

УДК 65.011.56

А.И. Долгов, Д.А. Короченцев, В.В. Преснухин

О ВЕРОЯТНОСТНОМ МЕТОДЕ ОБРАБОТКИ НЕЧЕТКИХ ЧИСЕЛ

Как уже было показано [3], при сложении нечетких возможностных чисел в соответствии с принципом обобщения [1], используемого в теории нечетких множеств, возникает большая относительная ошибка представления степени размытости результата, ставящая под сомнение возможность применения таких чисел и методов их обработки в реальных системах принятия решений.

Это приводит к необходимости изыскания новых способов представления и обработки нечетких чисел, обеспечивающих приемлемую для практики точность вычислений. В качестве такого способа предлагается использовать нечеткие числа, функции, принадлежности которых описаны вероятностными законами и использовать для их обработки методы теории вероятностей.

В данном случае в качестве вероятностного закона наиболее целесообразен традиционный нормальный закон [2] и его модификации. Нечеткие числа, используемые в практических приложениях теории нечетких множеств, чаще всего имеют ограниченную область определения функции принадлежности (например: треугольные, трапециидальные и т.д.), поэтому по аналогии введём в рассмотрение нечеткие вероятностные числа с функцией принадлежности в виде ограниченного нормального закона, отличающегося от известного усечённого нормального закона [2] тем, что в результате преднамеренного линейного преобразования значения функции принадлежности в граничных точках области её определения приравняются к нулю.

Помимо этого в практических приложениях используются нечеткие числа, отличающиеся несимметричностью [1]. Это определяет необходимость введения в рассмотрение нечетких вероятностных чисел с функциями принадлежности в виде ассиметричного нормального закона и ограниченного ассиметричного нормального закона.

В качестве методов обработки рассматриваемых видов нечетких вероятностных чисел разработаны и программно реализованы процедуры сложения и умножения, основанные на дискретном методе аппроксимации результата операции, при этом как слагаемые, так и результаты могут описываться симметричными и несимметричными, неограниченными и ограниченными нечеткими вероятностными числами.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Борисов А.Н., Крумберг О.А., Федоров И.П.* Принятие решений на основе нечетких моделей. Примеры использования. – Рига: Зинатне, 1990. – 184 с.
2. *Вентцель Е. С.* Теория вероятностей: Учеб. для вузов. – 5-е изд. – М.: Высш. шк., 1998. – 576 с.
3. *Долгов А. И., Короченцев Д. А., Преснухин В. В.* Сравнительная оценка ошибки сложения возможностных и вероятностных нечетких чисел. / Материалы Международной научно-технической конференции. Т.1. – Таганрог: Изд-во ТТИ ЮФУ, 2007. – 330 с., с. 234-238.

УДК 621.32

Ю.И. Волчков

ОТЛАДОЧНЫЙ КОМПЛЕКС ДЛЯ ОСРВ scmRTOS

Операционная система – набор программного обеспечения (ПО), позволяющего максимально эффективно использовать ресурсы вычислительного устройства

в рамках поставленных задач. Многозадачные ОС позволяют разбить поток выполнения программы на несколько независимых, асинхронных по отношению друг к другу процессов и организовать взаимодействие между ними. Операционная система реального времени (ОСРВ, Real-Time Operating System - RTOS) – ОС, обладающая одним важным свойством: время реакции на события в такой ОС детерминированы, другими словами, имеется возможность оценить, через сколько времени с момента поступления события оно будет обработано.

ScmRTOS является операционной системой реального времени с приоритетной вытесняющей многозадачностью. ОС поддерживает до 32 процессов (включая системный процесс IdleProcess, т.е. до 31 пользовательских процесса), каждый из которых имеет уникальный приоритет. Все процессы статические, т.е. их количество определяется на этапе сборки проекта, и они не могут быть добавлены или удалены во время исполнения. Динамическое удаление процессов также не реализовано, так как в этом немного смысла – ПЗУ, используемое процессом, при этом не освобождается, а ОЗУ для последующего использования должно иметь возможность быть выделяемым/освобождаемым с помощью диспетчера памяти, который сам по себе весьма «толстая» вещь и, как правило, не используется в проектах на однокристальных МК.

Без специальных отладочных средств невозможно контролировать одновременную работу нескольких процессов, без потери «реального времени». Зачастую при разработке необходимо знать сколько времени тратит центральный процессор на обработку того или иного процесса, доходит ли очередь до процессов с низким приоритетом, и, таким образом, сделать выводы о корректности функционирования ОС и принять меры для оптимизации процессов.

Распространенное решение данной проблемы – подключение микроконтроллера через последовательный порт по интерфейсу RS-232 к персональному компьютеру, благодаря чему разработчик получает возможность составить ПО таким образом, чтобы устройство в контрольных точках алгоритма отправляло сообщения в персональный компьютер, и контролировать работу реализованного алгоритма.

