

ВЗАИМОСВЯЗЬ ЖИЗНЕННЫХ ЦИКЛОВ ОПЕРАЦИОННОЙ СИСТЕМЫ И МИКРОПРОЦЕССОРА ПРИМЕНИТЕЛЬНО К ОПЕРАЦИИ Понижения РАЗРЯДНОСТИ ВЫЧИСЛЕНИЙ

Приходько Дмитрий Игоревич

Магистр, инженер вычислительных и электронно-вычислительных машин, ФГУ ФНЦ НИИСИ РАН,

г. Москва

mitry1205@mail.ru

RELATIONSHIP OF LIFE CYCLES OF OPERATING SYSTEM AND MICROPROCESSOR IN RELATION TO OPERATION OF COMPUTATION REDUCTION

D. Prikhodko

Summary. The article examined the interaction of the operating system and the microprocessor as components of a multi-bit computing system in terms of life cycle theory and operation stage. An author's approach to considering the interaction of components of a multi-bit computing system is proposed. The features of the operation of the microprocessor in a multi-bit computer system were described, and from the point of view of life cycle theory, it was shown that the operating rules obey the cascading life cycle model. The features of operating the operating system in a multi-bit computer system were described, and from the point of view of life cycle theory, it was shown that the operating rules obey the version model of the life cycle. A structural diagram of the interaction between a multi-bit microprocessor and a multi-bit operating system was built using the example of the AMD64 microprocessor command system and using the Microsoft-family Windows multi-bit operating system as the basis.

Keywords: Operation, multi-bit operating system, • life cycle of information systems, AMD64 microprocessor.

Аннотация. В статье была рассмотрено взаимодействие операционной системы и микропроцессора как компонентов мультиразрядной вычислительной системы с точки зрения теории жизненных циклов, и этапа эксплуатации. Предложен авторский подход к рассмотрению вопроса взаимодействия компонентов мультиразрядной вычислительной системы. Были описаны особенности эксплуатации микропроцессора в мультиразрядной вычислительной системе, а с позиции теории жизненного цикла было показано что правила эксплуатации подчиняются каскадной модели жизненного цикла. Были описаны особенности эксплуатации операционной системы в мультиразрядной вычислительной системе, а с позиции теории жизненного цикла было показано что правила эксплуатации подчиняются версионной модели жизненного цикла. Была построена структурная схема взаимодействия мультиразрядного микропроцессора и мультиразрядной операционной системы на примере системы команд микропроцессора AMD64 и использования в качестве основы мультиразрядной операционной системы системы Windows семейства Microsoft.

Ключевые слова: Эксплуатация, мультиразрядная операционная система, жизненный цикл информационных систем, микропроцессор AMD64.

Введение

На момент создания данной статьи обзор взаимодействия компонентов мультиразрядных вычислительных систем ранее не было произведено. Важно отметить, что к мультиразрядным вычислительным системам относят вычислительные системы, которые способны работать на нескольких режимах работы с возможностью смены режима работы без использования перезагрузки.

Для существенного увеличения надежности выполнения задач микропроцессором при работе в космических [1–3] или иных неблагоприятных условиях, в частности это ионизирующее излучение, используются

несколько методов, которые являются классическими для технических систем.

Их можно описать следующим образом:

1. **Отключение поврежденных дублей.** Подразумевается для вычислительных систем, когда в неё внедрен избыточный функционал, и затем постепенно под воздействием ионизирующего излучения идет постепенное сокращение дублирующих компонентов, и в результате снижается производительность системы, однако функционал сохраняется полностью без изменения.
2. **Снижение функционала системы.** Подразумевает отключение избыточного функционала системы до максимально допустимого уровня, при

котором допускаем. Для этого может использоваться и операция понижения разрядности вычислений [4, 5].

При рассмотрении понятия эксплуатация с позиции теории жизненного цикла, в которой эксплуатация является одним из этапов, одним из самых сложных вопросов, возникающих в процессе анализа, является взаимодействие микропроцессора с используемой операционной системой, которая обладает возможностями работать на нескольких режимах работы микропроцессора без переноса приложений, и поэтому является мультиразрядной системой.

