

Операционная система реального времени QPRT

В. Ю. Егоров email: vec@cryptosoft.ru

НТП «Криптософт», Пенза, Россия

***Аннотация.** Излагаются принципы построения защищенной операционной системы реального времени QPRT.*

***Ключевые слова:** операционная система, реальное время, QP ОС, QPRT*

Введение

Научно-техническое предприятие «Криптософт» является предприятием-разработчиком средств и систем защиты информации (аппаратно-программных, программно-аппаратных, программных), а также ведет работы по перспективным направлениям разработки, в том числе – квантовые коммуникации, разработка операционных систем. Остановимся более подробно на направлении разработки операционных систем. НТП «Криптософт» в инициативном порядке разработана защищенная операционная система QP ОС, являющаяся отечественной разработкой, выполненной полностью «с нуля» специалистами предприятия [1,2].

Защищенная операционная система QP ОС не является очередным клоном Unix-систем, основанных на свободно-распространяемых репозиториях, основа защищенной операционной системы QP ОС – оригинальное ядро собственной разработки. Защищенная операционная система QP ОС включена в Реестр отечественного программного обеспечения, а также имеет сертификат соответствия ФСБ России и может быть использована как базовая программная платформа для построения на ее основе автоматизированных систем в защищенном исполнении. Она применяется в АСЗИ силовых ведомств и органов государственной власти.

Полученный при разработке защищенной операционной системы QP ОС научно-технический задел позволил специалистам предприятия создать операционную систему QPRT, как базовую платформу систем со встроенным программным обеспечением, для применения в изделиях предприятия [3]. ОС QPRT разработана как операционная система реального времени, в отличие от защищенной операционной системы QP ОС, которая является операционной системой общего назначения.

Основное отличие операционной системы реального времени от операционной системы общего назначения, в том, что, для правильного выполнения программ необходим не только правильно составленный алгоритм и правильно запрограммированный, но и должны быть соблюдены ограничения по времени выполнения тех или иных программных модулей, исполняющихся в среде операционной системы. Так, примерами систем реального времени могут быть VxWorks или операционная ОС РВ, развиваемая НИИ СИ РАН [4].

Необходимость в разработке операционной системы QPRT возникла из потребности предприятия в программной платформе для встраиваемых систем.

1. Общие сведения о системе

При создании операционной системы QPRT были заданы следующие требования:

- малые размеры исполняемого кода системы;
- минимальные требования к аппаратной платформе исполнения ОС (бинарный код ядра системы занимает менее 1 Мб на устройстве хранения),
- поддержка многопроцессорности реального времени исполнения программных модулей,
- возможность совмещения в одной операционной среде исполнения задач разного уровня критичности (приоритета исполнения);
- совместимость с различными аппаратными платформами для встраиваемых систем, в том числе отечественного производства.

При использовании QPRT как базовой программной платформы, в качестве встраиваемых систем могут быть использованы такие системы, как

- системы на чипе (SoC) на основе микроконтроллеров, в том числе, предназначенные для управления производственными процессами и системами,
- устройства Интернета вещей (IoT),
- системы управления интеллектуальными беспилотными устройствами.

Кроме этого, операционная система QPRT может быть использована в качестве учебной платформы для изучения основ построения ОС реального времени при обучении студентов соответствующих специальностей.

При разработке операционной системы QPRT была поставлена задача по обеспечения совместимости с разными аппаратными платформами, в том числе отечественными, используемыми при

создании встраиваемых систем. На сегодняшний день имеется задел для реализации операционной системы QPRT на следующих аппаратных платформах: Intel x86, Intel x64, ARM v7 (A, M), ARM v8, отечественной аппаратной платформой «Комдив-64».

В целях отладки и изучения принципов работы QPRT она также может быть запущена как процесс в ОС Microsoft Windows. В дальнейшем также планируется рассмотрение вопроса применения системы для процессоров семейства RISC-V.

В основу принципа построения операционной системы QPRT положен принцип функционирования в среде ОС единственного процесса – самой системы. Все остальные исполняемые модули являются нитями (потоками) основного процесса. Число нитей ограничено только возможностями аппаратных ресурсов системы. Такой подход значительно уменьшает код операционной системы, повышает её производительность и применимость для микроконтроллеров.

У каждой из нитей внутри процесса системы могут быть один из трех уровней приоритета, как показано в таблице. Это обеспечивает возможность совмещения исполнения в одном процессе нитей с разным уровнем критичности. Соответственно, исполнение нитей наивысшего приоритета не может прервать ни одна из одновременно с ней исполняющихся нитей или аппаратное прерывание. Выполнение нити с низшим приоритетом может быть прервано диспетчером задач по истечению выделенного ей кванта времени или для передачи управления нитям с более высоким приоритетом. Выполнение нити со средним приоритетом может быть прервано аппаратным прерыванием.