Но здесь очень часто появляется другая проблема – отсутствие в микроконтроллере свободных последовательных интерфейсов. Так, например, у MSP430F169 всего два модуля USART. С этой проблемой столкнулся автор данной статьи. В разрабатываемом устройстве один модуль USART функционирует в режиме I2C другой – в режиме SPI. Поскольку на шину I2C теоретически можно подключить до 1024-х устройств, решение этой проблемы нашлось в виде создания специального отладочного устройства, которое бы принимало данные по шине I2C, и отправляло по интерфейсу USB в персональный компьютер. Так как проблема распространенная, решено было сделать отладочное устройство универсальным. То есть чтобы прием данных мог осуществляться по нескольким интерфейсам, и разработчик мог использовать наиболее удобный для конкретной разработки.

На рис. 1 представлена структурная схема разрабатываемого отладочного комплекса. Планируется реализовать возможность выбора протокола, по которому осуществляется прием данных. Отладочный комплекс будет содержать четыре интерфейсных модуля приема отладочной информации: I²C, SPI, UART и последовательный порт. В последнем случае используется 8 пинов микроконтроллера, на которые байт за байтом поступает отладочная информация. О готовности текущего байта сигнализирует сигнал-строб.

Очевидно, что следует реализовать возможность конфигурирования аппаратных модулей I²C, SPI и UART. По замыслу это будет выполняться одновремен-

но с выбором одного из протоколов из интерфейсной программы в PC, а конфигурирующие команды к отладочному устройству будут передаваться через интерфейс USB. В устройстве эти команды принимаются соответствующим модулем, дешифрируются в дешифраторе и поступают на исполнение в ядро.

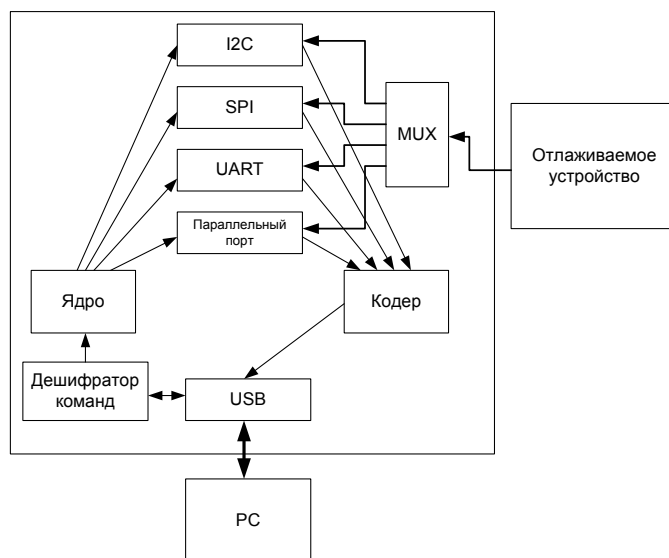


Рис. 1. Структурная схема отладочного комплекса

С отлаживаемого устройства отладочные данные через мультиплексор попадают на один из интерфейсных модулей, передаются на кодер, подготавливающий их к отправке на персональный компьютер, на который их посылает модуль USB.

Для удобства обработки данных на персональном компьютере, кодер упаковывает всю информацию в пакеты, ограниченные стартовым и стоповым условием, и в случае нетекстовых данных преобразует ее в формат ASCII. Кроме того, к пакету добавляется префикс, указывающий на тип данных (например, сообщение или сведения об использовании процессорного времени, внутреннее, сгенерированное в отладочном устройстве или внешнее, сгенерированное в отлаживаемом устройстве).

Рассмотрим интерфейсную часть комплекса (рис.2). Через USB-модуль происходит обмен данными с аппаратной частью комплекса. Декодер служит для выделения из принятого пакета информационной составляющей, и перенаправляет ее либо в модуль отображения сообщений, либо в анализатор использования процессорного времени. Данные о загрузке процессора отображаются графическим модулем в виде диаграммы или графика.

На данном этапе реализовано устройство, принимающее данные по I2C от отлаживаемого устройства, кодирующее его в удобный для передачи вид, и отправляющее по USB в интерфейсную часть, которая представляет собой лишь модуль отображения сообщений и частично управляющий интерфейс. Поскольку протокол I2C на практике оказался весьма капризным, решено было ввести возможность перезагрузки аппаратной части комплекса соответствующей командой с персонального компьютера. Для расширения универсальности адрес I2C модуля, также задается с PC.