Указанная особенность работы операционной системы и микропроцессора уже подразумевает особый механизм взаимодействия с точки зрения этапа эксплуатации [6] из теории жизненного цикла. Анализ этой особенности и является целью настоящей статьи. Для выполнения анализа авторами, в качестве примеров технических изделий будет рассмотрен микропроцессор совместимый с системой команд AMD64, который относится к классу CISC (полный набор команд), и по построению стека имеет тип архитектуры Фон-Неймана и академика Лебедева. А для примера «версий» операционных систем будет взята Windows, поскольку существует в виде версий для всех режимов работы AMD64.

Методология

При изучении вопроса, описанного в цели статьи: взаимосвязь жизненного цикла операционной системы и микропроцессора применительно к операции понижения разрядности вычислений, необходимо определить каким образом осуществляется переключение режимов работы микропроцессора, и как именно на это должна реагировать операционная система. Однако, прежде чем рассмотреть эту взаимосвязь необходимо определиться с тем, к какому типу жизненного цикла относятся операционная система и микропроцессор архитектуры AMD64.

Набор команд AMD64 предусматривает 3 режима работы [7]:

1. **Real mod** — совместим с i8086. Это уровень прошивки BIOS и UEFI.
2. **Protected mod** — совместим с i386 и более новыми моделями. Этот режим используется в настоящее время на устаревающих компьютерах и бюджетных гаджетах (планшеты, ноутбуки маломощные).
3. **Long mod** — совместим с Amd Athlon 64. Это последний режим работы для микропроцессоров AMD64, и самый функциональный. Он работает практически во всех последних микропроцессорах

AMD и Intel. До некоторого времени использовалась еще и версия IA-64 (Itanium), однако на рынке она оказалась неудачной.

Из всех перечисленных режимов работы для ОС Windows и программ сейчас основным режимом работы является режим Long Mod, который вытесняет постепенно режим Protected mod, который был основным до 2010 года. До 2020 года от поддержки 32-битных микропроцессоров уже отказались компании виртуализации — VMWARE, Oracle выпускающая VirtualBox, и многие другие.

Последним решением, что этот тип приложений уходит в прошлое — это выпуск Windows 11 без поддержки этого режима.

Типы поддерживаемых приложений микропроцессором для каждого режима работы выглядит таким образом:

1. **Real mod.** Так как режим совместим с i8086, то работающие приложения могут использовать лишь возможности режима i8086 с некоторыми дополнительными функциями в виде использования регистров 32 бит общего назначения.
2. **Protected mod.** Типы приложений, которые поддерживаются на этом режиме, совместимы с процессором i386 и новее, что позволяет использовать доступные возможности, начиная с i386 и заканчивая некоторыми расширениями MMX, SSE, SSE2.0 которые присутствовали в последних поколениях моделей микропроцессоров.
3. **Long mod.** Для этого режима характерна работа всех последних на момент публикации статьи приложений, которые были выпущены для процессора AMD Athlon 64. Ограничения по работе вызваны только тем, что используются не все возможности последних поколений в виде SSE4.0 и новее.

Тем не менее, выпущенные программы для старых микропроцессоров, к которым относятся приложения Protected mod и Real mod вполне способны работать полностью на современных микропроцессорах при условии совпадения режимов, но иногда и режимов более новых.

Например запуск программы Real mod (приложение 16bit Windows) на режиме Protected mod на системе Windows NT 3.50 продемонстрирован на Рис. 1.

Важно отметить, что, хотя и приложение собрано старым компилятором под старые системы, особых трудностей в работоспособности на Windows NT 3.50 оно не испытывает особых проблем при запуске про-

