

УДК 681.513

Л.А. Шувалова, Д.Н. Моамар, Т.Ю. Уткина

Черкасский государственный технологический университет, Черкассы

## СТРУКТУРА ПРОГРАММНОГО КОМПЛЕКСА СИНТЕЗА И ВЕРИФИКАЦИИ МОДЕЛЕЙ ЦИФРОВЫХ АВТОМАТОВ

В статье рассмотрено назначение и условия применения программного обеспечения, которое входит в состав программного комплекса; предназначенного для синтеза алгоритмов автоматизированного управления фасовочно-упаковочными машинами. Детально освещены методы и способы лингвистической компиляции, описаны типы используемых ими данных. Дано описание режимов, в которых может применяться или модифицироваться комплекс для расширения его функций.

**Ключевые слова:** программный комплекс, верификация моделей, цифровой автомат.

### Введение

Развитие микроэлектроники диктует новые требования к технологиям проектирования цифровых изделий. Основным параметром любой разработки – time-to-market – предполагает активизацию усилий ведущих фирм мира в целях поиска новых методов и средств моделирования, верификации и тестирования проектируемых устройств систем управления.

Во многих автоматизированных системах используются управляющие автоматы, которые представляют устройства с конечным числом состояний. Различают два типа автоматов: автомат Мура и автомат Мили, вопросы формального описания которых подробно рассмотрены в работах [1, 2]. Однако принципы построения цифровых автоматов по заданным циклограммам состояний исполнительных механизмов автоматизированных систем раскрыты недостаточно подробно.

**Цель данной статьи** заключается в разработке структуры программного комплекса, предназначенного для автоматизированного синтеза алгоритмов и программ управления исполнительными механизмами фасовочно-упаковочных машин (ФУМ) в среде Active-HDL и выбора оптимальных сценариев управления, обеспечивающих повышение производительности расфасовки и упаковки продукции, улучшающих качество продукции и уменьшающих технологические потери фасуемых компонентов.

### 1. Представление алгоритма функционирования ФУМ в виде конечного автомата

Рассмотрим функционирование автомата, управляющего механизмами ФУМ, согласно циклограмме, разработанной технологом и приведенной на рис. 1. ФУМ обеспечивает отделение стаканчиков, заполняет их продуктом с помощью дозатора, закрывает фольговой крышкой, которую маркирует и приваривает к стаканчику, а затем укупоривает

полимерной крышкой [3].

Технологические процессы выполняются параллельно, но при этом механизмы должны согласовывать свои действия для достижения общей цели. Если в установленный момент времени не достигнуто одно из заданных на циклограмме положений каким-либо исполнительным механизмом, о чём свидетельствуют сигналы, полученные от соответствующих датчиков, то происходит аварийный останова машины.

Разработка графа автомата в среде Active-HDL осуществляется инструментальными средствами FSM-редактора [4]. Однако продолжительность включенного состояния некоторых исполнительных механизмов не зависит от сигналов, поступающих от датчиков, а устанавливается согласно требованиям технологического процесса. Для синтеза графа автомата ФУМ, формирующего управляющие сигналы заданной длительности, предлагается ввести вспомогательные переменные:

$$Z_j\_setj: TIME:= 0 \text{ sec}; Z_j\_res: TIME:= 0 \text{ sec}; \\ Z_j\_delta: TIME:= \delta j,$$

где  $Z_j\_setj$  – переменная, фиксирующая момент включения  $j$ -го механизма;  $Z_j\_res$  – переменная, фиксирующая момент выключения  $j$ -го механизма;  $Z_j\_delta$  – переменная, фиксирующая продолжительность включенного состояния  $j$ -го механизма;  $\delta j$  – временной интервал, заданный технологом.

