

УДК 621.3.07

Г.В. Федоров¹, А.С. Плехов^{1,2}

ПРИМЕНЕНИЕ МОДЕЛЬНО-ОРИЕНТИРОВАННОГО ПОДХОДА ПРИ ПРОЕКТИРОВАНИИ ЦИФРОВЫХ СИСТЕМ УПРАВЛЕНИЯ ЭЛЕКТРОТЕХНИЧЕСКИМИ ОБЪЕКТАМИ

¹Нижегородский государственный технический университет
им. Р.Е. Алексеева

²ООО «Энергосбережение»

Приводится описание современных тенденций в области проектирования систем управления электротехническими комплексами. Рассмотрена возможность применения модельно-ориентированного подхода при проектировании цифровых регуляторов систем управления электротехническими объектами. Дано описание процесса разработки с использованием модельно-ориентированного подхода на примере проектирования цифрового регулятора, который является элементом системы электропривода постоянного тока с подчиненным регулированием параметров. Приведено описание основных этапов работы в программной среде *Matlab/Simulink*.

Ключевые слова: модельно-ориентированное проектирование, программно-аппаратное моделирование, системы автоматического управления, регуляторы, электропривод, *Matlab/Simulink*, программируемая логическая интегральная схема (ПЛИС).

1. Введение

Проектирование систем управления электроприводами и другими электротехническими комплексами включает следующие этапы:

- построение модели объекта управления;
- анализ и построение регулятора;
- моделирование работы объекта и системы управления;
- реализация системы управления на целевой аппаратной платформе.

На современном этапе научно-технического прогресса использование традиционной методологии при разработке многоуровневых систем управления сложными электротехническими комплексами сопряжено с рядом трудностей: отсутствие единой среды разработки, поддерживающей все этапы проектирования устройства; дорогостоящий процесс верификации и тестирования, как готового изделия, так и его отдельных модулей, ввиду использования физических моделей и т.д. С учетом подобных об-

стоятельств использование модельно-ориентированного подхода (МОП) является перспективным направлением среди возможных методов проектирования [1, 2].

Подобные задачи удобно решать в графической среде имитационного моделирования. К наиболее распространенным программам, предназначенным для их решения, относятся: *Labview*, *SimInTech*, *Matlab/Simulink*. Далее будет рассматриваться описанный подход на примере программного пакета *Matlab/Simulink* [3], поскольку он имеет в своем составе библиотеки с готовыми блоками для моделирования устройств различных инженерных областей. Выбор среды *Matlab/Simulink* обусловлен также ее популярностью среди инженеров разных специальностей и областей для решения собственных задач технических вычислений и моделирования.

Стандартная библиотека *Matlab/Simulink* под названием *HDL Coder* содержит в себе элементы, предназначенные для реализации на их основе модели проектируемого цифрового устройства, для которой впоследствии возможно сгенерировать платформо-независимый, унифицированный *HDL*-код для дальнейшего развертывания синтезированной схемы на целевой аппаратной платформе. В качестве таковой рассматривается программируемая логическая интегральная схема (ПЛИС) [4]. Логика работы данного класса устройств не определяется при изготовлении, а задается посредством программирования с использованием языков описания аппаратуры: *Verilog*, *VHDL* и др.

Тестирование и верификация *HDL*-кода для реализации пропорционально-интегрального-регулятора выполняются с использованием программно-аппаратного моделирования. Данный вид моделирования подразумевает взаимодействие аппаратной платформы с пространством программной модели, которая симулирует внешние воздействия на реальное устройство, соединенное с ЭВМ, на которой выполняется моделирование, через физическое подключение. Такой подход позволяет идентифицировать ошибки при прототипировании и верификации на более ранних этапах проектирования изделия.

Набор тестовых значений подается как на *Simulink*-модель, так и на тестируемый компонент – запрограммированную плату – через физический интерфейс, реакция устройства на тестовый данные передается обратно в пространство *Simulink*-модели и сравнивается с результатами моделирования. При использовании подобной методики разработчику доступна интерактивная верификация, позволяющая управлять ходом процесса моделирования и выполнять дополнительный анализ сигналов. Становится доступным наглядное сравнение результатов. В настоящее время данный способ моделирования находит широкое применение в разработке и испытании встраиваемых систем, к которым также относятся цифровые

системы автоматического управления для электроприводов. Он позволяет во многом сократить количество ресурсов, затрачиваемых на разработку устройства.

