

УДК 621.3+62-5+004.413.2

Разработка системы управления электропривода на основе метода модельно-ориентированного программирования

И.С. Полющенко

Сведения об авторе

Полющенко Игорь Сергеевич — кандидат технических наук, старший преподаватель кафедры электромеханических систем Смоленского филиала НИУ «МЭИ», e-mail: polyushenckov.igor@yandex.ru

Описаны этапы и приведены результаты проектирования микропроцессорной системы управления электропривода с вентильным электрическим двигателем с применением метода модельно-ориентированного программирования, основанного на использовании системы компьютерной математики Matlab, ее компонентов и пакетов расширений в качестве средства разработки и отладки программного обеспечения систем управления техническими системами. Работа с указанным методом значительно упрощает и ускоряет процесс создания программного обеспечения, так как оно проектируется в удобной и наглядной графической форме, аналогичной обычным математическим моделям Matlab/Simulink. Показано, что средства модельно-ориентированного программирования позволяют применять аппаратные и вычислительные ресурсы микроконтроллера, а также математические средства системы Matlab для решения конкретных задач управления. Использование средств модельно-ориентированного программирования показано на примере проектирования функционально завершенной системы управления электропривода. При разработке электропривода решен комплекс задач управления вентильным электрическим двигателем, регулирования координат в режимах стабилизации скорости вращения, слежения и позиционирования, доступа к параметрам и настройкам системы управления, интерактивного управления и связи с системой управления верхнего уровня по цифровому интерфейсу. Средства модельно-ориентированного программирования позволили представить программное обеспечение микропроцессорной системы управления электропривода в графической форме, указать связи между модулями и функциями программного обеспечения и порядок их взаимодействия, описать последовательность выполнения алгоритма управления электроприводом и взаимодействия его системы управления с периферийными устройствами. Доступ системы управления к коммуникационной сети по цифровому интерфейсу и протокол обмена данными между электроприводом и системой управления верхнего уровня также реализованы с помощью метода модельно-ориентированного программирования. Для интерактивного управления электроприводом разработано специальное приложение для персонального компьютера. Показано, что программное обеспечение, разработанное с применением метода и средств модельно-ориентированного программирования, по функциональности не уступает программному обеспечению, созданному с применением обычных средств программирования. Данный электропривод может быть использован в следящих электромеханических системах, требующих динамичного регулирования координат. Результаты разработки программного обеспечения с применением метода модельно-ориентированного программирования могут быть востребованы в качестве учебно-методических материалов.

Ключевые слова: вентильный электропривод, микропроцессорное управление, регулирование координат, система управления электропривода.

Development of an electric drive control system by using the model-oriented programming technique

I.S. Polyushchenkov

Information about author

Polyushchenkov Igor S. — Ph.D. (Techn.), senior lecturer of electromechanical systems dept., Smolensk branch of MPEI, e-mail: polyushenckov.igor@yandex.ru

This article describes the stages of and presents the results from designing a microprocessor system for control of an electric drive with a converter-fed electric motor by applying the model-oriented programming technique. The technique is based on using the Matlab computer mathematics system, its components, and extension packages as a means of elaborating and debugging software for control of technical systems. Application of the above-mentioned techniques makes it possible to greatly simplify and accelerate the software development process, because this software is designed in convenient and clear graphic form similar to the usual mathematical models of Matlab/Simulink. It is shown that the tools of model-oriented programming make it possible to use both hardware and computing resources of a microcontroller, as well as the Matlab system's mathematical tools to address specific control tasks. The use of model-oriented programming technique tools is shown by the example of designing a functionally complete electric drive control system. In designing the electric drive, a comprehensive set of tasks has been solved, the list of which includes control of a converter-fed electric motor, control of

coordinates in the speed stabilization, tracking, and positioning modes, access to the control system parameters and settings, interactive control, and communication with the higher-level control system via a digital interface. The use of tools offered by model-oriented programming technique made it possible to present the software of the microprocessor electric drive control system in graphic form, to specify connections between the modules and software functions and their interaction, and to describe the sequence of executing the electric drive control algorithm and the interaction of the drive control system with peripherals. Access of the control system to the communications network via the digital interface and the data exchange protocol between the electric drive and the higher-level control system are also implemented using the model-oriented programming technique. To implement interactive control of the electric drive, a special application for the personal computer has been developed. It is shown that the software developed using the model-oriented programming technique and its tools has functionality at least as good as that developed using conventional programming tools. The developed electric drive may be used in electromechanical servo systems requiring dynamic control of the coordinates. The software development results obtained from using the model-oriented programming technique can also be used as teaching and learning aids.

Key words: converter-fed electric drive, microprocessor control, coordinate control, mathematical modeling, electric drive control system.

В настоящее время микропроцессорная техника имеет широчайшее распространение в системах управления техническими системами. Это объясняется ее значительными функциональными возможностями и высокой производительностью, удобством и гибкостью реализации разнообразных алгоритмов управления на ее основе. Опыт проектирования микропроцессорных систем управления и совершенствование программных и технических средств их разработки постоянно переосмысливается. В результате этого появляются новые методы и технологии программирования, которые в свою очередь служат для усовершенствования средств разработки микропроцессорных систем управления.

