

Министерство образования и науки Российской Федерации

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра математической
кибернетики и компьютерных наук

**СРАВНЕНИЕ ПОДХОДОВ К РАЗРАБОТКЕ СИСТЕМЫ УПРАВЛЕНИЯ
КОФЕЙНЫМ АВТОМАТОМ В СРЕДАХ IBM RATIONAL RHAPSODY
И MATLAB SIMULINK**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Качанова Михаила Олеговича

Научный руководитель

доцент, к. т. н.

Д. Ю. Петров

Заведующий кафедрой

к. ф.-м. н.

С. В. Миронов

Саратов 2018

ВВЕДЕНИЕ

В наше время для больших систем управления, требующих отказоустойчивость, доступность и понятность кода, распределенность по блокам, таким как высокотехнологичные виды вооружений, автоматизированные технологические линии, требуется очень много времени и сил на разработку. Для того, чтобы ускорить и упростить процесс создания этих систем, были разработаны программные среды, которые позволяют с помощью интуитивных для человека графических средств создавать сложные модели, в которых можно проследить все протекающие процессы, а потом уже сгенерировать программный код.

Одними из таких сред являются Matlab Simulink и IBM Rational Rhapsody. У них есть как очень похожие черты, так резкие отличия. В этой работе необходимо сравнить эти среды по различным факторам.

Для этого необходимо выполнить следующее:

1. разработать идентичные модели системы управления кофейным автоматом в вышеуказанных средах;
2. применить методы разработки встраиваемых систем в обеих средах;
3. сравнить подходы к разработке и найти сходства и различия.

В бакалаврской работе представлено три главы. Первая описывает существующие методы, использующие автоматное программирование: процесс Harmony для Rational Rhapsody, способы разработки в Matlab Simulink и во встроенной среде Stateflow и достижения в этой области нашего соотечественника, А. А. Шалыто.

Во второй главе подробно описана разработка кофейного автомата в программных средах Matlab Simulink и Rational Rhapsody, а так же показаны примеры правильной работы этих программ.

В третьей главе работы как итог представлено сравнение реализаций в вышеупомянутых средах: по сложности разработки, по кодогенерации и применению во встраиваемых системах, по возможностям отладки.

Далее представлены заключение, в котором подведены итоги сравнения и выявлены особенности автоматного программирования, список использованных источников с двадцатью источниками и два приложения.

В первом приложении представлена один из главных файлов полученной программы в Matlab Simulink, код объекта "coffee".

Во втором приложении представлены все файлы с кодом получившихся двух программ.

1 Подходы к проектированию

Harmony процесс как метод разработки в Rational Rhapsody

Процесс Harmony [1] разбивается на фазы, которые могут рассматриваться как деятельности самого крупного масштаба. Каждая фаза определяется с помощью одного или нескольких последовательностей работ. Каждая последовательность работ определяет упорядоченный набор деятельностей простых задач, выполняемых участниками процесса, и результирующие артефакты. Для отображения последовательностей работ часто используются диаграммы деятельности UML. Артефакты - это то, что создается или изменяется при выполнении определенных деятельностей в рамках процесса. Наиболее значимым артефактом является разрабатываемая система; однако существует множество других артефактов — таких как, исходный код, модель программного обеспечения, спецификация требований, и так далее.

Simulink и Stateflow

Simulink [2] – это среда имитационного графического моделирования. Она позволяет строить динамические модели, включая дискретные, непрерывные и гибридные, нелинейные и разрывные системы, с помощью блок-схем и направленных графов.

Интерактивная среда Simulink, позволяет использовать уже готовые библиотеки блоков для моделирования электросиловых, механических и гидравлических систем, а также применять развитый модельно-ориентированный подход при разработке систем управления, средств цифровой связи и устройств реального времени.

Согласно официальному сайту [3], Matlab Simulink Stateflow - это инструмент численного моделирования систем, который позволяет разрабатывать сложные гибридные системы. Такими системами, например, являются бытовые приборы, высокотехнологичные виды вооружений, автоматизированные технологические процессы. Все они состоят из аналоговых и цифровых частей. Поэтому основная особенность гибридных систем - это дискретная и непрерывная динамика [4].

В настоящее время для моделирования дискретной динамики реактивных систем широко используется визуальный формализм - Statechart (диаграммы состояний и переходов). Основные неграфические компоненты таких диаграмм - это событие и действие, основные графические компоненты - со-

стояние и переход [5].

Событие - нечто, происходящее вне рассматриваемой системы, возможно требуя некоторых ответных действий. События могут быть вызваны поступлением некоторых данных или некоторых задающих сигналов со стороны человека или некоторой другой части системы. События считаются мгновенными (для выбранного уровня абстрагирования).