В момент включения  $j$ -го механизма выполняются следующие действия:  $Z_j\_set:= \text{now}$ ;  $Z_j\_res:= \text{now}$ , при этом данным переменным присваивается текущее значение модельного времени. Затем в текущем состоянии автомата проверяется неравенство  $(Z_j\_res - Z_j\_set) < Z_j\_delta$ . Если оно выполняется, то к переменной  $Z_j\_res$  добавляется значение периода синхросигналов:  $Z_j\_res:= Z_j\_res + tCLK$ . Если выполняется условия выражения:  $(Z_j\_res - Z_j\_set) \geq Z_j\_delta$ , то  $j$ -й механизм выключается и автомат переходит в новое состояние.

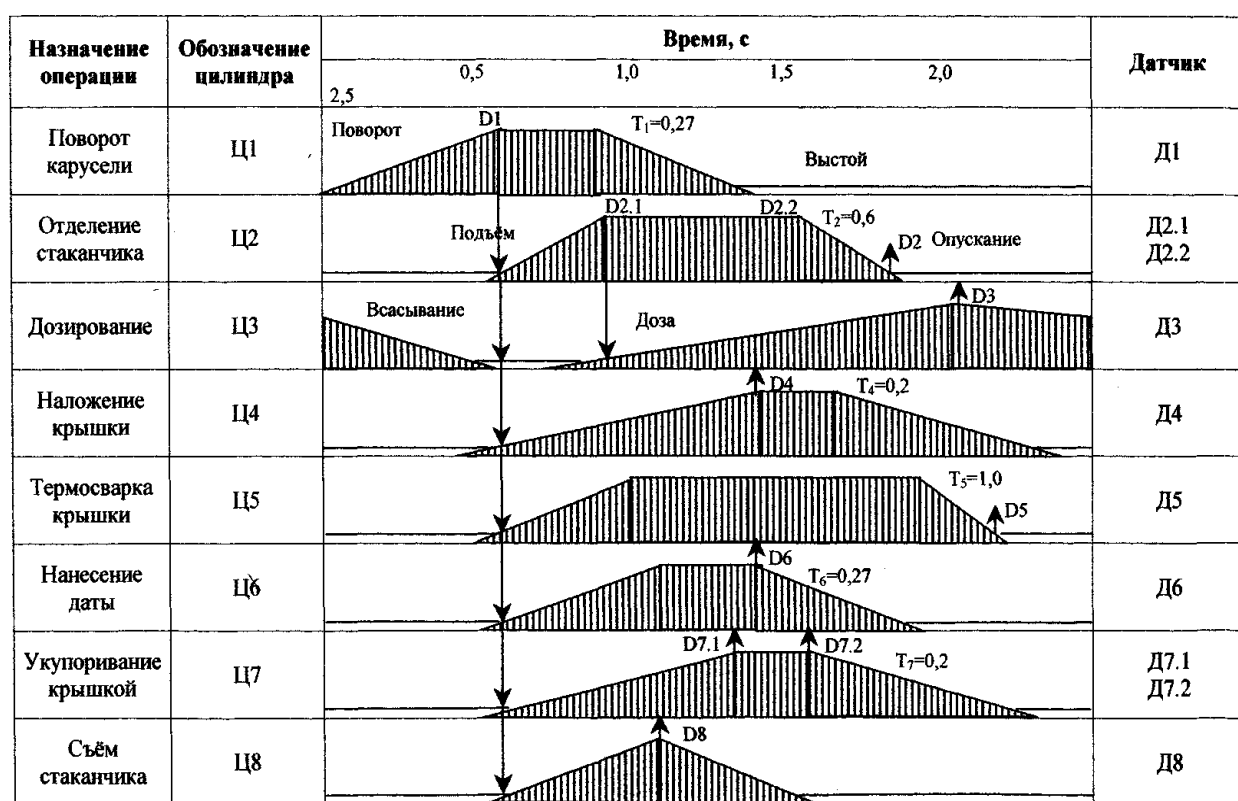


Рис. 1 Циклограмма работы исполнительных механизмов

Для построения сложных автоматов удобно применять иерархическое описание, что позволяет получить представление об общих принципах работы системы, а затем перейти к детальному анализу конкретных аспектов. При этом понимание работы системы будет происходить сверху вниз по иерархии описания.