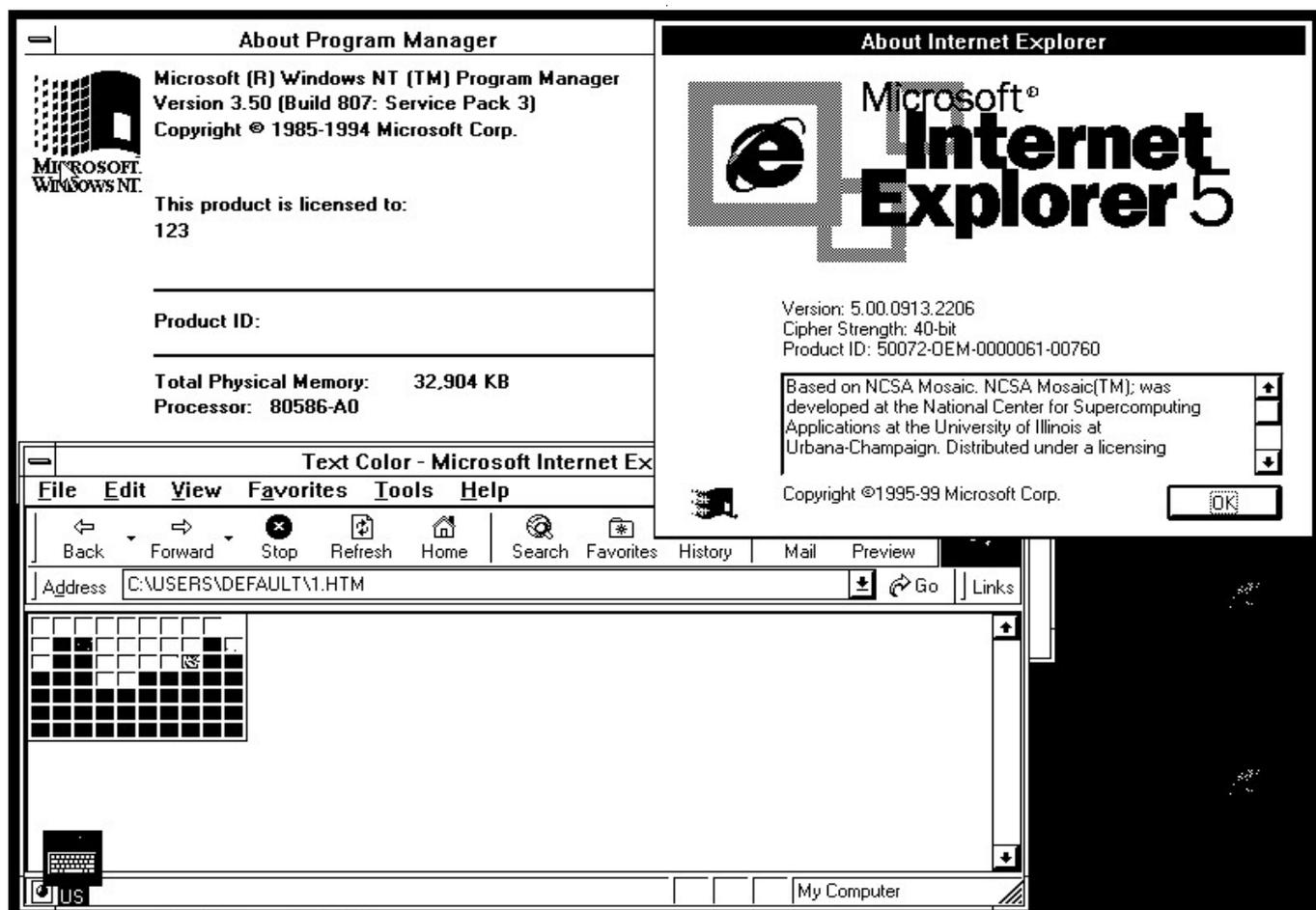


Рис. 1. Скриншот запуска программы Windows 3.1 на примере Internet Explorer 5.0, под системой Windows версии NT 3.50.

граммы, тем не менее в некоторых случаях из-за ограничений некоторые приложения запуститься не смогут. К таким приложениям относят в первую очередь драйвера и некоторые приложения при условии использовании низкоуровневых средств — дефрагментаторы, утилиты реестра и пр.

В свою очередь запуск приложений, написанных для режима Protected Mode, на режиме Long mode имеют свои ограничения, например, низкоуровневые утилиты и драйвера работать не будут полностью, приложения Real Mode не работают без специальных эмуляторов окончательно. На примере запуска программы GreenPad (см. рис. 2), которая была перенесена на старые версии ОС Windows не официально, прекрасно себя чувствует на Windows 10 64 bit (набор команд AMD64), которая является предпоследней версией Windows на момент публикации статьи.