При проектировании системы управления технической системы, как правило, требуется описать ее в виде модели, показав ее иерархию, последовательность и условия взаимодействия ее составных частей между собой. На этапе эскизного проектирования для проверки работоспособности алгоритмов управления, исследовании динамических свойств и режимов работы системы при использовании современных средств компьютерного математического и имитационного моделирования указанное требование может быть легко выполнено. Однако это не всегда возможно сделать в удобной и наглядной форме при использовании средств разработки прикладного программного обеспечения микропроцессорных систем управления (например, Keil, IAR). Кроме того, определенные ограничения на производительность труда разработчика при создании микропроцессорной системы управления накладывает уровень знаний специализированных языков программирования и наличие навыков их применения.

В статье рассматривается подход к проектированию программного обеспечения для системы управления электропривода, при котором разработка и отладка программного обеспечения проводятся с непосредственным использованием методов и программных средств компьютерного моделирования.

Рассматриваемый подход, получивший название модельно-ориентированного программирования [1], позволяет совместить методы и средства компьютерного моделирования и проектирования микропроцессорных систем. Указанный способ проектирования программного обеспечения применяется при разра-

ботке систем управления, обработки сигналов, связи и других технических систем.

Суть процесса разработки программного обеспечения при таком подходе состоит в следующем. На первом этапе в системе компьютерной математики Matlab графическим образом составляется модель системы управления с виртуальной моделью объекта управления. Подобные модели широко используются для отладки алгоритмов управления и анализа динамических свойств технических систем. В качестве виртуального объекта управления может быть, например, использована модель электрического двигателя из библиотеки SimPowerSystems/Machines и/или модель полупроводникового преобразователя. Далее компьютерную модель требуется адаптировать к структуре выбранного микроконтроллера, получив, так называемую исполняемую модель, в которой учитывается специфика аппаратных средств микроконтроллера, который планируется использовать в реальной системе управления. На этом же этапе следует составить временную диаграмму выполнения алгоритма управления, назначить приоритеты и указать условия выполнения его составных частей, распределить программные и аппаратные ресурсы микроконтроллера, а также указать линии для подключения микроконтроллера к аппаратной периферии.

Все перечисленное выше должно быть отражено в исполняемой модели средствами модельно-ориентированного программирования. Модель же объекта управления (в нашем случае — модель электрического двигателя и транзисторного преобразователя) исключается из исполняемой модели, которая взаимодействует с физическим объектом с помощью аппаратной периферии микроконтроллера, например, по линиям, предназначенным для подключения датчиков или транзисторных ключей.

При разработке компьютерной модели и исполняемой модели на ее основе удобно использовать следующие компоненты Matlab:

- математические средства библиотек Simulink [2];
- средства для работы с периферией и памятью микроконтроллера (собственно, средства модельно-ориентированного программирования);
- средства пакета расширения Stateflow для реализации логических последовательностей при управлении;

S -функции на встроенном языке программирования Matlab [3];

функции на языке программирования выбранного микроконтроллера, включенные в состав исполняемой модели.

В качестве средства модельно-ориентированного программирования, может быть использован пакет Waijung Blockset [4], который является внешней надстройкой для Matlab и имеет обширный набор функциональных блоков, выполняющих типовые вычислительные функции. Эти блоки позволяют использовать аппаратные ресурсы микроконтроллера и работать с его памятью, осуществлять связь с периферийными устройствами технических систем — выполнять функции ввода-вывода цифровых и аналоговых сигналов, захвата и генерирования последовательностей импульсов, передачи, отслеживания и захвата информационных пакетов.

Выбор указанного средства программирования обусловлен, в частности, тем, что этот пакет предназначен для программирования микроконтроллеров семейства STM32, а в разработанном электроприводе применен микроконтроллер типа STM32F407 как обладающий высокой производительностью и обширным набором встроенных аппаратных модулей. Он может быть эффективно использован в управлении различными техническими системами, в том числе и электроприводами.

Чтобы из исполняемой модели, представленной в графической форме, получить программу на языке Си, и создать, таким образом, приложение для дальнейшей загрузки в выбранный микроконтроллер, могут быть использованы компиляторы GNU ARM, Embedded Coder и LCC-Win32, интегрированные в Matlab. Программный код, сгенерированный из исполняемой модели, аналогичен коду, который генерируется при использовании обычных средств разработки.

Основные принципы модельно-ориентированного программирования рассмотрены далее на примере

разработки программного обеспечения для системы управления электропривода с вентильным двигателем (ВД) [5]. Функциональная схема этого электропривода представлена на рис. 1.

Система регулирования имеет внутренний контур регулирования скорости вращения электрического двигателя с регулятором скорости РС и внешний контур регулирования угла поворота его ротора с регулятором положения РП. Внешний контур замыкается при работе электропривода в режиме регулирования положения. В режиме стабилизации скорости вращения ротора используется только внутренний контур. Регулирование тока осуществляется методом отсечки с помощью узла токоограничения (ТО).

В режиме позиционирования задающее воздействие на регулятор положения вычисляется задатчиком перемещения, который формирует S -образную траекторию движения с ограничением скорости движения и ускорения. Датчик положения ротора (ДПР) используется для замыкания контуров регулирования. Вычислитель координат (ВК) предназначен для расчета скорости вращения ротора двигателя и угла его поворота. Вентильный двигатель, имеющий возбуждение от постоянных магнитов, питается от силового полупроводникового преобразователя (СПП), который осуществляет широтно-импульсную модуляцию (ШИМ) напряжения. Блок коммутации (БК) и датчики Холла (Н_А, Н_В, Н_С) реализуют позиционную обратную связь в функции углового положения ротора для управления силовым полупроводниковым преобразователем таким образом, чтобы обеспечить заданное направление вращающего момента электрического двигателя. Задающие воздействия для электропривода в режиме слежения и в режиме стабилизации скорости вращения могут задаваться аналоговыми или цифровыми задатчиками, логической схемой управления (например, по сигналам реле времени и путевых датчиков) или системой управления верхнего уровня.