II. Построение модели контура системы управления с последующей генерацией HDL-кода для выполнения верификации

В результате анализа литературных источников по рассматриваемой теме было выявлено отсутствие базовых примеров и наличие недосказанности в отношении описания маршрута проектирования цифровых систем по технологии МОП. Данный аспект носит негативный характер, поскольку разработчик, впервые ознакомившийся с данной технологией, зачастую не имеет возможности закрепить полученные знания, повторив эксперимент.

Рассмотрено использование описанного подхода при верификации HDL-кода для создания цифрового фильтра – регулятора, который находит активное применение при настройке внутреннего контура регулирования тока на модульный оптимум в системах электроприводов постоянного тока с подчиненным регулированием параметров [5]. Контур регулирования представлен на рис. 1.

Рассматриваемый внутренний контур образуется регулятором тока, силовым преобразователем, объектом регулирования и обратной связью по току через датчик тока.

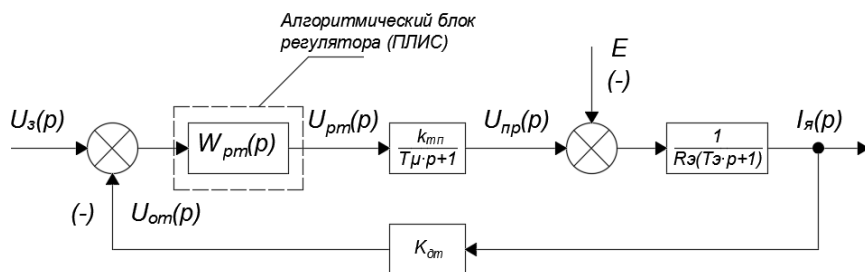


Рис.1. Структурная схема контура тока

При расчете регулятора тока контур настраивается на модульный оптимум, при этом влиянием ЭДС пренебрегают. В качестве регулятора тока в системе подчиненного регулирования электропривода постоянного тока следует применять пропорционально-интегральный регулятор. В качестве аппаратной платформы выступает отладочная плата на базе *Altera Cyclone IV EP4CE22F17C6N FPGA*. Использование программного обеспечения *Matlab/Simulink* [6] для генерации HDL-кода сопряжено с некоторыми ограничениями, накладываемыми на создаваемую разработчиком мо-

дель. Основным требованием является выполнение дискретизации синтезированной модели объекта и регуляторов, необходимость которой вызвана дискретным выполнением алгоритма регулирования на аппаратной платформе.

Необходимо иметь установленную среду разработки *Quartus II* для ПЛИС фирмы *Altera*, а также иметь установленные драйвера для работы интерфейса *JTAG*, через который осуществляется программирование ПЛИС в процессе тестирования. На рис. 2 представлена разработанная в среде *Matlab/Simulink* модель внутреннего контура регулирования тока, настроенная для генерации *HDL*-кода.

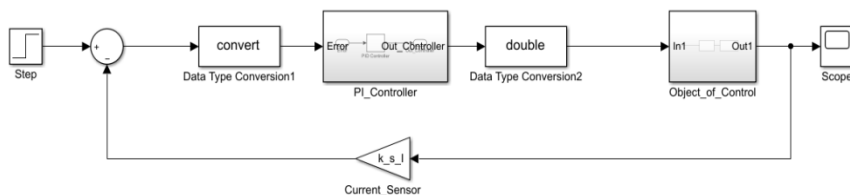


Рис. 2. *Simulink*-модель контура тока.

Подсистема *PI_Controller* содержит в себе блок *Discrete PID Controller*. Настраиваемые параметры блока *PID Controller* включают в себя: тип контроллера (*PID*, *PI*, *PD*, *P* или *I*), структуру контроллера (параллельную или идеальную), временную область (непрерывную или дискретную), ограничение выходного сигнала на заданном уровне и т.д. Регулятор настроен на использование арифметики с фиксированной точкой. Подсистема *Plant* содержит в себе передаточную функцию объекта, которая включает в себя одну большую постоянную времени и одну малую, а также суммарный коэффициент усиления контура тока и представляет собой последовательное соединение двух апериодических звеньев первого порядка [5]. При составлении передаточной функции объекта были использованы стандартные блоки *Simulink* для составления непрерывной модели – *Transfer Fun*, которые впоследствии были дискретизированы с использованием инструмента *Simulink Model Discretizer* методом экстраполяции нулевого порядка, частота дискретизации составляет 20 МГц.

