

УДК 681.3.06.069

О.С. Бушлякова, Н.В. Замятин

Виртуальные и нейросетевые модели для отладки программного обеспечения без реального прибора

Предлагается метод отладки программного обеспечения (ПО) для бортовой аппаратуры (БА) без использования реального прибора. ПО разрабатывается при помощи комплекса программных средств – унифицированной системы «Лотос». Отладка ПО производится на виртуальной модели прибора, реализованной в программе для автоматизированного проектирования Proteus. Схемы сложных устройств представляются в виде нейросетевых моделей.

Ключевые слова: виртуальная модель, нейросетевая модель, программное обеспечение.

В процессе создания БА важным этапом является ее отработка при помощи контрольно-проверочной аппаратуры (КПА) и соответствующего ПО [1]. Рассмотрим схему испытаний прибора (рис. 1).

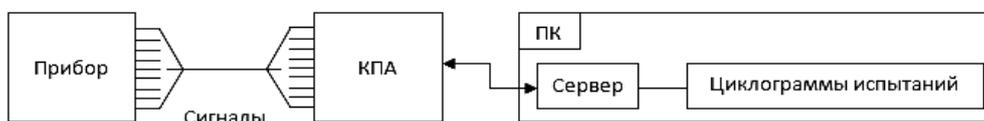


Рис. 1. Схема испытаний прибора

Испытания прибора проводятся для помощи КПА и персонального компьютера (ПК) с установленным на него ПО для испытаний. Алгоритм проверок сформулирован в циклограммы, при обработке которых на сервер отправляются сигналы. Сервер преобразует полученные сигналы и передает их КПА, которая взаимодействует с прибором. КПА в свою очередь передает полученные от прибора сигналы обратно серверу, в результате чего формируются протоколы проверок.

Для упрощения проведения испытаний реальные приборы заменяются их виртуальными моделями. Виртуальная модель представляет собой электрическую схему, которая показывает связи компонентов объекта, их параметры, а также формирует выходные и регистрирует входные сигналы, имитируя работу реального прибора. Испытания ПО при помощи виртуальных моделей прибора позволяют провести полный анализ работы БА без использования реального прибора, то есть еще до того, как прибор создан.

Схема испытаний с использованием виртуальной модели представлена на рис. 2.

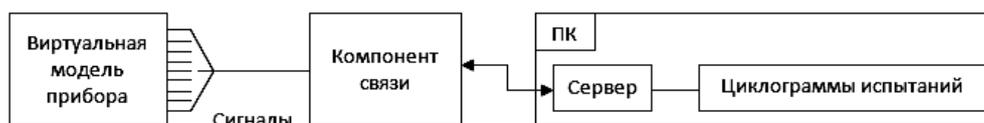


Рис. 2. Схема испытаний с виртуальной моделью прибора

Связь виртуальной схемы и ПО осуществляется посредством введения в нее специального компонента – протокола VSM, реализующего ввод/вывод сигналов и передачу этих сигналов серверу в виде доступных для использования переменных. Путем задания в циклограмме воздействий на схему прибора и контроля его реакций через данный компонент проводятся испытания прибора (его виртуальной схемы) и отлаживаются алгоритмы проверок [2].

Разработка ПО для испытаний приборов ведется в унифицированной системе «Лотос». Схема испытаний с использованием системы «Лотос» и виртуальной модели прибора представлена на рис. 3.

Компоненты системы обеспечивают выполнение следующих функций:

- подготовка входных данных;
- автоматизация проведения испытаний в режиме реального времени;
- непрерывный мониторинг состояния объекта контроля;
- формирование протоколов испытаний [3].



Система «Лотос» дает возможность унифицировать программные модули и использовать их в других проектах, таким образом, упрощая процесс разработки и автоматизируя испытания различных видов приборов.

Виртуальная модель прибора создается в пакете программ для автоматизированного проектирования электронных схем Proteus.

