

Для упрощения процедуры расчетов и сокращения ее сроков можно использовать программный продукт, реализованный в MS Excel, интерфейс, которого представлен на рис. 4.

Введите количество эмитированных акций, шт.	15220
Введите количество акций для оценки, шт.	<b>2599</b>
Размер пакета, %	17,1
Скидка на неконтрольный пакет, %	<b>30</b>
Премия за контроль, %	<b>0</b>

Рис. 4. Пример расчета премии скидок и премий

При проведении процедуры согласования результатов оценки бизнеса, полученных тремя различными подходами, необходимо каждому из результатов присвоить определенный вес значимости. При этом необходимо учитывать следующие факторы: влияние на достоверность величины рыночной стоимости исходных данных и методики расчетов, полнота исходной информации, ее достоверность, количество и значимость субъективных показателей и предположений. Таким образом, для решения задачи согласования полученных результатов можно применять программное обеспечение, реализующее метод анализа иерархий [4].

Предложенные информационные модели и специальное программное обеспечение позволяют формализовать процессы принятия решений при оценке бизнеса и дают возможность повысить объективность и качество оценки стоимости компании, что важно при выработке стратегии, увеличивающей стоимость компании.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Есипов В.Е., Маховикова Г.А., Терехова В.В.* Оценка бизнеса. 2-ое изд. – СПб.: Питер, 2006. – 464 с.
2. *Козырь Ю.В.* Стоимость компании: оценка и управленческие решения. – М.: Издательство «Альфа-пресс», 2004. – 200 с.
3. *Черемных С.В., Семенов И.О., Ручкин В.С.* Структурный анализ систем: IDEF-технологии. – М.: Финансы и статистика, 2001.
4. *Черноруцкий И. Г.* Методы принятия решений. – СПб.: БХВ-Петербург, 2005. – 416 с.

УДК 621.3

**Е.С. Бовин**

#### **ПРИМЕНЕНИЕ СОБЫТИЙНОЙ МОДЕЛИ И ВИЗУАЛЬНОГО ПРОГРАММИРОВАНИЯ ПРИ ПРОЕКТИРОВАНИИ АППАРАТНЫХ СРЕДСТВ НА БАЗЕ МИКРОКОНТРОЛЛЕРОВ**

Поиск новых подходов в программировании обуславливается постоянной необходимостью повышения производительности труда программиста и качества продукции в условиях ограниченных ресурсов и требований по снижению себестоимости. Одним из путей решения этой нетривиальной задачи является создание инструментария, облегчающего разработку комплексных программных проектов. Первичное облегчение процесса проектирования достигается за счет более удобных средств разложения задачи на отдельные составляющие. Второй шаг – поиск

и применение эффективных методов защиты от ошибок в программах, их автоматизированной проверки, ужесточения требований к программному коду. И, наконец, обеспечение возможности составить алгоритм программы на достаточно емком, простом, доступном языке программирования, не требующем продолжительной специальной подготовки, – это третий необходимый компонент для решения поставленной задачи.

Наиболее удобные и эффективные в этом смысле идеи – объектное и событийное программирование, визуальный подход к программированию.

Алгоритм – это последовательность команд микропроцессору устройства. Объектный подход к программированию основывается на понятии объекта, инкапсулирующего все алгоритмы для работы с ним. Объект — это совокупность переменных состояния и связанных с ними методов (операций), определяющих как объект взаимодействует с окружающим миром. Таким образом, программный объект – это описание любого физического объекта, который:

- имеет какое-то состояние (или находится в каком-то состоянии);
- имеет определенное поведение.

Возможность управлять состояниями объекта посредством вызова методов и в итоге определять поведение объекта, совокупность таких методов – интерфейс объекта.

Применяя объектную модель программирования удобно создавать эффективные доступные для каждой серии микроконтроллеров библиотеки из наборов объектов. Программный код таких библиотек создается опытными специалистами высокого класса, что обеспечивает практически полное отсутствие ошибок. Последующее применение этих объектов реализуется уже на значительно более простом уровне через их интерфейсы. В самом распространенном случае, применение объекта даже не требует понимания принципов сложного внутреннего устройства самого объекта! Реализация конечного алгоритма работы программы с использованием объектов приводит к существенному снижению ошибок, так как необходимость вновь реализовывать сложные низкоуровневые процедуры работы с различными блоками микроконтроллеров (и периферийных устройств) отсутствует!

Событийный подход к программированию включает использование объектов, способных реагировать на события, происходящие в системе. Основная специфика подхода – инициатива принадлежит не самой программе, а внешнему миру, на события которого программа реагирует. Обработка различных событий описывается набором последовательностей команд, выполняемых при наступлении внешних по отношению к ней событий, время и последовательность наступления которых, как правило, заранее не известны и могут отличаться.

Программа, для написания которой событийного программирования не требуется, обычно занимает всё предоставленное ей время процессора и выполняется линейно. В современной технике выполнение практических всех типов подобных линейных задач реализуется на основе специализированных микросхем, без использования микропроцессорных устройств. Но самый частый случай – программа большую часть времени находится в состоянии ожидания внешних воздействий, и число различных событий велико. Естественно, эффективнее написать её с использованием событийного программирования. Но для получения и распределения событий необходим менеджер событий, и в случае программирования микроконтроллеров, эту сложную и целиком аппаратно-зависимую часть всего программного кода потребуется написать самостоятельно! Как же быть?

Микроконтроллеры XMEGA фирмы ATMEL – это новое, более совершенное поколение уже хорошо знакомых, популярных и признанных AVR-микроконтроллеров. Они имеют встроенную Flash-память объемом от 16 до 1024

килобайт, производительность до 32 MIPS и рабочее напряжение от 1,8 до 3,6 вольт. Микроконтроллеры XMEGA программно-совместимы с серийно выпускаемыми семействами tinyAVR и megaAVR.