Таблица

Принцип приоритезации в QPRT

Название системного приоритета	Описание системного приоритета	Вид диспетчеризации
Realtime	Ничто не может прервать исполнение нити.	Добровольная
Higher	Исполнение нити может быть прервано только аппаратным прерыванием.	Добровольная
Normal	Нить может быть прервана диспетчером задач для передачи управления нитям более высокого приоритета или по завершению кванта времени нити.	Принудительная, режим разделения времени

Единственной сущностью, исполняемой при любом приоритете, является диспетчер задач системы. При этом обеспечивается возможность принудительного переключения нитей с низшим приоритетом (режим разделения времени) и добровольная (нити сами отдают управление диспетчеру задач) всех остальных нитей с более высокими приоритетами. Каждой нити при ее создании присваивается базовый уровень системного приоритета. Такая приоритезация позволяет создать гибкую систему управления выполнением нитей, отвечающую разным задачам при построении встраиваемых систем на платформе QPRT.

В операционной системе QPRT в целях повышения быстродействия реализовано единое пространство виртуальной памяти для всех исполняемых нитей. Исполняемый код нитей, данные и ресурсы аппаратных устройств располагаются в одном адресном пространстве. Возможность выгрузки областей оперативной памяти на внешние запоминающие устройства (своппинг) отсутствует. Типы используемых областей памяти показаны на рис. 1.

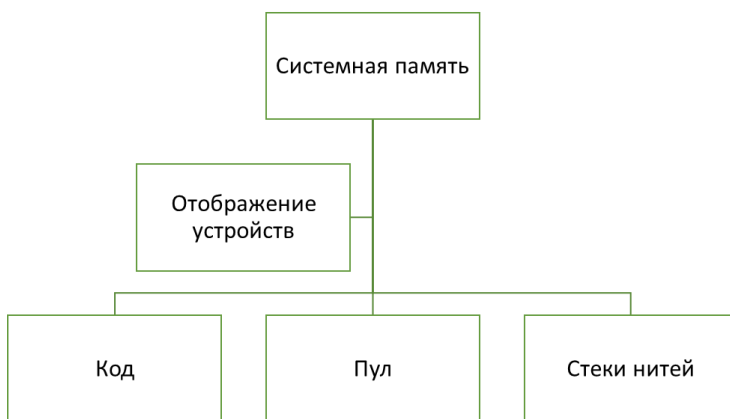


Рис. 1. Типы областей памяти в QPRT

Кроме приоритезации, в операционной системе QPRT для управления исполнением нитей, реализованы различные объекты синхронизации. Объекты синхронизации позволяют реализовать различные стратегии исполнения нитей в системе. Отличительной особенностью системы является присвоение приоритетов типам объектов синхронизации, как показано на рис. 2. Следует отметить, что в системе возможно ожидание объектов синхронизации с более низким

уровнем приоритета, чем текущий приоритет нити, что позволяет резко снизить количество тупиков (deadlocks). И наоборот, нить с низким уровнем приоритета не имеет права ожидать объект синхронизации с высоким уровнем приоритета. При этом любая нить может временно повышать свой приоритет с дальнейшим возвратом к базовому приоритету нити.

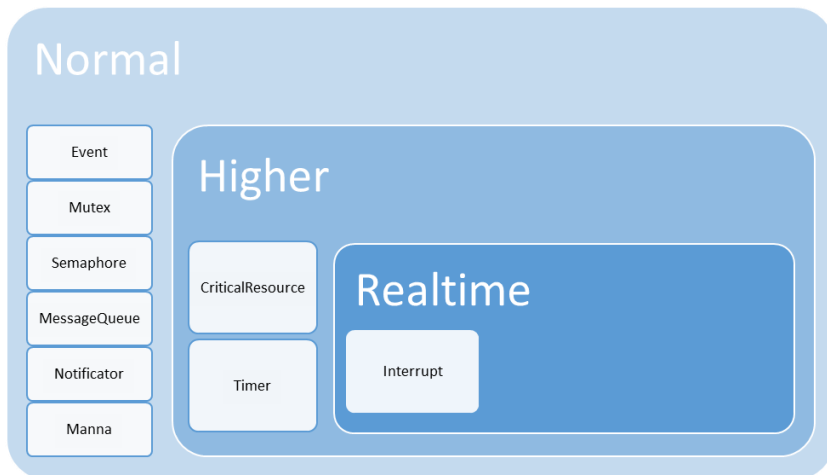


Рис. 2. Объекты синхронизации в QPRT.