Рис. 3. Схема испытаний с использованием системы «Лотос» и виртуальной модели

Отличие от аналогичных по назначению пакетов программ, таких как, например, Electronics Workbench Multisim, MicroCap, Tina и тому подобных в развитой системе симуляции, включающей в себя интерактивную отладку в режиме реального времени и пошаговую отладку для различных семейств микроконтроллеров: 8051, PIC (Microchip), AVR (Atmel) и др. Proteus имеет обширные библиотеки компонентов, в том числе и периферийных устройств: светодиодные и жидкокристаллические индикаторы, температурные датчики, часы реального времени интерактивных элементов ввода-вывода: кнопок, переключателей, виртуальных портов и виртуальных измерительных приборов, интерактивных графиков, которые не всегда присутствуют в других подобных программах [4].

Пример виртуальной модели унифицированного модуля для выдачи команд представлен на рис. 4.

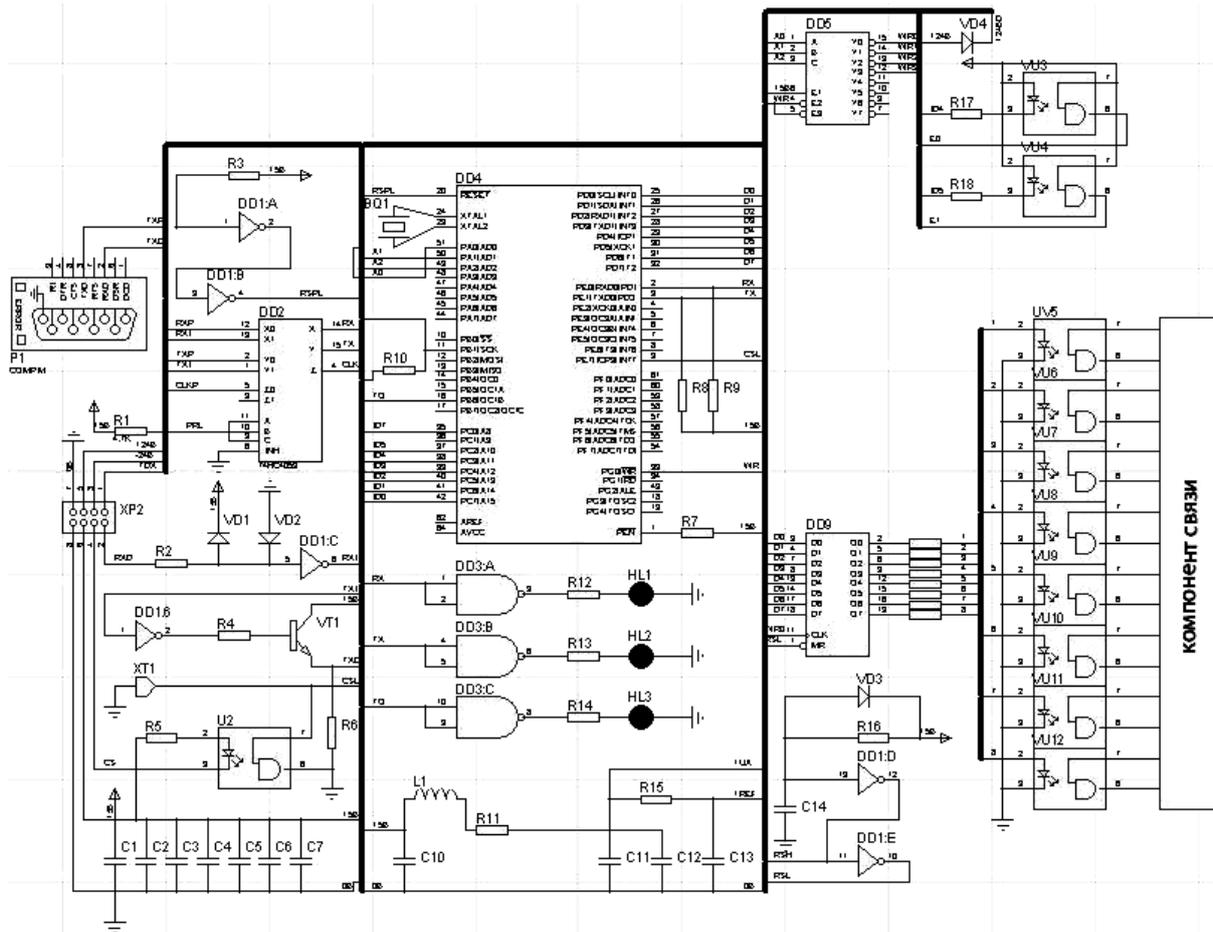


Рис. 4. Пример виртуальной модели

Модуль предназначен для выдачи 15 дублированных радиоконанд длительностью не более 0,6 с и 30 длительных команд. Все команды гальванически развязаны и рассчитаны на коммутацию напряжения не более 50 В и ток не более 0,5 А.

Модуль состоит из четырех ячеек A1...A4 для выдачи длительных команд на внешний соединитель XS1; четырех ячеек B1...B4, микросхем DD6...DD8, блоков DR1, DR2, резисторов R19, R20, конденсаторов C15...C18 для выдачи радиокоманд; узла управления записью (микросхема DD5); узла формирования сигнала сброса RSL при включении питания (микросхемы DD1.4, DD1.5, диод VD3, резистор R16, конденсатор C14); управляющего микроконтроллера ATMEGA128 (микросхема DD4); интерфейсной части (резисторы R2, R4...R6, диоды VD1, VD2, элементы DD1.3, DD1.6, оптроны VU1, VU2, переключатель XT1); узла программирования микроконтроллера DD4 (соединитель XP1, резисторы R1, R3, элементы DD1.1, DD1.2, микросхема DD2); фильтров по питанию (конденсаторы C1...C7, C10...C13, дроссель L1, резисторы R11, R15).