Учитывая, описанные особенности поведения микропроцессора, можно констатировать факт того, что

современный микропроцессор с точки зрения предоставляемых возможностей может быть представлен как несколько версий, предоставляющий различный интерфейс без необходимости менять физически микропроцессор. Например, архитектура AMD64 включает наборы инструкций IA32 и x86 для совместимости с приложениями, которые были написаны для прошлых поколений микропроцессоров, так как в противном случае получится аппаратная несовместимость при выполнении программы. Например, программы, созданные для микропроцессора i8085 не могут быть выполненными на микропроцессоре i8086, что связано с различными механизмами адресации и подтверждено официальной документацией. Для решения этой задачи в те годы писали программные конверторы для языка assembler, и полученные решения в последствии перекомпилировали. Затем был создан эмулятор i8085 и на нем можно запустить систему CP/M-80 для этого поколения микропроцессоров.

Но так как каждый режим является ещё и не просто версией, а последней доступной версией в семействе

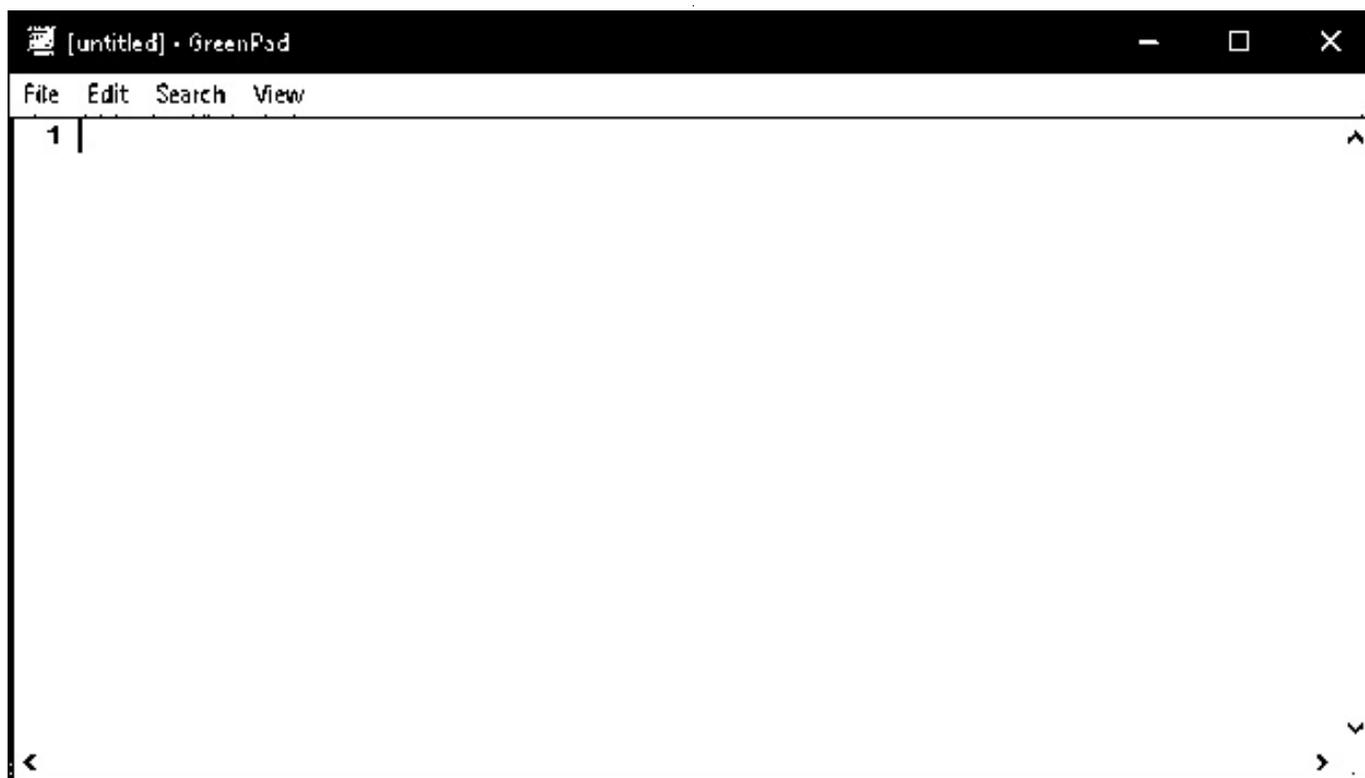


Рис. 2. Приложение GreenPad, перенесенное для использования в Win32s пакета, запущенное на системе Windows 10 64 bit.

микропроцессоров старого режима, то значит, что приложения смогут использовать максимально эффективным образом возможности каждого режима работы. Например, Real Mod допускает использование системы команд i286 в режиме 16 бит, хотя в этом микропроцессоре появились первые зачатки режима Protected mode.