Одной из отличительных особенностей операционной системы QPRT является представление любого аппаратного прерывания компьютера в виде объекта синхронизации высокого уровня приоритета. Это позволяет программисту создавать простой программный код для обработки прерываний. В сочетании с большим пулом нитей это позволяет создавать элегантные решения по взаимодействию с аппаратурой, как, например, это показано на листинге для обработчика прерываний таймера.

Листинг

Пример обработчика прерывания системного таймера.

```
/**
 * Нить системного времени
 */
DWORD
KrnTimeThread(
    _In_ PVOID pParam)
{
    while (TRUE)
```

```

{
// Ждём прерывание системного таймера
KrnWaitForSynchroObject (
    (PSSynchroObject)g_pSystemTimerInterrupt);
// Обновляем системное время
KrnUpdateCurrentTime();
// Возобновление системного таймера
HalResumeSystemTimerInterrupt(g_pSystemTimerInterrupt);
}
}

```

Наличие в системе поддержки высокоточного аппаратного таймера позволяет реализовать возможность задания ограничений по времени выполнения нитей для нитей с уровнями приоритетов Higher и Realtime с помощью механизмов операционной системы. При этом реакция системы на превышение длительности интервала выполнения нити может быть запрограммирована в исходном коде. Таким образом, реализуется стратегия как жесткого, так и мягкого реального времени, в зависимости от задач, решаемых информационно-управляющей системой. Принцип задания точек во времени иллюстрируется на рис. 3.

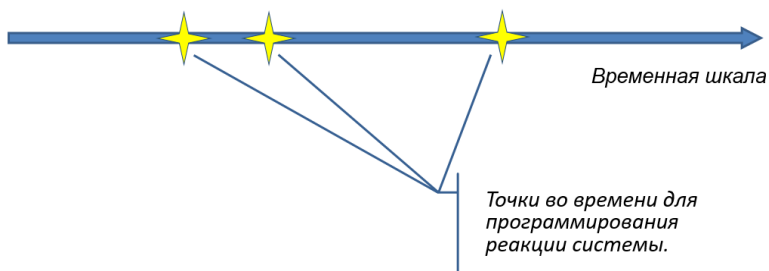


Рис. 3. Возможность задания точек во времени с помощью высокоточного таймера.

Идея применения высокоточного таймера была положена в основу управления процессом диспетчеризации задач в предшествующих публикациях автора, (см., например, [5]).

2. Применимость операционной системы QPRT

Необходимо также упомянуть об особенностях разработки программного обеспечения для операционной системы QPRT. Разработка программ для ОС выполняется на языке программирования С. Для разработки программ, исполняемых в среде операционной системы QPRT, подготовлен пакет разработчика, включая набор библиотек системных функций, интеграцию со средой разработки

программ, наличие системного отладчика через последовательный порт, идентичного отладчику в операционной системе QP ОС. Для системы разработан комплект документации. Специалисты предприятия готовы оказать консультационную помощь разработчикам программного обеспечения для операционной системы QPRT.

Следует отметить, что, так как операционная система QPRT разработана специалистами предприятия «с нуля», то возможны как доработки самой системы под нужды Заказчика, так и разработка новых программных модулей и законченных встраиваемых систем управления в соответствии с требованиями тактико-технического задания.

Заключение

Операционная система QPRT может быть применена как по прямому назначению для создания информационно-вычислительных систем, так и для обучения студентов высших учебных заведений основам системного программирования. Для этих целей имеется реализация прототипа системы как обычной прикладной программы в среде MS Windows. Программирование системы QPRT можно осуществлять в среде MS Visual Studio.

Список литературы

1. Операционная система QP ОС [Электронный ресурс]. – Режим доступа: <https://cryptosoft.ru/qpos>
2. QP ОС: защищенная российская операционная система [Электронный ресурс]. – Режим доступа: <https://securitymedia.org/articles/interview/qp-os-zashchishchennaya-rossiyskaya-operatsionnaya-sistema.html>
3. Егоров, В.Ю. Архитектура операционных систем: курс лекций. – Пенза : Изд-во ПГУ, 2019. – 142 с.
4. Гриценко, Ю. Б. Системы реального времени: учебное пособие / Ю. Б. Гриценко. – Томск : ТУСУР, 2017. – 253 С.
5. Егоров, В.Ю. Регионы времени как объекты операционной системы общего назначения / В.Ю. Егоров, Е.А. Матвеев // Информатика и ее применения. – Москва, 2008. – Т. 2. – № 4. – С. 74-84.