Для программирования микроконтроллера ATMEGA128 используется последовательный SPI-интерфейс. При подключении программатора к соединителю XP1 низкий уровень сигнала «PRL» подается на управляющие выводы 9...11 мультиплектора DD2. Таким образом, на выводы 2, 3, 11 микросхемы DD4 поступают соответственно сигналы «MOSI», «MISO», «SLK» от программатора. Сигнал «RSPL», поступающий на вывод 20 микросхемы DD4, переводит микроконтроллер ATMEGA128 в режим программирования.

Для программирования используется программа PonyProg2000. После загрузки программы нижнего уровня *.hex в микросхему DD4 программатор должен быть отключен от соединителя XP1. Управляющий сигнал «PRL» станет «1», и мультиплексор DD2 подключит UART-порт (выводы 2, 3) микросхемы DD4 к интерфейсу RS-232. На контакты A38, B38 соединителя XP2 приходит сигнал «TxD», на контакты A39, B39 – сигнал «RxD», на контакты A40, B40 – сигнальная земля 0В.

Резистор R2, диоды VD1, VD2 служат для ограничения сигнала «RxD» до TTL-уровня. Инверсный сигнал «RX» поступает через микросхему DD2 на вход приемника UART микросхемы DD4 (вывод 2). С выхода передатчика UART (вывод 3) сигнал «TX» через микросхему DD2, элемент DD1.6, резистор R4, оптрон VU1 поступает в магистраль в виде сигнала «TxD». Оптрон VU2 служит для приема сигнала «CS» (выборка МБП на магистрали) и разрешения работы передатчика VU1 на магистраль.

Индикация работы UART-порта микросхемы DD4 осуществляется элементами DD3.1, DD3.2, резисторами R12, R13, индикаторами HL1, HL2. Сигнал «TO» (вывод 16 микросхемы DD4) является вспомогательным и может использоваться для отображения каких-либо действий микросхемы DD4 в автономном режиме на индикаторе HL3.