Действия - это реакции моделируемой системы на события. Подобно событиям, действия принято считать мгновенными.

Состояние - условия, в которых моделируемая система пребывает некоторое время, в течение которого она ведет себя одинаковым образом. В диаграмме переходов состояния представлены прямоугольными полями со скругленными углами.

Переход - изменение состояния, обычно вызываемое некоторым значительным событием. Как правило, состояние соответствует промежутку времени между двумя такими событиями. Переходы показываются в диаграммах переходов линиями со стрелками, указывающими направление перехода.

Постановка задачи

Была поставлена задача реализовать работу кофейного автомата в программных средах Matlab Simulink с помощью встроенной среды Stateflow и Rational Rhapsody. Система должна выполнять следующие действия:

- в машине должен присутствовать бойлер, который поддерживает нужную температуру воды;
- машина должна осуществлять подачу воды, молока, кофе, иметь пресс;
- тех. обеспечение может пополнять запасы молока, кофе, стаканчиков;
- пользователь (например, покупатель) имеет следующие возможности: выбрать тип кофе, вставить деньги, забрать стаканчик.

Далее на основе полученного опыта необходимо сравнить методы разработки в этих средах по следующим факторам:

1. по сложности разработки;
2. по кодогенерации и применению во встраиваемых системах;
3. по возможностям отладки.

2 Разработка системы управления кофейным автоматом

Разработка в среде Rational Rhapsody

Была разработана система управления кофейным автоматом в среде Rational Rhapsody с помощью диаграмм вариантов использования, последовательности, activity и структурных диаграмм.

Сначала были построены варианты использования, затем, когда все требования определены, была создана структурная диаграмма, где были описаны все классы и данные.

Далее, самое важное, нужно было разработать для каждого класса диаграммы состояний (см.рис. 1) и описать все методы в них.

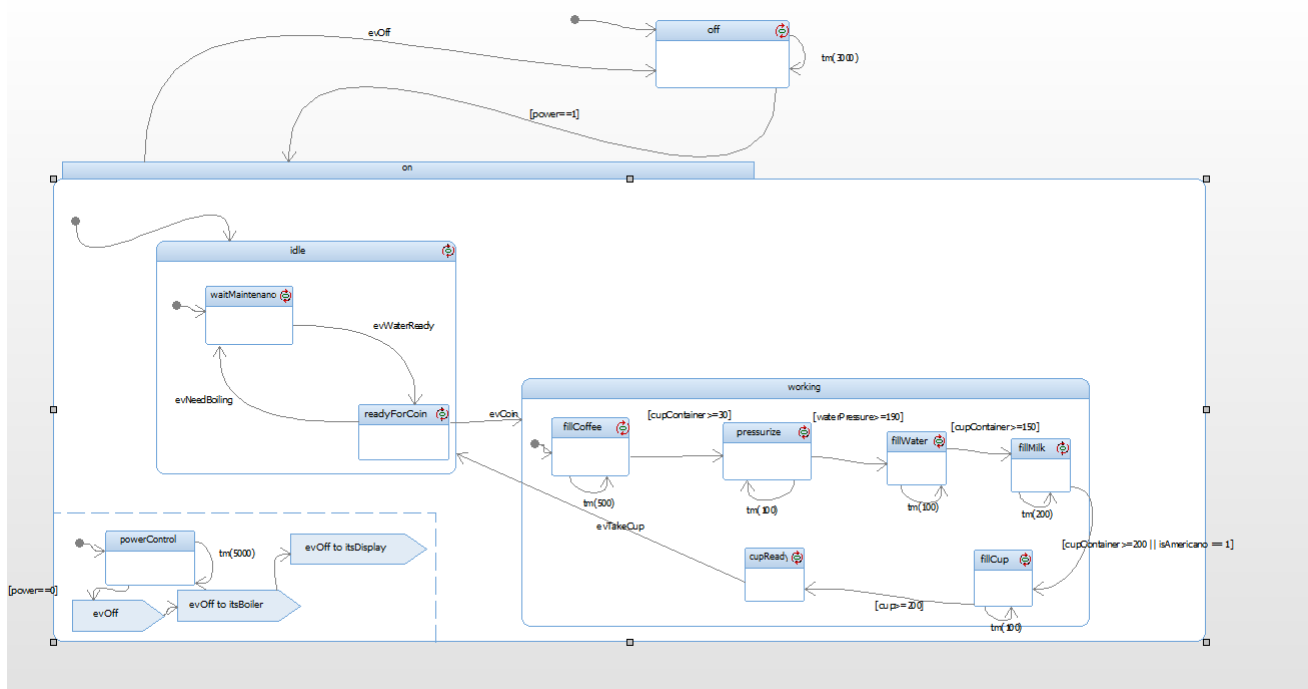


Рисунок 1 – Класс CoffeeMachine

Затем необходимо создать диаграмму последовательности и панельную диаграмму (см.рис. 2), в которой будут отображаться интерфейс и индикаторы. Далее система была протестирована и было установлено, что все работает правильно. Был получен C++ код с помощью встроенной в Rhapsody кодогенерации.

