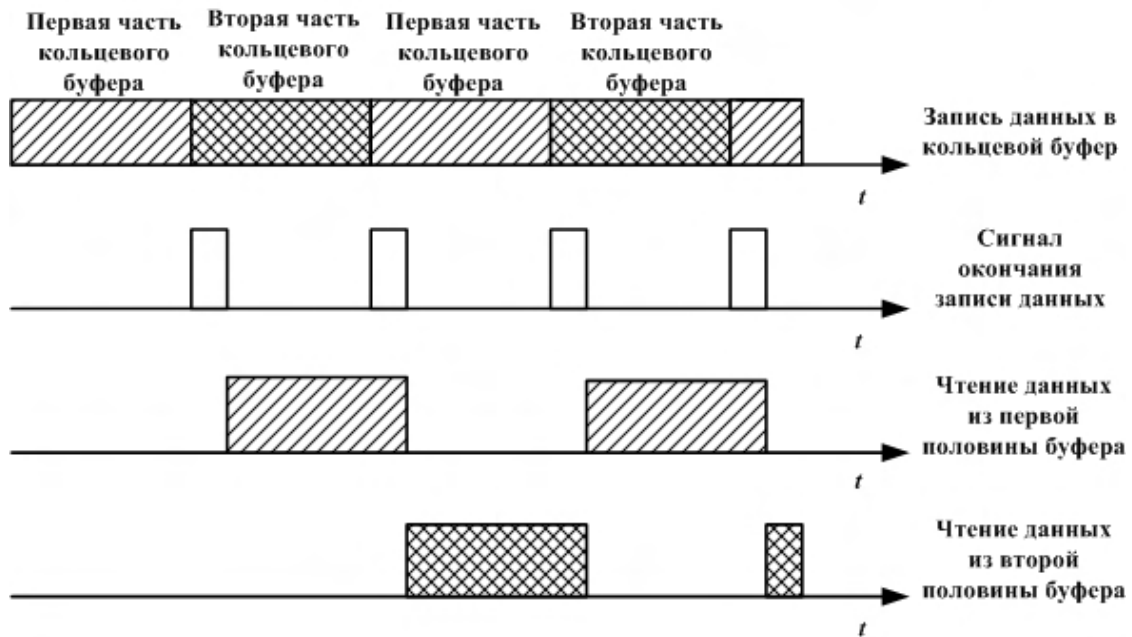


Рис. 2. Временная диаграмма процедуры считывания данных методом половин

Аналого-цифровое преобразование происходит независимо от того, считаны данные из буфера или нет, и, следовательно, возможна ситуация, когда хост-контроллер USB, управляющий передачей данных от АЦП, перезапишет непрочитанные из кольцевого буфера данные и они будут потеряны (рис. 3).

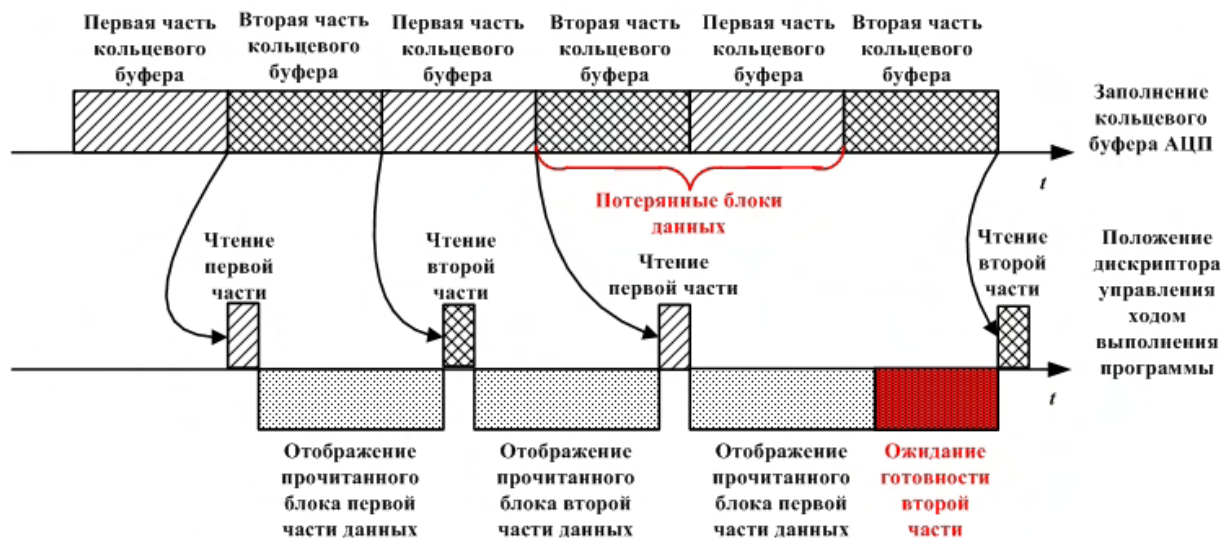


Рис. 3. Потеря данных при работе в последовательном режиме ввода и обработки-отображения данных

Использование последовательной организации работы процедур ввода и обработки-отображения возможно при обеспечении быстрого отображения принимаемых данных, т.е. обработка и отображение данных должны происходить за время, меньшее, чем время заполнения АЦП половины кольцевого буфера.