Блоки *Data Type Conversion* предназначены для согласования типов данных между моделью регулируемого объекта *Object_of_Control* и регулятора *PI_Controller*. Цепь обратной связи содержит датчик тока, представленный пропорциональным звеном *Current_Sensor*. Для проведения предварительной настройки модели использовалась команда *hdlsetup*. Для получения возможности использовать *HDL Coder* совместно со сторонним

инструментом разработки устройств на ПЛИС, был указан системный путь к этому инструменту с помощью команды *hdlsetuptoolpath* в окне выполнения команд *Matlab*. Далее осуществлена идентификация используемой отладочной платы с помощью команды *fpgaBoardManager*. Следующим этапом является генерация *HDL*-кода для подсистемы *PI_Controller*. Для этого к подсистеме *PI_Controller* была применена команда *Generate HDL for Subsystem*. После успешной генерации кода, выполнен запуск *FPGA-in-the-Loop (FIL) Wizard* [7] через команду *filWizard*.

Simulink-модель, предназначенная для верификации сгенерированного *HDL*-кода, представлена на рис. 3.

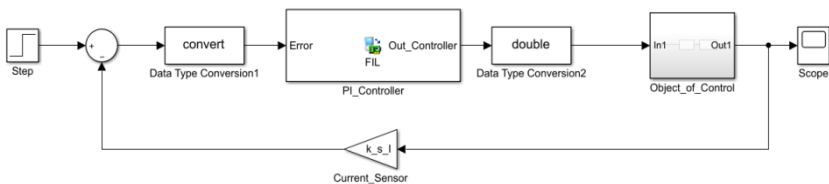


Рис. 3. *Simulink*-модель с импортированным *FIL*-блоком.

В процессе симуляции тестируемого *HDL*-кода на разработанной модели, получен график реакции системы на единичное ступенчатое воздействие. Таким же способом была протестирована исходная модель рис. 2. Результаты представлены на рис. 4. Высокое соответствие экспериментальных данных (ошибка менее 1 %) теоретическим может быть обусловлено высокой частотой тактирования отладочной платы, которая составляет 50 МГц, а также выбором большой разрядности данных для представления коэффициентов регулятора.

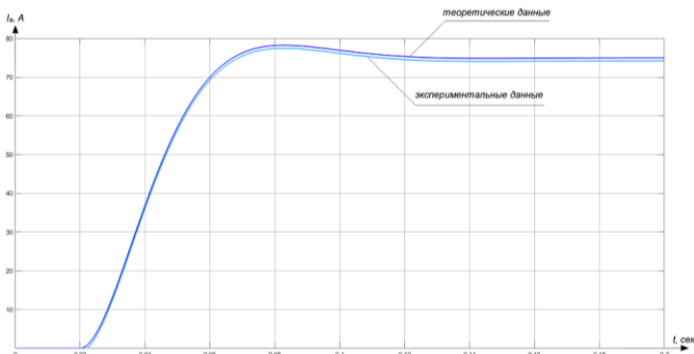


Рис. 4. Результат верификации *HDL*-кода с использованием *FIL*-блока

III. Заключение

Применение модельно-ориентированного подхода и программно-аппаратного тестирования заметно ускоряет процесс разработки, а также обеспечивает дополнительную визуализацию результатов при проведении тестов, заметно сокращая тем самым, время цикла разработки и процесс управления файлами, а именно, их хранение, систематизацию, передачу и координацию работы программистов при работе над большим проектом.

Использованные программные инструменты позволяют выполнять отладку алгоритмов управления электроприводов различного типа на потенциальной аппаратной платформе. В данной методике поведение объекта управления (электродвигателя) симулируется программно, в соответствии с применяемой моделью, что позволяет отказаться от физического моделирования. За счет этого сокращается расход ресурсов на процесс отладки и разработки системы, а в случае разработки систем управления высокоответственными объектами появляется возможность не подвергать их излишним перегрузкам в процессе отладки.