Разработка в среде Matlab Simulink

Для начала, необходимо построить основную схему, которая будет отвечать за логику. Это диаграмма состояний — Statechart. Для этой диаграммы необходимо построить состояния, в данном случае это состояния events (состояние, где обрабатываются события), Boiler (состояние, где написана логи-

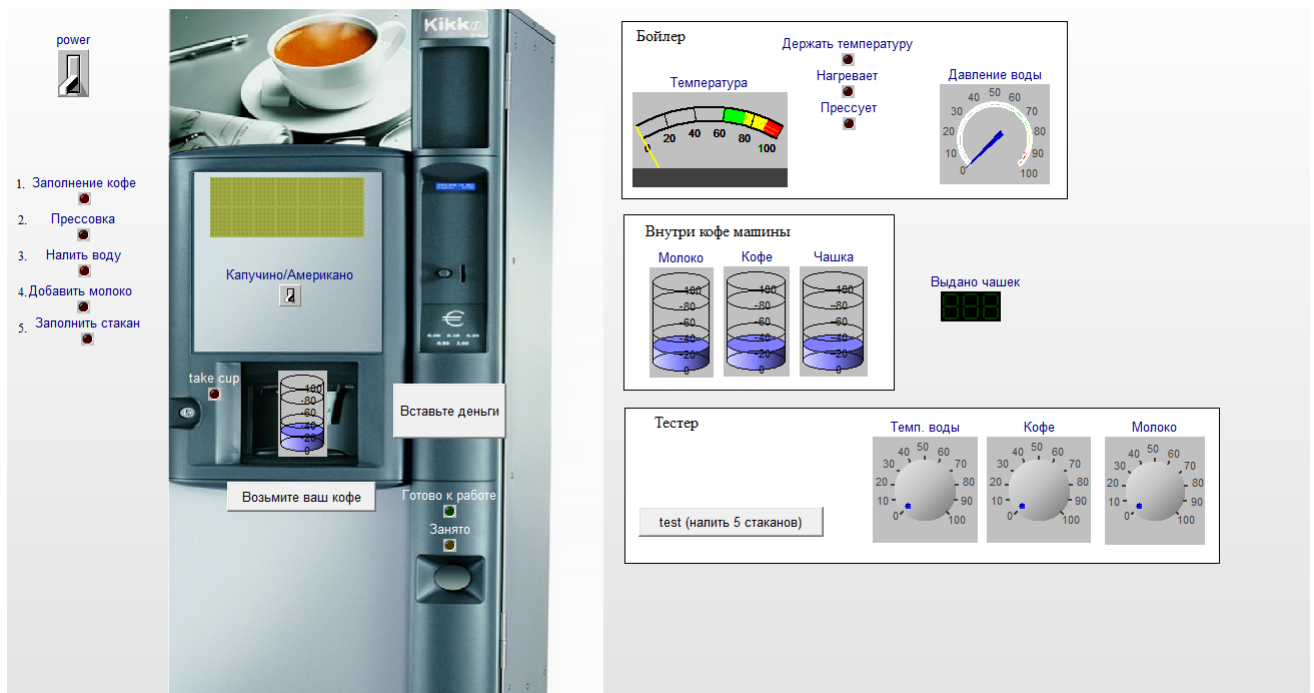


Рисунок 2 – Интерфейс

ка бойлера), CoffeeMachine (состояние, где написана логика самой системы управления автоматом). Для этих состояний были построены диаграммы под-состояний (см.рис. 3).