Сначала пользователь знакомится с основными компонентами описываемой части модели и принципами их взаимодействия, и далее переходит к углубленному и более детальному описанию интересующего его компонента.

Иерархический граф нижнего уровня автомата, управляющего первым циклом возрастающего фронта технологических операций, приведен на рис. 2.

В состояниях контроля S1, S3 и S5 сравнивается текущее модельное время  $T_m$  с заданным значением времени, в которое выбранный исполнительный механизм должен перейти в конечное положение. Положения механизмов фиксирует множество датчиков  $D = \{d_1, d_2, \dots, d_k\}$ . Если при  $T_m = T_{d_i}$  i-ый механизм не достиг конечного положения, то формируется флаг ошибки, например, Error = 'A'. Каждому исполнительному механизму соответствует специально установленное значение флага ошибки, например, взятое из символов латинского алфавита.

При фиксировании любой неисправности цифровой автомат переходит в состояние анализа отказов и по значению флага ошибки можно легко иден-

тифицировать отказавший механизм, что уменьшает трудоемкость работ по восстановлению исполнительных механизмов.

Графы автоматов нижнего уровня объединяются в общий иерархический граф верхнего уровня и отображают весь процесс управления технологическими операциями для выбранного сценария.

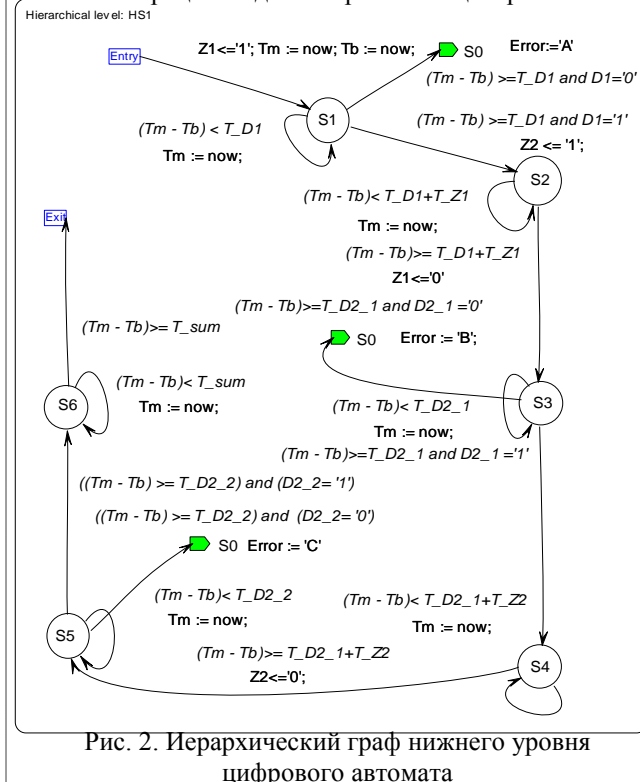


Рис. 2. Иерархический граф нижнего уровня цифрового автомата

## 2. Структура комплекса для синтеза и верификации моделей цифровых автоматов

Для автоматизированного синтеза алгоритмов и программ управления исполнительными механизмами ФУМ в среде Active-HDL разработан программный комплекс, который может использоваться для выбора оптимальных алгоритмов и программ, обеспечивающих повышение производительности расфасовки и упаковки продукции, улучшающих качество продукции и уменьшающих технологические потери фасуемых компонентов.

Структура файлов комплекса приведена на рис. 3.