© Федоров Г.В., 2020

© Плехов А.С., 2020

Библиографический список

- [1] Деменков Н.П. Модельно-ориентированное проектирование систем управления // Промышленные АСУ и контроллеры. 2008. № 11. С. 66-69.
- [2] FPGA in the Loop // The MathWorks, Inc. [Электронный ресурс]. URL: <https://www.mathworks.com/help/hdlverifier/fpga-in-the-loop.html> (дата обращения 20.11.2019).
- [3] Черных И.В. Моделирование электротехнических устройств в MATLAB, SimPowerSystems и Simulink. М.: ДМК Пресс; СПб.: Питер, 2008. – 288 с.
- [4] Poon J.J., Kinsy M.A., Pallo N.A., Devadas S., Celanovic I.L. Hardware-in-the-loop testing for electric vehicle drive applications // 2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition (APEC), Feb. 5-9, 2012, Orlando, FL, USA. Pp. 2576-2582.
- [5] Тихомиров В.А., Мельников В.Л., Соколов В.В. Теория и расчет систем электропривода с подчиненным регулированием параметров. Н. Новгород: НГТУ, 2018. – 107 с.
- [6] Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. СПб.: БХВ – Петербург, 2002. – 608 с.
- [7] Matlab R2018b, версия 9.5.0.94 // The MathWorks, Inc. [Электронный ресурс]. URL: <https://www.mathworks.com/> (дата обращения 09.12.2019).

G.V. Fedorov¹, A.S. Plekhov^{1,2}

APPLICATION OF MODEL-ORIENTED APPROACH IN DESIGNING DIGITAL CONTROL SYSTEMS FOR ELECTROTECHNICAL OBJECTS

¹ Nizhny Novgorod State Technical University n.a. R.E. Alekseev

Nizhny Novgorod, Russia

² «Energoberezhenie» Co Ltd

Nizhny Novgorod, Russia

Abstract. A description of current trends in the field of design of control systems for electrical complexes is given. The possibility of applying a model-oriented approach to the design of digital controllers for control systems of electrical objects. A description of the development process using a model-oriented approach is given on the example of designing a digital controller, which is an element of a direct current electric drive system with subordinate control of parameters. The description of the main stages of work in the software environment Matlab / Simulink.

Key words: model-oriented design, hardware-in-the-loop simulation, automatic control systems, controllers, electric drive, Matlab/Simulink, FPGA.

References

- [1] N.P. Demenkov, “Model-based design of control systems”, *Industrial Automatic Control Systems and Controllers*, vol. 11, pp. 66-69, Nov. 2008.
- [2] FPGA in the Loop [Online]. Available at: <https://www.mathworks.com/help/hdlverifier/fpga-in-the-loop.html/> (Accessed: Nov. 20, 2019).
- [3] I.V. Chernyh, *Modelirovanie elektrotekhnicheskikh ustrojstv v MATLAB, SimPowerSystems i Simulink [Simulation of electrical devices in MATLAB, SimPowerSystems and Simulink]*. Moscow: DMK Press; St. Petersburg: Piter, 2008 (in Russian).
- [4] J.J. Poon, M.A. Kinsy, N.A. Pallo, S. Devadas and I.L. Celanovic, “Hardware-in-the-loop testing for electric vehicle drive applications”, in proc. *2012 Twenty-Seventh Annual IEEE Applied Power Electronics Conference and Exposition (APEC)*, Feb. 5-9, 2012, Orlando, FL, USA, pp. 2576-2582.
- [5] V.A. Tihomirov, V.L. Mel'nikov, V.V. Sokolov, *Teoriya i raschet sistem elektroprivoda s podchinennym regulirovaniem parametrov [Theory and calculation of electric drive systems with subordinate control of parameters]*. N. Novgorod: NNSTU, 2018 (in Russian).
- [6] R.I. Grushvickij, A.H. Mursaev and E.P. Ugryumov, *Proektirovanie sistem na mikroskhemah programmiruemoj logiki [Designing systems on programmable logic chips]*. St. Petersburg: BHV – Peterbug, 2002. (in Russian).
- [7] Matlab R2018b, version 9.5.0.94. [Online]. Available at: <https://www.mathworks.com/> (Accessed: Dec. 9, 2019).