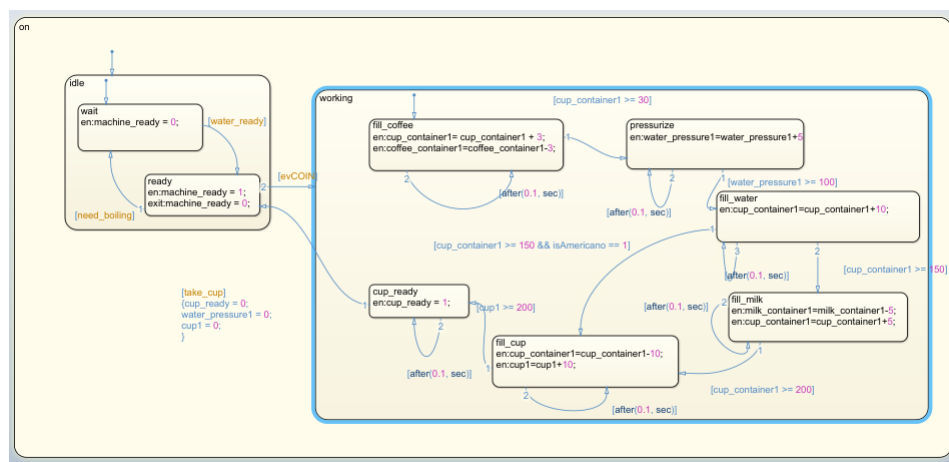


Рисунок 3 – Подсостояние CoffeeMachine — on

Далее были построены диаграмма последовательности для отладки и все индикаторы и интерфейс с помощью встроенной библиотеки Dashboard (см.рис. 4).

Программа была отлажена и было проверено, что работает она правильно. Далее был получен код на языке C++ с помощью встроенной кодогенерации.

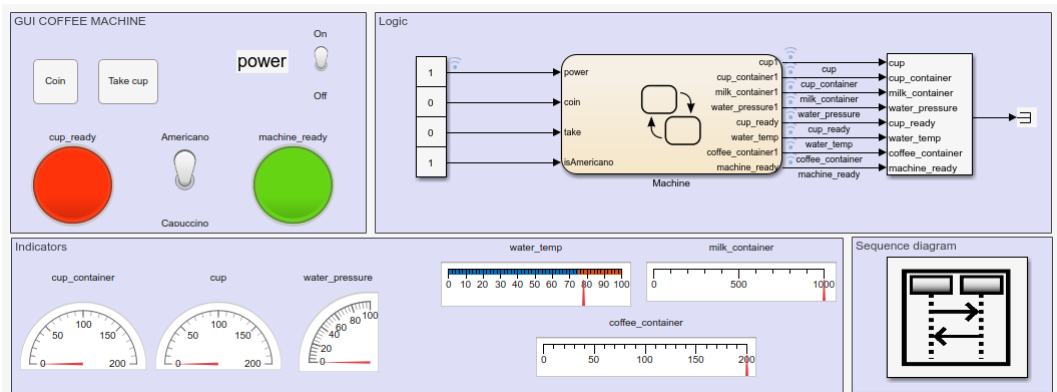


Рисунок 4 – Готовая программа

3 Сравнение реализаций проекта

По сложности и предназначению

В ходе разработки было выявлено, что Rhapsody предназначен для очень больших проектов с применением процесса Harmony, то есть начиная с задачи определения требований, и заканчивая интеграцией и тестированием. Для таких проектов требуются много ресурсов и времени. Как правило, много групп разработчиков работают над одним проектом и метод визуализации моделирования очень помогает понимать работу каждого отдельного человека. Но для таких относительно небольших проектов, как кофейный автомат эта система более чем избыточна, требует слишком много усилий. Matlab Simulink, в свою очередь, не имеет такого богатого функционала в плане определения требований и спецификации, но позволяет быстрее и не менее качественно создавать небольшие проекты, что более подходит для имитационного моделирования.