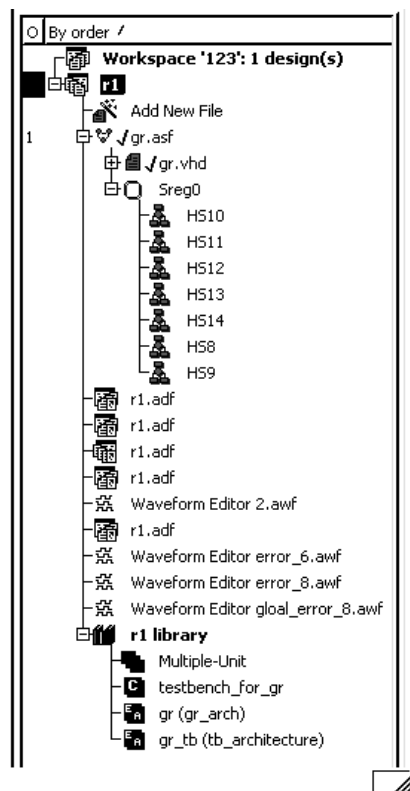


Рис. 3. Структура файлов комплекса

Основными функциями программного комплекса являются: синтез алгоритмов, представленных в виде графов состояний цифровых автоматов с конечным числом состояний, анализ ошибок вносимых пользователем, оптимизация алгоритмов, автоматическая компиляция алгоритмов в программы автоматизированного управления исполнительными механизмами на язык программирования VHDL, обработка файлов с алгоритмами, реализующими многосценарное управление.

Комплекс может функционировать в следующих режимах:

1) синтеза алгоритмов управления исполнительными механизмами ФУМ, данный режим является основным, в нем пользователь может разработать необходимый граф состояний цифрового авто-

мата, обеспечивающий строгое соблюдение последовательности выполнения следующих операций:

- активизации механизмов в заданные моменты времени;
- контроля достижения механизмами конечных положений;
- задания продолжительности включенных состояний механизмов и последовательности их выключения;

2) объединения отдельных алгоритмов в сценарии управления восходящим и нисходящим фронтами технологических операций путем применения иерархического представления проекта;

3) компиляции алгоритмов в программу на языке VHDL, в этом режиме пользователю доступны средства компиляции фирменного пакета Active-HDL, с помощью последнего можно обнаруживать и устранять ошибки в алгоритмах, создаваемые пользователем;

4) разработки стимулирующих воздействий (TestBench), определяющих параметры управляющих сигналов и имитирующих срабатывание датчиков, фиксирующих достижение исполнительными механизмами конечных положений;

5) верификации скомпилированных программ управления исполнительными механизмами, данный режим необходим для контроля соответствия причинно-следственных связей между исполнительными механизмами данным, указанным в циклограммах ФУМ с перестраиваемой структурой, изменяя параметры управляющих сигналов и моменты срабатывания датчиков, пользователь получает возможность объективно оценивать возникающую ситуацию;

6) имитации отказов механизмов или датчиков, контролируемых их положения, что обеспечивает прогнозирование последствий неисправностей технологического оборудования и исключает аварийные ситуации уже на этапе отладки алгоритмов, а также снижает трудоемкость поиска и устранения неисправностей во время эксплуатации ФУМ.

Выходными данными программного комплекса являются откомпилированные программы управления исполнительными механизмами, тексты стимулирующих воздействий (TestBench) на языке программирования VHDL, а также временные диаграммы, отображающие состояние процесса управления и ошибки, возникающие при появлении неисправностей компонентов ФУМ. Временные характеристики программного комплекса зависят от сложности синтезируемых алгоритмов и быстродействия ПЭВМ, на которой они синтезируются, компилируются и верифицируются. В среднем, процесс синтеза алгоритма продолжается несколько десятков минут, хотя в некоторых случаях может потребоваться более продолжительный отрезок времени. На

выполнение процесса компиляции и генерации VHDL кода необходимо затратить несколько секунд для любого синтезируемого алгоритма. В отличие же от длительности синтеза, компиляции алгоритма и генерации VHDL кода, процесс разработки стимулирующих воздействий может потребовать значительное время на свое осуществление. Длительность выполнения, прежде всего, зависит от сложности синтезируемого алгоритма, от количества механизмов и датчиков, контролирующих их положения.