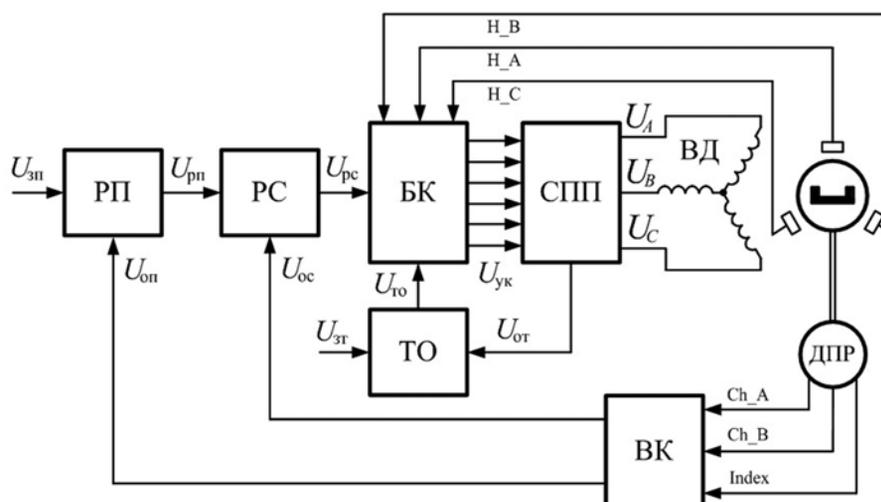


Рис. 1. Функциональная схема вентильного электропривода

В целом, системой управления рассматриваемого вентильного электропривода должны решаться следующие задачи.

Синхронизация управляющих воздействий на ВД с положением его ротора.

Ограничение тока статорной цепи, стабилизация скорости вращения ротора и регулирование угла его поворота в режимах слежения и позиционирования.

Обмен данными с системой управления верхнего уровня, интерактивное управление и доступ к параметрам и настройкам системы.

Взаимодействие с периферийными датчиками и реализация логической схемы управления.

Блок-схема реализуемого алгоритма управления ВД, которая показывает распределение программных и аппаратных ресурсов микроконтроллера, последо-

вательность взаимодействия программных модулей, а также последовательность взаимодействия системы управления электропривода с объектом управления и системой управления верхнего уровня изображена на рис. 2.

Как уже было отмечено выше, для выполнения указанного алгоритма использован высокопроизводительный микроконтроллер STM32F407, имеющий аппаратную периферию, которая может быть использована при управлении электроприводом.

Таймер, работающий в режиме захвата, для определения скорости вращения ротора путем измерения интервала времени между импульсами на одном из каналов энкодера.

Таймер, работающий в режиме счета импульсов от энкодера, для определения угла поворота ротора.