Следовательно, микропроцессор с точки зрения теории жизненного цикла при условии использования операции жизненного цикла взаимодействует с операционной системой с позиции каскадной модели жизненного цикла.

Однако, поскольку вычислительные системы состоят не только из микропроцессора, но ещё и операционной системы, то возникает вопрос о характере поведения операционной системы при выполнении смены режима работы микропроцессора.

Ответ на этот вопрос можно дать следующим образом:

1. Смена режима микропроцессора должна будет неизбежно происходить смена «версии» операционной системы, что связано со сменой правил адресации.

2. Это влечет за собой замену или модификацию драйверов, а также перестройки набора функций в системе.
3. Перестройка функционала системы ведет к перестройке правил определения механизмов обращения к функциям операционной системы внутри программного обеспечения, что требует от программ специальной обработки.

Поэтому программы, которые будут работать в операционных системах должны обладать специальными свойствами, которые заключаются в следующих особенностях:

1. Программа содержит необходимый набор команд на выбранные режимы микропроцессора.
2. Для специальных областей программы, отвечающих за оптимизацию — должны быть обрмлены специальными инструкциями, и использоваться только для каждого из выбранных режимов работы. К примеру, у IA-32 программах используется возможность выбора разрядности стека — 32 и 16 бит, что использовалось в 90-х годах.
3. Формат исполняемого файла должен распознаваться всеми выбранными версиями системы.