Для выдачи ДК используется порт PD0...PD7 микроконтроллера DD4. Запись в соответствующий регистр (микросхема DD9 ячеек A1...A4) осуществляется по сигналам «WR0»...«WR3», формируемым микросхемой DD5 из сигналов «A0»...«A2» и «WR». При включении питания формируемый сигнал «RSL» сбрасывает регистры DD9 ячеек A1...A4 в ноль. При записи 31 и 32 ДК загораются индикаторы HL4 и HL5 соответственно.

Для выдачи ПК используется порт PC0...PC7 микроконтроллера DD4. Сигналы «PC0»...«PC3» используются для адресации. Сигнал «PC4» разрешает выдачу нечетных ПК через оптрон VU3. Сигнал «PC5» разрешает выдачу четных ПК через оптрон VU4. Сигнал «PC7» осуществляет пуск ПК. Длительность регулируется в пределах 0...0,6 с снятием сигнала «PC7» в ноль. Одновибратор DD6 аппаратно ограничивает длительность ПК на уровне 0,6 с. Индикацию 16 дублированной или 31 и 32 одиночных ПК можно наблюдать на светодиодах HL7 и HL6 соответственно. При включении питания сигналы сброса «RSL» и «RSH» блокируют выдачу ПК на внешний соединитель XS2.

Для связи схемы с внешними устройствами используется компонент P1 (COMPIM). При помощи данного компонента осуществляется взаимодействие с физическими устройствами через COM порт ПК.

Связь схемы с системой «Лотос» осуществляется при помощи компонента связи (протокола VSM). Компонент реализует передачу сигналов серверу, который преобразует полученные сигналы в переменные, используемые в циклограммах проверки.

При работе со схемами сложных устройств искусственные нейронные сети (ИНС) являются удобным и естественным базисом для представления информационных моделей. Нейросеть может быть достаточно формально определена [5], как совокупность простых процессорных элементов (часто называемых нейронами), обладающих полностью локальным функционированием и объединенных однонаправленными связями (называемыми синапсами). Сеть принимает некоторый входной сигнал из внешнего мира и пропускает его сквозь себя с преобразованиями в каждом процессорном элементе. Таким образом, в процессе прохождения сигнала по связям сети происходит его

обработка, результатом которой является определенный выходной сигнал. В укрупненном виде ИНС выполняет функциональное соответствие между входом и выходом и может служить информационной моделью G системы F .

Определяемая нейросетью функция может быть произвольной при легко выполнимых требованиях к структурной сложности сети и наличию нелинейности в переходных функциях нейронов [6]. Возможность представления любой системной функции F с наперед заданной точностью определяет нейросеть как компьютер общего назначения. Этот компьютер, в сравнении с машиной фон Неймана, имеет принципиально другой способ организации вычислительного процесса – он не программируется с использованием явных правил и кодов в соответствии с заданным алгоритмом, а обучается посредством целевой адаптации синаптических связей (и, реже, их структурной модификацией и изменением переходных функций нейронов) для представления требуемой функции.

Схема устройства представляется в виде «черного ящика», который характеризует собой модель, предсказывающую выходные параметры, полученные в ходе задания входных параметров и проведения численных экспериментов (рис. 5).