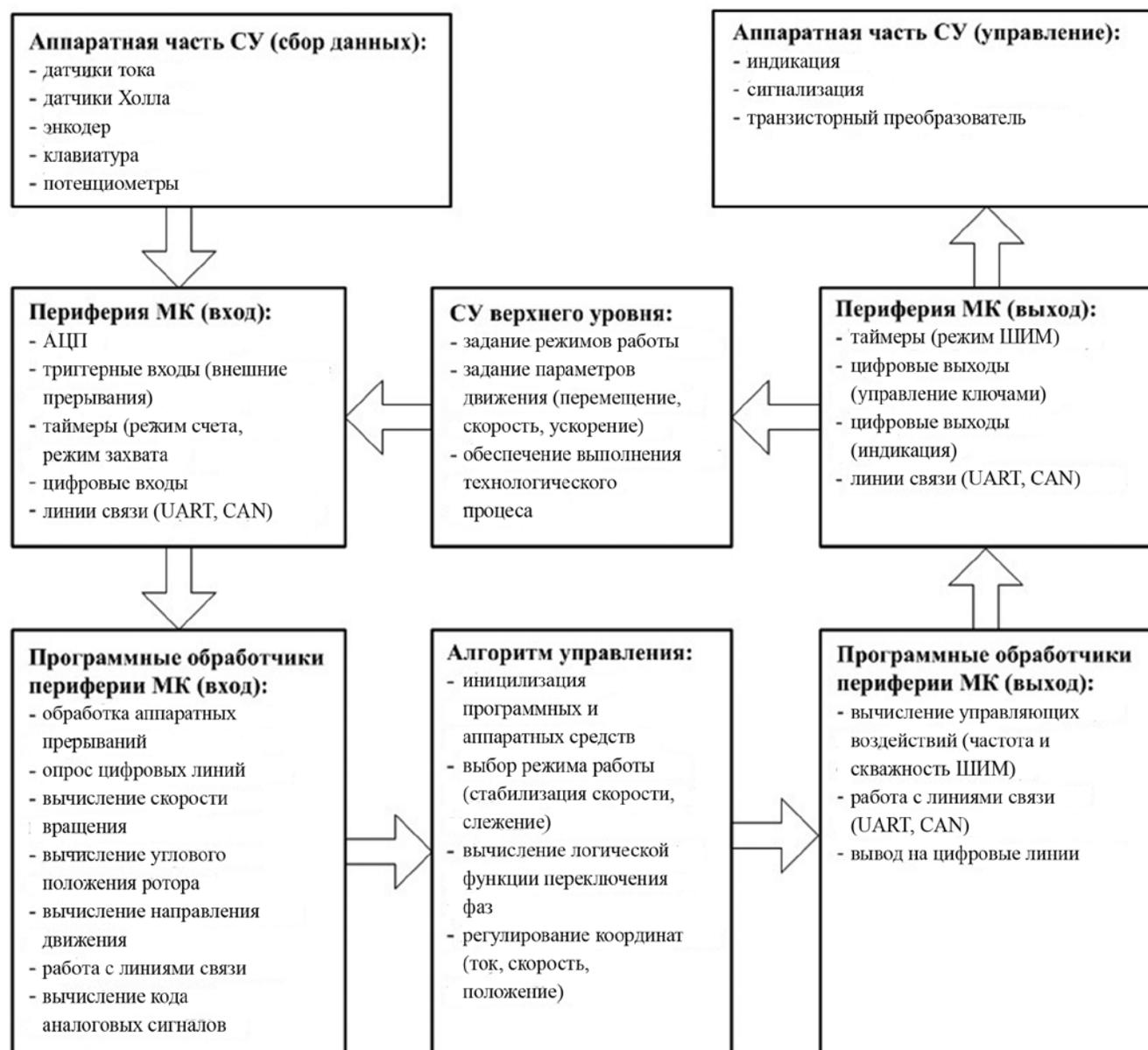


Рис. 2. Блок-схема алгоритма управления вентильным двигателем

Триггерные входы и обработчики внешних прерываний для работы с датчиками Холла при переключении фаз, а также при взаимодействии с периферийными датчиками.

Цифровые линии для реализации клавиатуры и сигнализации.

Модуль последовательного порта UART для обмена данными с персональным компьютером (ПК) в интерактивном режиме.

Таймеры для тактирования выполнения программных модулей (тактирование задатчика перемещения, опрос клавиатуры, управление обменом данными с ПК).

Модули АЦП для измерения аналоговых сигналов — задания скорости и перемещения, а также сигнала обратной связи по току.

Таймер, работающий в режиме генератора импульсов, для управления ключами силового преобразователя.

Вычислительные операции алгоритма управления могут быть синхронизированы с системным таймером, тактироваться системными прерываниями или выполняться при возникновении внешних или системных прерываний, а также при изменении программных флагов по мере выполнению каких-либо действий, что должно быть отражено в исполняемой модели. Модули программного обеспечения, синхронизированные с системным таймером, образуют цикл управления и регулирования координат электропривода, который имеет фиксированный интервал повторения, равный периоду дискретизации системы (Sample time).

Классификация основных операций алгоритма управления в зависимости от типа синхронизации приведена в таблице.

На рисунке 3 показана схема исполняемой модели электропривода на базе микроконтроллера STM32F407. Модель имеет несколько функционально завершенных подсистем и отдельных блоков для связи с аппаратной периферией системы управления. Ограниченный объем статьи не позволяет полностью показать устройство и состав исполняемой модели системы управления электропривода. Поэтому далее приведено описание некоторых из подсистем исполняемой модели, которые позволяют продемонстрировать возможность использования различных компонентов Matlab.

Блоки Regular ADC, Regular ADC1 и PWM Capture пакета Waijung используются для взаимодействия микроконтроллера с аппаратной частью системы управления — соответственно для измерения сигнала задания угла поворота (или скорости вращения) ротора, измерения сигнала обратной связи по току и измерения скорости вращения ротора вентильного двигателя. Блок Target Setup предназначен для настройки конфигурации целевого микроконтроллера и средств генерирования программного кода. Блок UART Setup предназначен для настройки модуля последовательного порта микроконтроллера.

Элементы и блоки, используемые в модели, могут быть объединены в подсистемы различных видов — виртуальные, монолитные, триггерные, стробируемые или управляемые S-функциями, имеющиеся в библиотеке Simulink/Ports&Subsystems. Применительно к разработке программного обеспечения электропривода с применением метода модельно-ориентированного программирования это позволяет выделить вычислительные потоки, а также установить последовательности и условия выполнения задач управления согласно таблице и рис. 2.

Подсистема Logic (рис. 4) используется при вычислении логической функции для последовательного переключения фаз вентильного двигателя в зависимости от углового положения его ротора. Указанная функция вычисляется по сигналам от датчиков Холла, опрос которых происходит по прерыванию при изменении логического уровня на линиях микроконтроллера, подключенных к датчикам. Также указанная подсистема выполняет функции распределения токов по фазам электрического двигателя и задания направления вращения ротора ВД.

Блок IRQ Custom Code позволяет задействовать в исполняемой модели обработчик внешних прерываний микроконтроллера, которые генерируются при перепаде логических уровней сигналов на любой из трех линий, к которым подключены датчики Холла. Подсистемы Hall_Interrupt и Event_Generator обеспечивают изменение программных флагов для вызова других подсистем. Подсистема Detector осуществляет опрос линий с датчиками Холла при возникновении внешних прерываний и вычисление логической переключающей функции. Подсистема Switcher позволяет задавать

Классификация задач управления по последовательности выполнения

Задачи системы управления		
Синхронизация с системным таймером	Синхронизация с системными флагами	Асинхронное выполнение
опрос аналоговых и цифровых входов; преобразование АЦП; вывод на цифровые линии (индикация, ШИМ); регулирование скорости, угла поворота и ограничение тока; сбор и обработка данных от датчиков	формирование и передача выходного пакета данных; опрос клавиатуры; реализация временных зависимостей при отработке логической схемы управления; вычисление траектории перемещения с заданными величинами скорости и ускорения	опрос датчиков Холла и вычисление логической функции для переключения фаз; прием и анализ пакета данных; доступ к параметрам и настройкам системы управления; обработка событий при работе с пугтевыми датчиками