Два новых периферийных блока контроллеров XMEGA – система событий и многоуровневый программируемый контроллер прерываний.

Система событий предназначена для разгрузки центрального процессора. Данная система построена таким образом, чтобы обеспечивать гибкость в предоставлении прав каждому из взаимодействующих с ней периферийных модулей:

- назначение внешних воздействий, создающих событие (перепад напряжения на определенном выводе микроконтроллера, работа с таймерами, работа с ЦАП\АЦП, и т. п.);
- выбор интерпретации поступающих событий (изменить режим работы таймера, выставить выходной сигнал на выводе микроконтроллера, и т. п.).

Применение системы событийного программирования повышает надежность обработки данных и принятие решений, в силу более точно прогнозируемого поведения функций критичных ко времени или стабильности выполнения, что повышает стабильность работы всего продукта.

Дополнительно к событийной системе впервые в XMEGA реализована 4-уровневая система прерываний с диспетчеризацией запросов. Кроме основных системных прерываний, возможно так же назначить дополнительно прерывания высокого, среднего и низкого уровней. Для обеспечения надежности «срабатывания» всех прерываний с самым низким приоритетом введена специальная процедура диспетчеризации. Применяемый циклический алгоритм диспетчеризации («карусельный») активизирует все процессы в строго фиксированном циклическом порядке.

Таким образом, в семействе контроллеров XMEGA фирмы ATMEL возможна реализация совместно объектной и событийной модели программирования.

Визуализация – это процесс графического отображения сложных процессов (в данном случае построения алгоритма программы) на экране компьютера в виде графических примитивов (графических фигур). Для визуализации интерфейсов объектов существует целый ряд специально разработанных элементов интерфейса – визуальных компонент, позволяющих отображать различную информацию и осуществлять управление.

Определяющими элементами процесса визуализации являются:

1) визуализируемая модель – модель, которая подвергается отображению с целью возможности изменения ее структуры или ее параметров (либо параметров ее отдельных частей);

2) окно инструментов – окно, содержащее набор элементов, из которых строится визуальная модель. Обычно элементы разделяются по их назначению на отдельные группы, размещающиеся на отдельных закладках окна инструментов;

3) окно свойств – окно, в котором отображаются параметры (свойства) выбранного элемента визуальной модели. Термин "свойство" пришел из объектно-ориентированного программирования и обозначает параметр объекта.

Визуализированная форма записи программ обладает максимальной гибкостью и переносимостью. Преобразовать табличную форму записи в любой известный язык программирования является "делом техники". Под графическими примитивами, которыми оперирует программист, скрывается полностью объектно-ориентированный код – объекты, их взаимодействие, система событий, и т. д., обеспечивая использование в проекте всех лучших качеств каждого из подходов программирования.

Применение визуального программирования сокращает сроки выхода на рынок конечного продукта, и одновременно разработка его не предъявляет к программисту высоких требований по профессиональной подготовке. Иначе говоря, составить сложную достаточно программу работы аппаратуры на базе микроконтроллера сможет инженер-разработчик, совершенно не знакомый с языками программирования!

В заключение хотелось бы отметить, что новая серия 8-битных контроллеров XMEGA от ATMEL предоставляет широкие возможности для создания качественно новых радиоэлектронных устройств, а также модернизации старого оборудования с минимальными затратами. Однако визуальный подход в программировании микроконтроллеров находится лишь в начале своего развития, и в настоящее время все еще сохраняется необходимость участия профессиональных программистов в проектировании систем сложности выше низкой на основе микроконтроллеров.

УДК 681.55

**Д.Ю.Фогель**

### **РЕАЛИЗАЦИЯ БЕСПРОВОДНОЙ СИСТЕМЫ ИЗМЕРЕНИЯ ТЕМПЕРАТУРЫ ТРЕХЗОННОЙ ПРОХОДНОЙ ПЕЧИ**

Сегодня задача создания распределенных систем сбора данных и мониторинга как никогда актуальна в самых различных прикладных областях. Например, при промышленном производстве некоторых изделий точная информация о температуре материала позволяет задать оптимальные условия его производства, а также оптимизировать длительность процесса изготовления, однако использование традиционных проводных соединений, при создании таких систем, не всегда эффективно из-за высокой стоимости монтажных и пусконаладочных работ, а также технического обслуживания. Кроме того, в некоторых ситуациях вообще невозможна прокладка кабелей по технологическим или организационным причинам [6]. Поэтому беспроводная передача данных выглядит весьма привлекательно для решения таких задач.

Цель данной статьи – рассмотреть возможность реализации беспроводной системы измерения температуры на примере трехзонной проходной печи обжига керамики.

#### **Описание характеристик объекта наблюдения**

Проходные печи применяют для нагрева металлических заготовок перед горячей обработкой давлением и при термообработке изделий и деталей, для обжига керамических и эмалированных металлических изделий и т.д. [1].

Исходя из описания трехзонной проходной печи обжига керамики [4], можно выделить следующие особенности объекта наблюдения, необходимые при реализации системы измерения температуры:

- 1) температура внутри проходной печи составляет 600 – 1200 °С;
- 2) тепловой режим трехзонной проходной печи - постоянный по времени и переменный по длине печи – зоны: нагрева, выдержки и охлаждения;
- 3) рабочее пространство проходных печей обжига керамики сравнительно невелико в поперечном сечении (ширина 1 – 6 м, высота 1 – 2 м, длина до 10 м);
- 4) средняя скорость передвижения тележек/пода внутри проходной печи составляет 2 – 4 м/ч;
- 5) наличие высоковольтных линий электропередачи.

#### **Разработка структурной схемы системы**