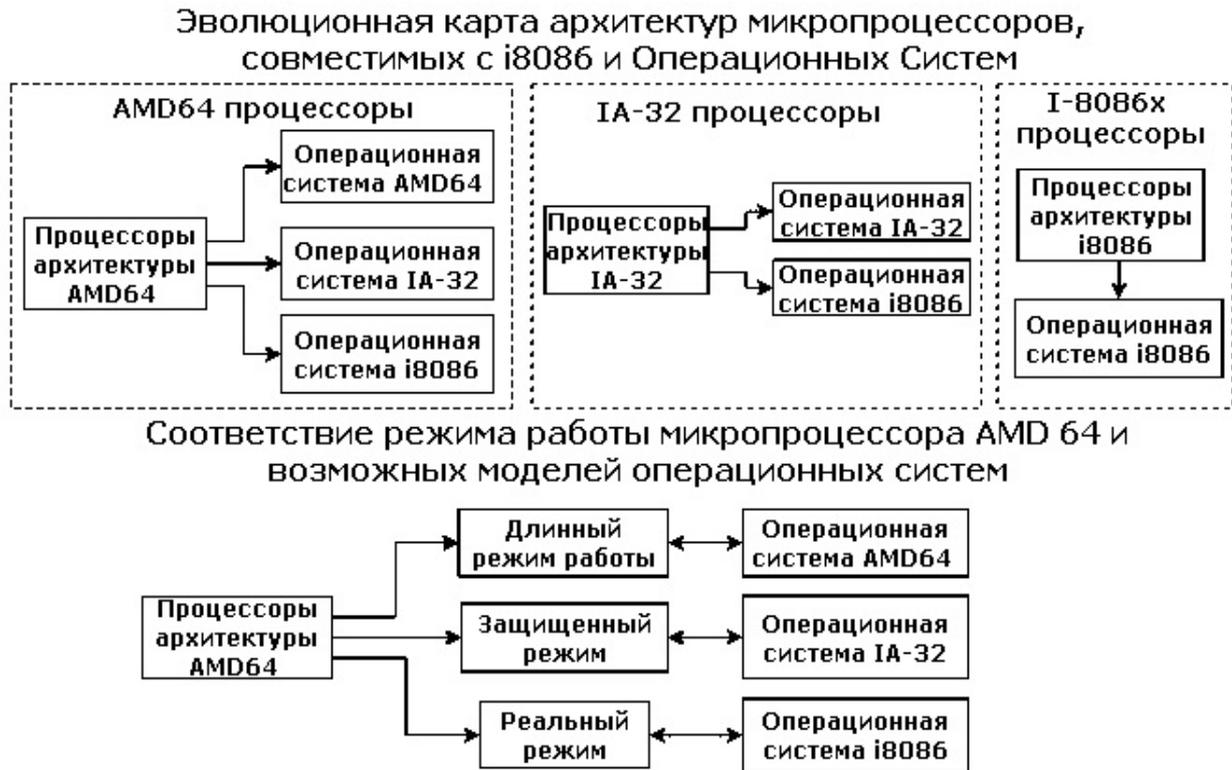


Рис. 3. Структурная схема взаимосвязи режимов микропроцессора и доступной операционной системы.

Следовательно, при смене режима работы или разрядности вычислений микропроцессора происходит полная смена «версии» операционной системы и некоторых приложений. По этой причине можно утверждать, что операционная система взаимодействует с микропроцессором по версионному жизненному циклу.

Результаты

Так как модель смены режима работы (разрядности вычислений) с точки зрения микропроцессора является жизненным циклом каскадного типа, тогда режимы работы микропроцессора с доступными операционными системами можно схематично построить в виде структурной схемы эксплуатации (см. рис. 3).

Поэтому, на основе рис. 3 получаются однозначные выводы об том, что смена режима работы микропроцессора согласно теории жизненного цикла, полностью подчиняется каскадному жизненному циклу, что объясняется несколькими важными причинами:

1. Весь функционал со всеми интерфейсами расположен на одном кристалле, что позволяет при выполнении смена режима не заменять физически устройство.

2. При смене режима работы со старшего на младший, появляются все возможности, которые были доступны для предыдущих поколений. При смене с AMD64 на режим IA-32 остается возможность использовать расширения MMX и SSE2.0.

Структурную схему взаимодействия операционной системы применительно к режимам работы микропроцессора, которая взаимодействует по версионному жизненному циклу, можно описать структурной схемой, которая изображена на Рис. 4:

По этой причине, взаимодействие операционной системы и микропроцессора, обладающими возможностями менять режимы работы при эксплуатации, можно представить в общем виде как взаимодействие версионной и каскадной модели жизненного цикла.

В частном случае на примере версий ОС Windows и микропроцессора AMD64 это взаимодействие может быть изображено схожим образом со схемой, изображенной на рис. 5.

Изображенная на рис. 5 структурная схема связи жизненных циклов операционной системы и микропроцессора демонстрирует, что жизненный цикл мультимедийного



**Соответствие режима работы микропроцессора AMD 64
и возможных моделей операционных систем**



Рис. 4. Структурная схема взаимосвязи операционной системы и доступных режимов микропроцессора.



Рис. 5. Структурная взаимосвязь режимов работы при эксплуатации микропроцессора и операционной системы.

тираэрядной вычислительной системы представляет собой связь жизненных циклов каскадного элемента (микропроцессор) и версионного элемента (операционная система).

Обсуждение

Механизмы смены режима работы микропроцессора и операционной системы необходимы в первую очередь

для того, чтобы добиться возможности эффективной работы в условиях экстремальной окружающей среды, например, космическом пространстве. Однако, требуется уточнить за счет чего достигаются указанные возможности в контексте анализа взаимодействия жизненных циклов микропроцессора и операционной системы.