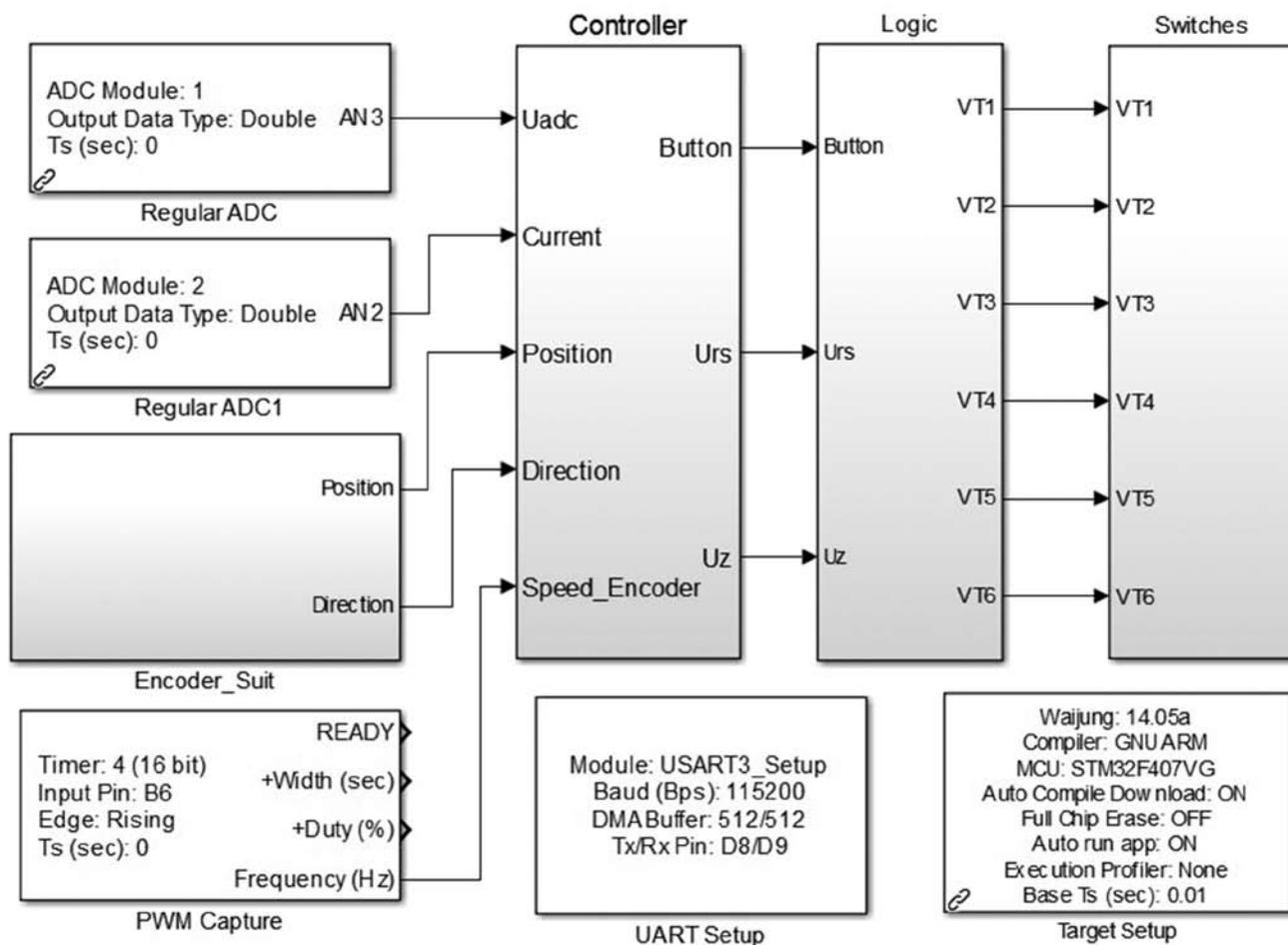


Рис. 3. Исполняемая модель вентильного электропривода

последовательность переключения фаз в зависимости от заданного направления вращения ротора (вход Uz), а также осуществляет разрешение и запрет работы преобразователя (вход *Button*).

Подсистема *PWM_Switch* используется для задания скважности широтно-импульсной модуляции (ШИМ) напряжения по сигналу задания скорости вращения ротора (вход *Urs*).

Устройство подсистемы *Encoder_Suit*, предназначенной для вычисления угла поворота ротора вентильного двигателя по сигналам, получаемым от энкодера, показано на рис. 5, а. С помощью блока *Encoder Read* таймер 1 задействуется в режиме подсчета импульсов с каналов А и В инкрементального энкодера, которые подключены к линиям PE9 и PE11 микроконтроллера.

Указанный блок вычисляет направление вращения (выход *Direction*) и угловое положение ротора ВД (выход *Position*) в пределах одного его оборота. По числу и направлению полных оборотов ротора и его текущему угловому положению блок *S-функции* вычисляет угол поворота ротора.

Подсистема *Switches* (см. рис. 3), устройство которой показано на рис. 5, б, используется для управле-

ния транзисторными ключами силового преобразователя (линии PA7, PB0, PB1, PC6, PC7, PC8). Частота широтно-импульсной модуляции напряжения составляет 10 кГц и задается в настройках блока *Basic PWM*.