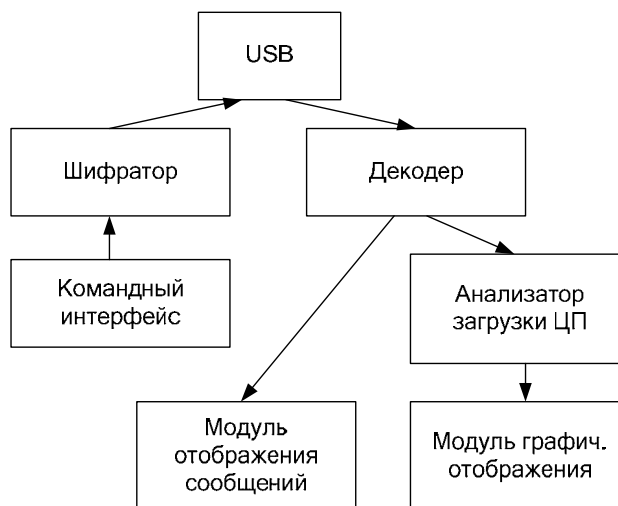


Рис.2. Структурная схема интерфейсной части

Важной особенностью ПО аппаратной части является постоянный контроль состояния шины I2C; реализованы обработчики всех возможных прерываний этого модуля во избежание потери контроля из-за внутренней ошибки. Кроме того, информация обо всех сбоях в аппаратной части также передается в персональный компьютер, но с префиксом внутреннего сообщения, и отображается в отдельном окне.

В scmRTOS не предусмотрено средств для контроля за использованием процессорного времени, поэтому придется модернизировать исходный код ядра. В файле OS_Kernel.cpp находятся функции, отвечающие за переключение задач, а именно:

- **void** TKernel::Sched()
- **void** TBaseProcess::Sleep(TTimeout timeout)
- **void** OS::WakeUpProcess(TBaseProcess& p)
- **void** OS::ForceWakeUpProcess(TBaseProcess& p)

Для нас представляет интерес первая функция, которая и осуществляет переключение задач. В ее код следует вставить функцию отправки сообщения, содержащего информацию о процессе, который в данный момент получит управление. Для того чтобы отправка отладочной информации не осуществлялась в серийных образцах, необходимо использовать условную компиляцию:

```

#define debug
...
#ifdef debug
    Send_dbg_msg("message");
#endif
  
```

Доработка операционной системы на этом не заканчиваются. Планируется организовать непрерывный контроль за всеми механизмами, включая средства межпроцессного взаимодействия, такие как семафоры, флаги событий, каналы и сообщения, чтобы была возможность отследить, к примеру, какой процесс захва-

тил конкретную переменную или в какой момент наступило какое-либо конкретное событие.

Итак, разрабатываемый комплекс будет наделен функциями, существенно облегчающими процесс отладки устройств, как на базе операционных систем, так и разрабатываемых по принципу машин состояний. Кроме того, он обладает значительной универсальностью, поскольку может принимать информацию от отлаживаемого устройства по четырем протоколам, каждый из которых легко настраивается из интерфейсной части отладочного комплекса. При использовании операционной системы scmRTOS удобно отслеживать все процессы и механизмы, происходящие в отлаживаемом устройстве, причем без потери «реального времени».

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Операционная система реального времени для однокристальных микроконтроллеров. – Новосибирск 2003-2006.

УДК 65.011.56

А.И. Долгов, А.Ф. Мартыненко, В.В. Преснухин

О ПРОГРАММНОЙ РЕАЛИЗАЦИИ КОЭФФИЦИЕНТНЫХ ШТАБНЫХ МЕТОДИК

Штабные коэффициентные методики, основанные на получении интегрального (выходного) показателя суммированием оцениваемых частных (входных) показателей, умножаемых на соответствующие весовые коэффициенты, широко используются в организациях и административных органах, занимающихся управленческой деятельностью, так как обеспечивают рациональное сочетание оперативности принимаемых решений с возможностями их количественного обоснования.

При современной технологии программирование каждой создаваемой штабной коэффициентной методики представляет собой уникальный процесс, что приводит к существенным трудозатратам и весьма усложняет совершенствование методик. Усилия по существенной коррекции существующих программ, особенно при отсутствии хорошей документации, часто оказываются соизмеримыми с усилиями по разработке новой программы. Кроме того, недостатком существующих штабных коэффициентных методик является отсутствие возможности учёта задаваемых логических условий получения результата. Так, например, получение интегральной оценки профессиональной подготовки специалиста, равной 5 (отлично) по двум отдельным частным оценкам практических навыков и теоретических знаний при условии, что интегральная оценка 5 выставляется при оценке практических навыков не ниже 5 и оценке теоретических знаний не ниже 4, с помощью традиционного способа оценки (с использованием вполне определённых коэффициентов) неосуществимо.

Выходом из данной ситуации является создание программной оболочки для разработки и реализации на ПЭВМ коэффициентных штабных методик в режиме, требующем лишь заполнения общепонятных экранных форм, а также обеспечение возможностей изменения величины весового коэффициента в зависимости от значения умножаемого показателя.

Разработанный с участием авторов вариант программной оболочки (свидетельство о государственной регистрации программы для ЭВМ № 2008610183 от 9 января 2008 г.) может быть использован для разработки и реализации самых разнообразных задач (технических, технологических, организационных и др.) широ-