Так как в простейшем случае, мультиэрядная операционная система состоит из нескольких версий

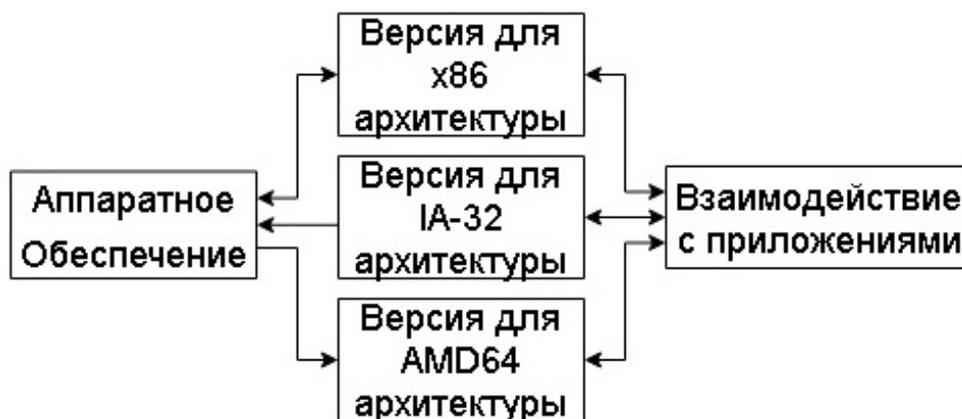


Рис. 6. Общая структура мультиразрядной операционной системы, реализованной для микропроцессора AMD64

стандартных операционных систем, функционирующих на одном или нескольких режимах адресации микропроцессора, то при смене режима микропроцессора произойдет просто переключение за счет внутренних механизмов операционной системы с одной «версии» под режим адресации на другую «версию» поддерживающую другой режим работы.

Например, структура мультиверсионной операционной системы, построенной на основе семейства операционных систем Windows, в упрощенном виде имеет вид, аналогичный рис. 6. Реализация структуры на этой системе требует, как техническую часть, для которой необходимо реализовать как минимум следующие механизмы:

1. Механизм загрузки каждой версии для каждого из выбранных режимов работы микропроцессоров в соответствии с методиками сохранения затем резервных версий, предназначенных для других режимов работы, в соответствующее место — или на жесткий диск или в оперативную память.
2. В соответствии с механизмом загрузки каждой версии, необходимо реализовать особые механизмы переключения версии с жесткого диска или оперативной памяти, которые будут осуществлять горячую смену версии без необходимости перезагрузки.
3. Механизмы драйверов операционной системы. Необходимо сделать специальные драйвера, которые смогут работать в нескольких режимах работы. Например, это драйвера дискового контроллера или адресов BIOS. На него очень сильно похож драйвер vxd, однако он требует возможностей микропроцессора i386 и новее, что для использования в 16-ти битном окружении не всегда возможно.

4. Разработку и проектирования особого программного обеспечения. Это создание специального программного обеспечения, которое необходимо для того, чтобы можно было при смене режима работы избежать потери данных в программах и потери времени на восстановление полученных результатов.

Для микропроцессора в свою очередь нужны следующие механизмы:

1. Доработки механизма смены режима работы, которые будут сохранять значения стека и регистров с целью дальнейшей обработки после переключения.
2. Механизмы преобразования целочисленных вычислений, чисел с фиксированной точкой и также вычислений для чисел с плавающей точкой.
3. Возможность использовать резервные триггеры. Это необходимо для повышения надежности сохраняемых данных внутри микропроцессора.

Описанные в статье требования к микропроцессору и операционной системе объясняется минимальным размером адреса в регистре, на который можно ориентироваться. В десятичной системе счисления размер адреса равен 0,7 от размера регистра в текущем режиме работы. Потеря порядка 0,3 значащих значений от числа является в рамках мультиразрядных систем допустимой, так как повреждение триггера в регистре может быть обнаружено не сразу или в случае выполнения критического процесса может привести к фатальному завершению, как программы, так и работы всей системы по причине повреждения целостности адресации в системе.

В качестве практического применения приведенного в статье подхода, можно применить к полученному результату [8] автором работ.