Подсистема *Controller* выполняет функции регулирования координат электропривода – тока, скорости вращения и угла поворота ротора вентильного двигателя. В контурах регулирования использованы ПИ-регулятор скорости и ПИД-регулятор положения, а также упреждающие связи по первой и второй производным задающего воздействия [6]. Регуляторы скорости и положения реализуются с помощью блоков ПИД-регуляторов из библиотек Simulink в дискретной форме (*z-форме*) с заданным шагом интегрирования (параметр настройки *Sample Time*).

Помимо задач регулирования координат система управления электропривода должна выполнять функции взаимодействия с системой управления верхнего уровня при функционировании в составе электротехнического комплекса или системы. Для этой цели предусмотрен обмен данными системы управления с персональным компьютером по последовательному порту. Интерактивное управление электроприводом

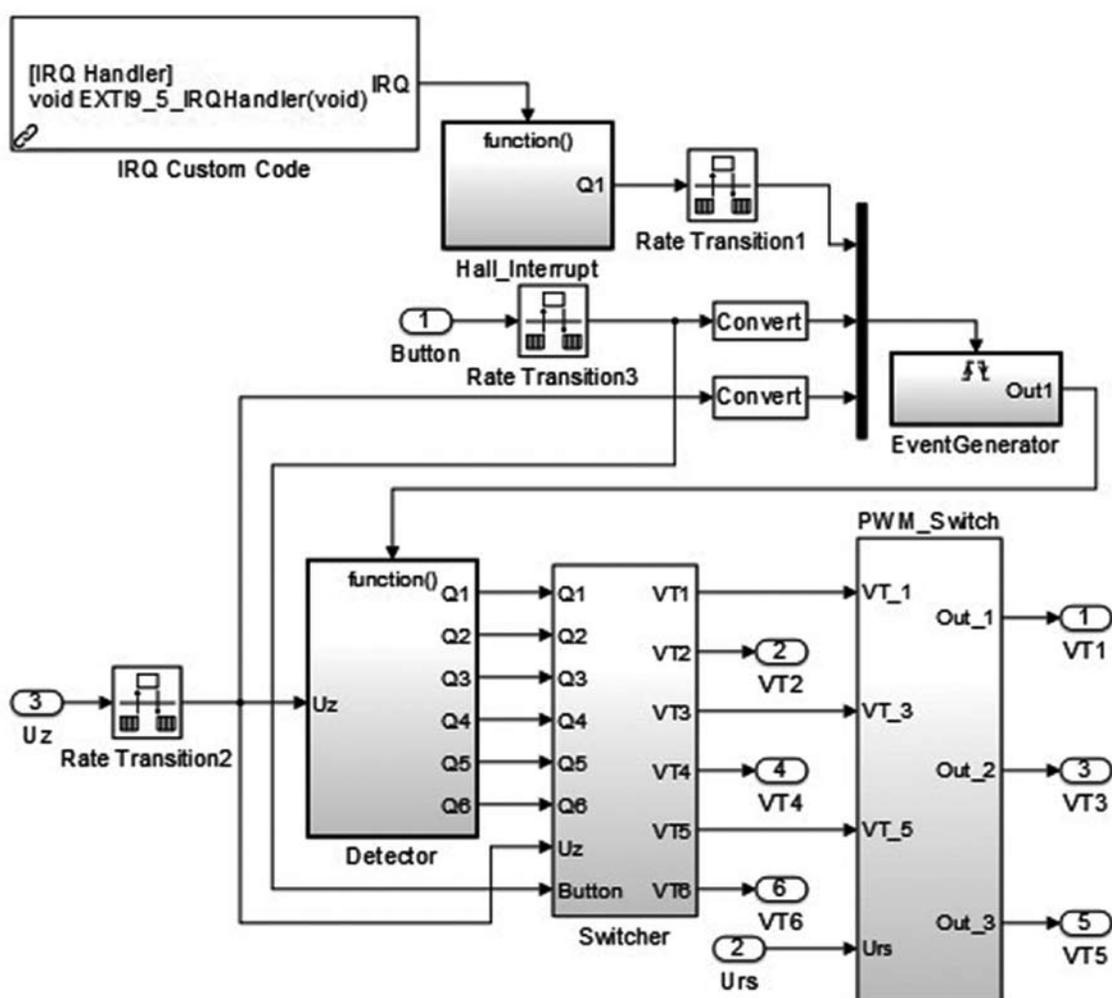


Рис. 4. Схема модели для переключения фаз вентильного двигателя по сигналам датчиков Холла

обеспечивает доступ к настройкам и параметрам системы управления в режиме реального времени. Этому способствует то обстоятельство, что параметры блоков Simulink, использованных в исполняемой модели, объединены в структурированный массив. Кроме задания управляющих воздействий в режимах регулирования координат предусмотрена функция записи параметров корректирующих звеньев и конфигурации аппаратной части электропривода. Система управления электропривода передает контрольную информацию о текущих координатах — угле поворота и скорости вращения ротора, потребляемом токе, а также информацию об исправности электропривода.

На рис. 6, а приведена схема модели обработчика пакетов данных, принятых по последовательному порту.

Пакет данных имеет фиксированную длину и состоит из нескольких полей: признак начала пакета, адрес абонента, код команды, два поля данных, контрольная сумма, признак конца пакета. Триггерные подсистемы `Message` и `Write_Param`, вызываемые по прерыванию от блока последовательного порта UART Rx библиотеки `Waijung`, реализуют протокол приема и обработки

данных. Блок `Message` выполняет проверку контрольной суммы и адреса абонента, а также выделяет блоки данных `Data1` и `Data2` формата `uint32`. Интерпретация пакета данных и выделение его полей осуществляется триггерной подсистемой `Write_Param` (рис. 6, б), которая вызывается по мере приема очередного пакета.