По кодогенерации и применению во встраиваемых системах

В программной среде Rational Rhapsody создается намного больше строк кода из-за большего количества компонентов, что нужно и удобно при разработке сложных систем, но при создании такой системы, как кофейных автомат больше подходит Matlab Simulink, в которой не создается много лишних файлов и данных (см.рис. 5).

```
case fillWater:
{
  if (is_EVENT_TYPE_OF(OnTimeoutEventId))
  {
    if (GetCurrentEvent() == working_timeout)
    {
      NOTIFY_TRANSITION_STARTED("14");
      popMultiTransition();
      if (working_timeout != NULL)
      {
        working_timeout->cancel();
        working_timeout = NULL;
      }
      NOTIFY_STATE_EXITED("ROOT.on_state_
6.working_fillWater");
      NOTIFY_STATE_ENTERED("ROOT.on_state_
6.working_fillWater");
      popMultiTransition();
      working_subState = fillWater;
      state_f_active = fillWater;
      //!! state ROOT.on_state_
6.working_fillWater (Entry)
      cupContainer+=10;
      itsDisplay_sendMessage(Display::filling_water);
      //!!
      working_timeout =
      scheduleTimeout(100,
      "ROOT.on_state_6.working_fillWater");
      NOTIFY_TRANSITION_TERMINATED("14");
      res = eventConsumed;
    }
  }
  else if (is_EVENT_TYPE_OF(OnNullEventId))
  {
    //!! transition 17
    if (cupContainer==5)
    {
      NOTIFY_TRANSITION_STARTED("17");
      popMultiTransition();
      if (working_timeout != NULL)
      {
        working_timeout->cancel();
        working_timeout = NULL;
      }
      NOTIFY_STATE_EXITED("ROOT.on_state_
6.working_fillWater");
      NOTIFY_STATE_ENTERED("ROOT.on_state_
6.working_fillMilk");
      popMultiTransition();
      working_subState = fillMilk;
      state_f_active = fillMilk;
      //!! state ROOT.on_state_
6.working_fillMilk (Entry)
      milkContainer+=5;
      cupContainer+=5;
      itsDisplay_sendMessage(Display::filling_milk);
      //!!
      working_timeout =
      scheduleTimeout(200,
      "ROOT.on_state_6.working_fillMilk");
      NOTIFY_TRANSITION_TERMINATED("17");
      res = eventConsumed;
    }
  }
}
}
break;
```

Код в Rhapsody

Код в Simulink

Рисунок 5 – Зависимость сигнала от шума для данных.

Из Rhapsody напрямую можно сгенерировать код только для языков C, C++, Java и Ada, в то время как из Simulink есть возможность генерировать код

в С и С++, а также в ней есть пакеты, которые позволяют работать напрямую с микроконтроллерами Arduino, STM32, Raspberry Pi и так далее, что позволяет небольшим командам разработчиков сильно ускорить моделирование системы, убирая вопрос интеграции кода в контроллер.

Способы отладки

- анимированные диаграммы последовательности и состояний;
- посылка сообщений, паузы;
- панельные диаграммы;

Matlab Simulink же в этом не уступает, в нем есть свой очень удобный функционал:

- те же анимированные диаграммы последовательности и состояний;
- удобный дебаггер;
- всевозможные визуализаторы сигналов;
- Simulation Data Inspector.

В итоге получается, что в плане отладки модели эти среды почти идентичны.

ЗАКЛЮЧЕНИЕ

В результате были построены идентичные модели системы управления кофейным автоматом в программных средах Matlab Simulink и IBM Rational Rhapsody, в ходе этой работы были выявлены общие особенности программирования с помощью автоматов и диаграмм: наглядность всех процессов, надежность работы системы, легкая отладка.

На основе выполненной работы, можно подытожить сравнение методов разработки в этих средах (см. таб. 1).

Таблица 1 – Сравнение методов разработки в Matlab Simulink и IBM Rational Rhapsody

	Rational Rhapsody	Matlab Simulink
Кодо-генерация	C++, C, Java	C++, C, интеграция с микроконтроллерами
Стадия разработки	Все стадии от определения требований и заканчивая интеграцией и тестированием	Имитационное моделирование
Отладка	Активные диаграммы состояний и последовательности, панельные диаграммы	Активные диаграммы состояний и последовательности, всевозможные визуализаторы сигналов
Применение	В крупных распределенных проектах	В средних и маленьких проектах
Сложность	Большие затраты по времени и силам ради понятности, безотказности системы, соответствия требованиям	Меньшие затраты по времени и силам, не уступающее по качеству, но меньшие возможности

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 The Harmony Process[Электронный ресурс].— URL: https://www.ibm.com/support/knowledgecenter/en/SSB2MU_8.2.1/com.btc.tcatg.user.doc/topics/atgreqcov_SecSysControllerHarmony.html (Дата обращения 31.05.2018). Загл. с экр. Яз. англ.
- 2 Simulink[Электронный ресурс].— URL: <http://matlab.ru/products/simulink> (Дата обращения 31.05.2018). Загл. с экр. Яз. рус.
- 3 Stateflow[Электронный ресурс].— URL: <https://matlab.ru/products/stateflow> (Дата обращения 31.05.2018). Загл. с экр. Яз. рус.
- 4 SIMULATION OF HYBRID SYSTEMS USING STATEFLOW[Электронный ресурс].— URL: <http://www.isir.upmc.fr/files/esm00.pdf> (Дата обращения 31.05.2018). Загл. с экр. Яз. англ.
- 5 Stateflow Diagram Objects[Электронный ресурс].— URL: <http://radio.feld.cvut.cz/matlab/toolbox/stateflow/concept4.html> (Дата обращения 31.05.2018). Загл. с экр. Яз. англ.