Рассмотренное взаимодействие жизненных циклов в статье дает первичное представление о возможностях проектирования мультиразрядной операционной системы и микропроцессора, которые позволяют с точки зрения техники существенно повысить надежность вычислительной системы.

Заключение

В статье были рассмотрены вопросы взаимодействия внутренних механизмов мультиразрядной вычислительной системы с точки зрения теории жизненных циклов. Был рассмотрен характер эксплуатации микропроцессора в экстремальных условиях окружающей среды, в результате анализа которого было выявлено, что эксплуатация микропроцессора в мультиразрядных вычислительных системах соответствует каскадному жизненному циклу эксплуатации.

В свою очередь рассмотрение характера эксплуатации операционной системы, когда она эксплуатируется в рамках мультиразрядной вычислительной системы с микропроцессором, который может изменять раз-

рядность, приобретает версионный характер эксплуатации с точки зрения теории жизненного цикла.

В обсуждении были описаны основные особенности эксплуатации микропроцессора, для внедрения которых, необходимо разработать специальные механизмы, в числе которых внутренние механизмы по преобразованию типов вычислений, сохранения слежка регистров, а также дополнительный набор триггеров.

В обсуждении были описаны основные особенности эксплуатации операционной системы, для создания которой из версии необходимы дополнительные специальные механизмы, из которых необходимыми в первую очередь являются механизм переключения версии операционной системы, особый механизм загрузки операционной системы с учетом хранения версий, особая реализация драйверов и программного обеспечения.

Была построена структурная схема взаимодействия микропроцессора и операционной системы как компонентов вычислительной системы.

ЛИТЕРАТУРА

1. Dong, Z., Guo, Y., Gong, Y., Li, C. A high reliability radiation hardened on-board computer system for space application (2016) Proceedings — 2016 6th International Conference on Instrumentation and Measurement, Computer, Communication and Control, IMCCC2016, № 7774866, pp. 671–674. DOI: 10.1109/IMCCC.2016.82.
2. Narita, T., Taeda, M., Kato, M., Kusano, M., Masukawa, K., Takada, M., Takada, H., Ishida, T., Fukuda, S., Matsuzaki, K., Takahashi, T., Nomachi, M. High-reliability SpaceWire engine implemented on the SOISOC3 microprocessor: Components, short paper (2016) Proceedings of the 2016 7th International SpaceWire Conference, SpaceWire 2016, № 7771608. DOI: 10.1109/SpaceWire.2016.7771608.
3. Polo, Ó.R., Sánchez, J., da Silva, A., Parra, P., Hellín, A.M., Carrasco, A., Sánchez, S. Reliability-oriented design of on-board satellite boot software against single event effects (2021) Journal of Systems Architecture, 114, № 101920. DOI: 10.1016/j.sysarc.2020.101920
4. Мокряков, А.В. & Приходько, Д.И. Общие концепции и идеи теории эксплуатации в контексте вычислительной техники. (2020). Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. № 12. С. 88–92.
5. Горшков, В.В., Приходько, Д.И. & Мокряков, А.В. Устройство для управления конфигурацией вычислительной системы. (2021). Патент на полезную модель 207176 U1, 15.10.2021. Заявка № 2021118609 от 25.06.2021.
6. Горшков, В.В., Приходько, Д.И. & Мокряков, А.В. Устройство для управления разрядностью вычислений. (2022). Патент на полезную модель 209758 U1, 22.03.2022. Заявка № 2021118607 от 25.06.2021.
7. Мокряков, А.В. & Приходько, Д.И. Критерии анализа эксплуатационных характеристик первичных мультизагрузчиков. (2020). Современная наука: актуальные проблемы теории и практики. Серия: Естественные и технические науки. № 12. С. 93–96.
8. М.В. Дебиев, Р.А.-М. Магомадов, Т.Ш. Амхаев, Ш.З. Зиниев. Эффекты внедрения автоматизированных электроэнергетических систем на основе интеллектуальных сетей. (2021). Вестник ГНТУ. Технические науки, том XVII, № 1 (23). С. 15–20.

© Приходько Дмитрий Игоревич (mitry1205@mail.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»