Экспериментальные исследования показали, что при применении стандартной библиотеки GDI затраты времени на графическое отображение данных на современном компьютере составляет порядка 30 мс. Скорость отображения данных зависит от многих факторов. Прежде всего, это быстродействие системы отображения в целом, т.е. быстродействие процессора, памяти и графического ускорителя, версии установленных графических библиотек. При этом следует учитывать, что общая производительность портативных компьютеров при решении подобных задач зачастую ниже, чем стационарных компьютеров, и, таким образом, влияет на максимально возможную частоту опроса каналов АЦП.

Повысить общую производительность можно путем распараллеливания процедур чтения данных из кольцевого буфера и их обработки-отображения. Согласно стратегии работы АПК ввода и обработки-отображения вибрационных сигналов в реальном режиме времени, приоритетным является непрерывность чтения вибрационных данных, поступающих от аппаратной части. Менее приоритетным является обработка и отображение полученных данных в реальном времени. Для обеспечения непрерывности чтения данных из кольцевого буфера необходима организация программной части АПКРВ, которая позволит обеспечить безусловную передачу управления потоку чтения данных в заданные моменты времени.

Специфика АПКРВ ввода и отображения вибрационных сигналов позволяет точно рассчитать ожидаемое время готовности данных для чтения. Это делает возможным безусловную передачу управления потоку чтения данных путем использования механизма таймера и механизма приоритетов.

Организация функционирования потока чтения данных следующая. В соответствии с частотой дискретизации АЦП вычисляется время готовности блока данных определенного размера. Согласно этому времени, настраивается работа таймера. Срабатывание таймера приводит к созданию нового потока чтения данных с приоритетом реального времени. Поток чтения данных проводит постоянный опрос флага готовности данных. Установка флага готовности запускает процедуру копирования данных из кольцевого буфера, по окончании которой поток завершается. Завершение потока чтения данных освобождает процессор для выполнения других задач, в частности, выполнения потока обработки-отображения данных, который работает до тех пор, пока по событию от таймера не будет инициирован очередной поток считывания данных из кольцевого буфера. Организация работы по такой схеме требует участия таймера операционной системы. Для этих целей предлагается использовать мультимедиа таймер ОС Windows, позволяющий задавать интервалы срабатывания с точностью до 1 мс.

Заключение

Таким образом, можно утверждать, что использование ОС Windows в качестве базовой ОС для аппаратно-программного комплекса анализа вибрационных сигналов со скоростью передачи данных до 1 Мбайт/с в режиме реального времени возможно. Это, прежде всего, обуславливается спецификой работы АПКРВ, которая позволяет точно рассчитать время готовности данных, что позволяет обойти основную проблему, которую призваны решать системы реального времени — проблему неопределенности возникновения критического события, требующего обработки. Дополнительным фактором является принадлежность задач, решаемых системой, к задачам мягкого (квази-) реального времени. К этим факторам необходимо добавить использование определенной организации АПКРВ. Прежде всего, это:

- 1) использование механизма прямого доступа к памяти для передачи данных от устройства к клиентскому приложению в процессе ввода данных;
- 2) обеспечение клиентскому приложению максимальной приоритетности при функционировании в операционной системе;

3) использование максимально оптимизированной ОС Windows и прикладных программных средств;

4) специальная организация взаимодействия процедуры чтения данных из кольцевого буфера и отображения данных с использованием механизма приоритетов и механизма таймеров операционной системы.

SYSTEM ORGANIZATION FOR SIGNAL ANALYSIS IN REAL-TIME MODE USING WINDOWS OPERATING SYSTEM

D.V. NOSKO

Abstract

Performed cause analysis, making difficult realization real-time software using Windows operating system. Proposed way for realization hardware and software package for capture, processing and presentation vibration signals in real-time on basis of Windows operating system.

Литература

1. *Золотарев С.* // Современная электроника. 2006. № 7. С. 52.
2. *Жданов А., Латыев А.* // PC Week. 2001. № 1. С. 22.
3. *Жданов А.* // PC Week. 1999. № 8. С. 17.
4. IEEE Standards Project P1003.4 POSIX Part 1: System Application Program Interface (API). Amendment 1: Rcaltime Extension. Draft 13. IEEE, 1992.
5. IEEE Standards Project P1003.4a Thread Extension for Portable Operating Systems. Draft 6. IEEE, 1992.
6. IEEE Standards Project P1003.4b POSIX Part 1: Rcaltime System API Extension. Draft 6. IEEE, 1993.
7. IEEE Standards Project P1003.13 Part 1: POSIX Rcaltime A)-plication Support (AEP). Draft 5. IEEE, 1992.
8. *Timmerman M.* // Real-Time Magazine 1997. № 2. С. 6–13.
9. *Соломон Д., Русинович М.*, Внутреннее устройство Microsoft® Windows 2000. Структура и алгоритмы работы Windows 2000 и NTFS 5. СПб., 2001.
10. *Русинович М., Соломон Д.*, Внутреннее устройство Microsoft® Windows: Windows Server 2003, Windows XP и Windows 2000. СПб, 2005.
11. *Ramamritham K., Shen C., Gonzalez O., Sen S., Shirgurkar S.* // IEEE RTAS'98 (Real-Time Technology and Applications Symposium), June 3–5, 1998. P. 32.