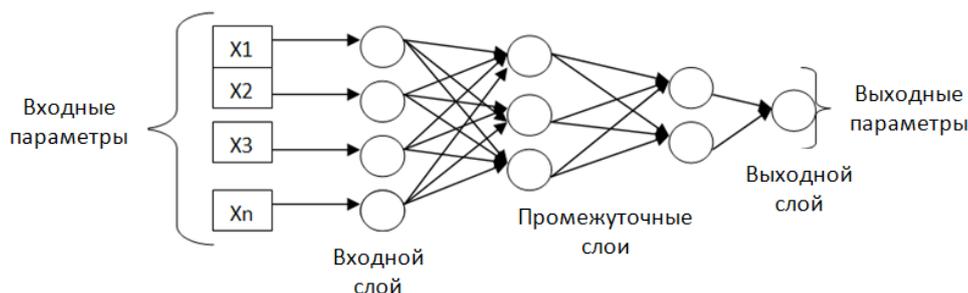


Рис. 5. Пример нейросетевой модели

В методе «черного ящика» моделируется внешнее функционирование системы. С точки зрения пользователя модели структура системы скрыта в самом «черном ящике», который имитирует поведенческие особенности системы.

Модель синтезируется путем выбора параметров из условия наилучшего, при заданной функции ценности, соответствия решений уравнений поведению системы. При этом структура системы никак не отражается в структуре уравнений модели.

Функционирование системы описывается на основе данных экспериментов или наблюдений над реальной системой.

Преимущество использования виртуальной и нейросетевой моделей заключается в том, что процесс моделирования позволяет представить всю схему взаимодействия устройств прибора, выбрать необходимые режимы работы и параметры, а также произвести отладку ПО без использования реального прибора. Предложенный метод экономит время и улучшает качество разработки.

Литература

1. Модернизированная контрольно-испытательная аппаратура комплекса автоматики и стабилизации / А.В. Морозов, А.В. Дунаев, Л.В. Селиванова, П.Н. Кибин // Электронные и электромеханические системы и устройства : тез. докл. науч.-техн. конф. молодых специалистов (Томск, 10–11 апр. 2008 г.). – Томск: ОАО «НПЦ «Полус», 2008. – С. 15–16.
2. Замятин Н.В. Информационная система для оптимизации процесса разработки и отладки программного обеспечения / Н.В. Замятин, А.В. Дунаев, О.С. Голышева // Научная сессия ТУСУР–2010: матер. докл. Всерос. науч.-техн. конф. студентов, аспирантов и молодых ученых. – Томск: ТУСУР, 2010. – С. 285–287.
3. Дунаев А.В. Система «Лотос» – унифицированная система для испытаний выпускаемых приборов / А.В. Дунаев, С.В. Авдошкин, О.С. Голышева // Электронные и электромеханические системы и устройства : тез. докл. науч.-техн. конф. молодых специалистов (Томск, 10–11 апр. 2008 г.) / ОАО «НПЦ «Полус». – Томск, 2008. – С. 83–85.
4. FAQ (ЧаВО) по PROTEUS для начинающих и не только [Электронный ресурс]. – Режим доступа: <http://kazus.ru/forums/showthread.php?t=13198>, свободный (дата обращения: 12.01.2012).
5. Типы информационных моделей [Электронный ресурс]. – Режим доступа: <http://academout.ru/referates/45/6.php>, свободный (дата обращения: 16.02.2012).

6. Нейросетевые информационные модели сложных инженерных систем [Электронный ресурс]. – Режим доступа: <http://www.intuit.ru/department/expert/neuroinf/4/2.html>, свободный (дата обращения: 16.02.2012).

Бушлякова Ольга Станиславовна

Аспирант каф. автоматизации обработки информации ТУСУРа,
инженер-программист ОАО «НПЦ «Полюс»
Тел.: (382-2) 55-49-48, доп. 897
Эл. почта: polus@online.tomsk.net

Замятин Николай Владимирович

Д-р техн. наук, профессор каф. автоматизации обработки информации ТУСУРа
Тел.: (382-2) 41-34-29
Эл. почта: zam@fet.tusur.ru

Bushlyakova O.S., Zamyatin N.V.

Virtual and neural network models for software testing without real devices

A method for testing of on-board equipment software without real devices is offered. The software is developed with the help of software complex - unified system «Lotos». Software testing in a device virtual model is designed in automated design program «Proteus». The schemes of complex devices are presented as network models.

Keywords: virtual model, neural network model, software.