Рассматриваемый программный комплекс является Windows-приложением и для своего нормального функционирования требует как минимум такой конфигурации аппаратных средств, которая обеспечивает функционирование дисковой операционной системы Windows XP. Рекомендуемый объем оперативной памяти 256 Мбайт, графическая подсистема, поддерживающая VGA-режимы. Минимальный размер свободного пространства на жестком диске, необходимый для запуска системы – 2 Гбайт. Объем дискового пространства, необходимый для нормального функционирования программного комплекса зависит от размера, синтезируемого алгоритма, откомпилированных файлов программ на языке VHDL. Для нормальной работы системы при любых значениях перечисленных факторов рекомендуется иметь на жестком диске до 200 Мбайт свободного пространства. Необходим также манипулятор мышь. Для выполнения программы необходимо наличие операционной системы Windows XP или новых версий. Обязательными условиями нормального функционирования комплекса являются наличие среды автоматизированного проектирования цифровых систем Active-HDL версии 7.1 и выше, которая может занимать до 1.4 Гбайт дискового пространства.

## Выводы

Предлагаемый программный комплекс обеспечивает автоматизацию синтеза моделей цифровых автоматов и учитывает реальные динамические характеристики исполнительных механизмов, что увеличивает точность задания временных параметров управляющих сигналов.

Используя стимуляторы, можно быстро устанавливать образцы сигналов, которые необходимы для быстрой проверки моделей. Они позволяют в интерактивном режиме изменять значение любого порта или сигнала, чтобы создать желательные условия для синтезируемой модели.

## Список литературы

1. Антик М.И. Синхронные цифровые автоматы: учебное пособие / М.И. Антик. – М.: МИРЭА, 2004. – 100 с.
2. Шувалова Л.А. Применение модели конечного автомата для синтеза систем управления технологическими процессами / Л.А. Шувалова // Питання прикладної математики і математичного моделювання: зб. наук. пр. – Дніпропетровськ: ДНУ, 2006. – С. 261-267.
3. Дубинец Л.А. Принципы проектирования адаптивных фасовочно-укупорочных машин для молочной и плодомолочной промышленности / Л.А. Дубинец // Вимірювальна та обчислювальна техніка в технологічних процесах. – 2002. – № 2. – С. 132-136.
4. Семенец В.В. Проектирование цифровых систем с использованием языка VHDL / В.В. Семенец, И.В. Хаханова, В.И. Хаханов. – Х.: ХНТУРЕ, 2003. – 492 с.

Поступила в редколлегию 22.12.2009

Рецензент д-р техн. наук, проф. С.М. Первунинский, Черкасский государственный технологический университет, Черкасы.

## СТРУКТУРА ПРОГРАМНОГО КОМПЛЕКСУ СИНТЕЗУ І ВЕРИФІКАЦІЇ МОДЕЛЕЙ ЦИФРОВИХ АВТОМАТІВ

Л.А. Шувалова, Д.Н. Моамар, Т.Ю. Уткіна

*У статті розглянуто призначення та умови застосування програмного забезпечення, що входить до складу програмного комплексу; призначеного для синтезу алгоритмів автоматизованого управління фасувально-пакувальними машинами. Детально висвітлені методи і засоби лінгвістичної компіляції, описані типи використовуваних ними даних. Дано опис режимів, в яких може застосовуватися або модифікуватися комплекс для розширення його функцій.*

**Ключові слова:** програмний комплекс, верифікація моделей, цифровий автомат.

## STRUCTURE OF PROGRAMMATIC COMPLEX OF SYNTHESIS AND VERIFICATION OF MODELS OF DIGITAL AUTOMATS

L.A. Shuvalova, D.N. Moamar, T.Yu. Utkina

*In the article the purpose and terms of using software entering in the complement of programmatic complex intended for the synthesis of algorithms of the automated management of the packing machines are considered. The methods and facilities of the linguistic compilation are explained in detail; the types using by them data are described. The description of the modes in which can be used or modified the complex for expansion its functions is given.*

**Keywords:** programmatic complex, verification of models, digital automat.