Функция записи принятых параметров в память микроконтроллера написана на языке Си и интегрирована в исполняемую модель с помощью блока `Basic Custom Code1`.

Разработанное программное обеспечение использовано при создании электропривода с вентильным двигателем в ООО НПО «Рубикон-Инновация» (г. Смоленск). Опытный образец электропривода с электрическим двигателем фирмы `Maxon` мощностью 10 Вт в настоящее время проходит испытания. Электропривод предназначен для использования в антенной установке. При реализации электропривода разработана плата управления с микроконтроллером `STM32F407` и устройствами для его сопряжения с датчиками и задающими устройствами. Для электропривода разработана плата силового преобразователя, рассчитанная

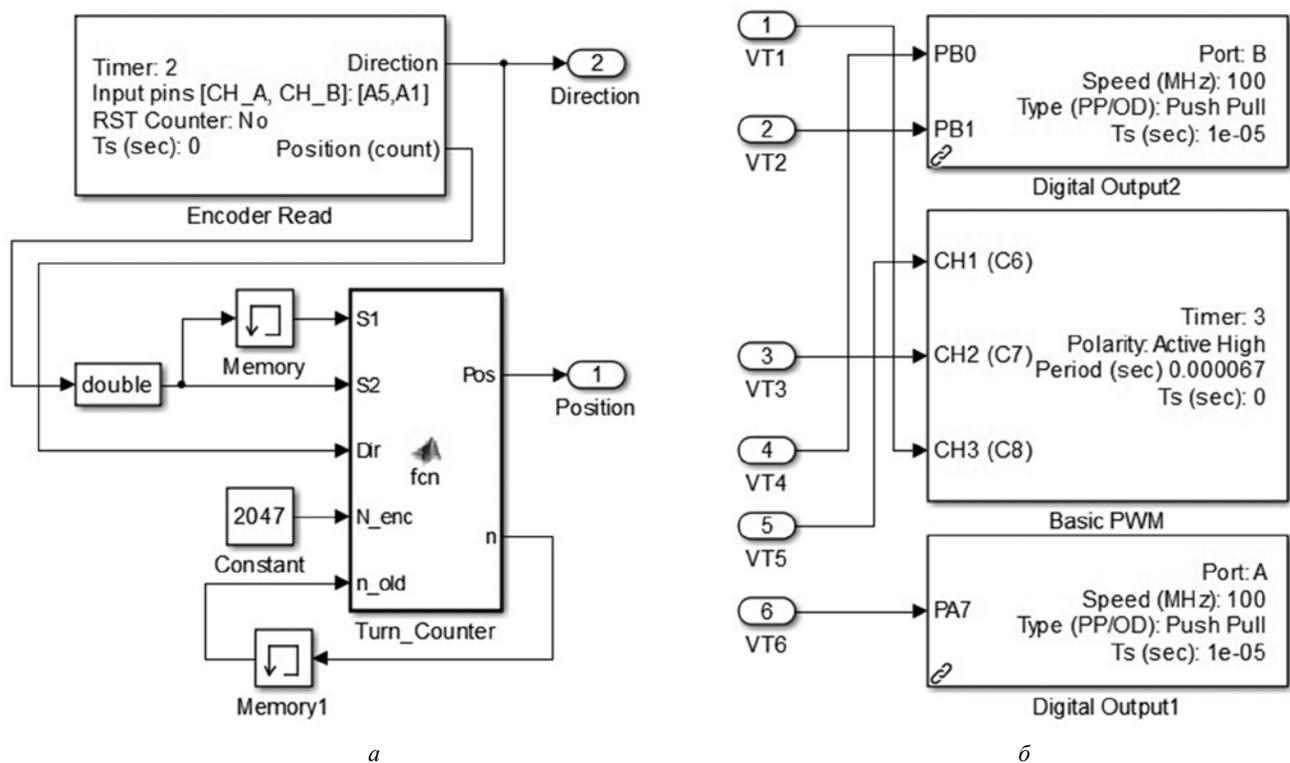


Рис. 5. Подсистемы исполняемой модели для вычисления угла поворота ротора (а) и управления силовым преобразователем (б)

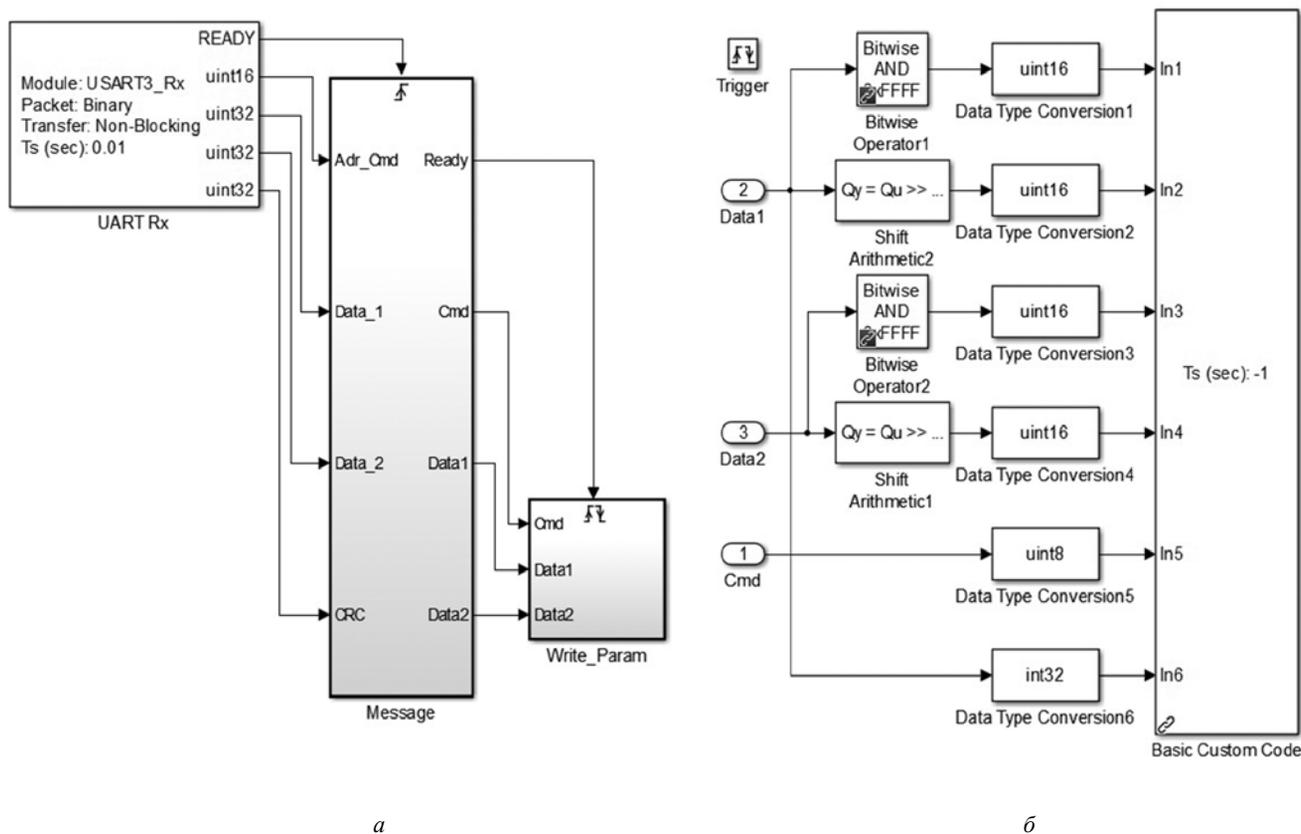


Рис. 6. Обработчик принятого пакета данных:

а — схема подсистемы исполняемой модели; б — схема обработки пакета данных

на мощность двигателя до 150 Вт. При испытаниях применялась экспериментальная установка, которая конструктивно повторяет механизм вертикальной оси антенной установки. Для проведения исследований электропривода было разработано специальное приложение для Windows, которое позволяет при управлении с компьютера задавать режимы и параметры движения, осуществлять оперативное управление и настройку корректирующих звеньев системы управления, а также отслеживать текущие координаты электропривода. Предусмотрено детектирование ошибок при обмене данными и аварийных режимов работы (превышение скорости движения, превышение допустимого тока, стопорение механизма). Параметры корректирующих звеньев могут быть сохранены на микросхему постоянного запоминающего устройства (ПЗУ) и прочитаны из нее по шине I2C, обмен данными по которой также реализован средствами модельно-ориентированного программирования. Разумеется, при практическом использовании электропривода в автономных условиях Matlab не требуется.

В целом система управления электропривода показала свою работоспособность при управлении вентильным двигателем в режимах слежения, стабилизации скорости движения и отработки движения по заданной программе. Точность регулирования координат и динамические свойства электропривода определяются структурой системы регулирования и настройками регуляторов.

Таким образом, опыт использования модельно-ориентированного программирования в рассматриваемом случае позволяет выделить основные преимущества этого метода.

Удобство и наглядность представления структуры системы управления и ее программного обеспечения в графическом виде.

Возможность непосредственного использования математических моделей или их составных частей для составления исполняемой модели.

Наличие библиотек функциональных блоков для использования программных и аппаратных ресурсов микроконтроллера.

При разработке не требуется знание особенностей языка программирования микропроцессорной техники.

В целом, основной практический результат, который может быть получен при использовании модельно-ориентированного программирования, состоит в том, что ускоряется и упрощается процесс разработки программного обеспечения системы управления технической системой.

Основными недостатками модельно-ориентированного программирования следует считать вероятность генерирования неоптимального программного кода, возможное недоиспользование ресурсов микроконтроллера, а также функциональные ограничения средств модельно-ориентированного программирования. Однако эти недостатки могут быть успешно устранены при грамотном подходе к проектированию с использованием различных средств разработки программного обеспечения и моделирования, предусмотренных в Matlab и пакетах его расширения.

Литература

1. **MathWorks.** Model-Based Design [Официальный сайт]. www.mathworks.com (дата обращения 10.05.2016).
2. **Черных И.В.** Моделирование электротехнических устройств в Matlab, SimPowerSystems и Simulink. СПб.: Питер, 2008.
3. **Дьяконов В.П.** Matlab 6/6.1/6.5 + Simulink 4/5 в математике и моделировании. М.: СОЛОН-Пресс, 2008.
4. **Waijung** Blockset [Электронный ресурс]. <http://waijung.aimagin.com> (дата обращения 11.05.2016).
5. **Терехов В.М.** Системы управления электроприводов / под ред. В.М. Терехова. М.: Издательский центр «Академия», 2005.
6. **Башарин А.В., Новиков В.А., Соколовский Г.Г.** Управление электроприводами. Л.: Энергоиздат. Ленинградское отделение, 1982.

Статья поступила в редакцию 23.05.2016