

ВВЕДЕНИЕ

В книге рассматриваются вопросы алгоритмизации и программирования задач логического (основанного на истинности и ложности) управления. Предлагается технология алгоритмизации и программирования указанного класса задач на основе теории автоматов. При этом центральным понятием разрабатываемого подхода является понятие "состояния".

Рассматриваются вопросы проектирования систем логического управления и управляющих автоматов. Обоснован выбор графов переходов в качестве языка спецификаций. Изложены вопросы о реализации алгоритмов системой взаимосвязанных графов переходов, их декомпозиции и композиции. Рассмотрены вопросы о взаимодействии головного и вызываемых графов, вложении и иерархии графов, реализации параллельных процессов.

Выполнено сравнение с другими языками спецификаций. Исследованы различные структурные модели автоматов и функциональных элементов задержки. Рассмотрены вопросы кодирования состояний автоматов. Описаны различные алгоритмические и программные модели автоматов и методы преобразования этих моделей.

Обоснован выбор многозначного кодирования состояний и конструкций, аналогичных конструкции switch (switch (англ.) - переключатель) языка СИ, для программной реализации автоматов. Изложен вопрос о программной реализации функциональных элементов задержки и моделей объектов управления.

Использование тетрады (состояние - система взаимосвязанных графов переходов - многозначное кодирование - конструкция switch), а также независимость от глубокой предыстории (в дальнейшем "предыстории" - "будущее зависит от настоящего и не зависит от прошлого") обеспечивает наглядность, структурность, вызываемость, вложенность, иерархичность, управляемость и наблюдаемость программ, а также их изоморфизм (изобразительную эквивалентность) со спецификацией, по которой они формально строятся. Это позволяет Заказчику, Технологию (Проектанту), Разработчику, Программисту, Оператору (Пользователю) и Контролеру однозначно понимать друг друга и точно знать, что должно быть сделано, что делается и что сделано в программно реализуемом проекте, т.е. решить для рассматриваемого класса задач проблему их взаимопонимания. Это позволяет также разделять работу и ответственность, легко и корректно вносить изменения в алгоритмы и программы. Изложен подход к автоматизации программирования в рамках предлагаемой технологии. Рассматриваются вопросы верификации, сертификации и тестирования.

Описываются методы программирования рассматриваемого класса задач в базе языков высокого уровня, языков инструкций, ассемблеров, функциональных и лестничных схем, а также на языке "Графсет", широко применяемом в программируемых логических контроллерах.

В связи с использованием языков программирования высокого уровня, применяемых, например, для промышленных (управляющих) компьютеров, излагаемый подход назван SWITCH-технологией, являющейся одним из направлений широко развиваемой в настоящее время CASE - технологии (Computer Aided Software Engineering). Предлагаемая технология может быть также названа STATE-технология (state (англ.) - состояние),

AUTOMATON-технология (automaton (англ.) - автомат). Показана связь предлагаемого подхода со структурным, событийно-управляемым и объектно-ориентированным программированием и проектированием.

Введены понятия "автоматное программирование" и "автоматное проектирование".

Излагаемая методология, по мнению автора, может рассматриваться как новое научное направление в теории автоматов и в теории и практике программирования по крайней мере для задач логического управления.

Предложенная технология может применяться при разработке и программной реализации алгоритмов логического управления объектами, используемыми в различных отраслях промышленности.

Работа выполнена в научно-производственном объединении (НПО) "Аврора" (Россия, Санкт-Петербург).

Автор выражает благодарность профессорам Д.А.Поспелову, В.Н.Вагину и Г.В.Козлову, помощь которых позволила опубликовать эту книгу.

Автор глубоко признателен Российскому фонду фундаментальных исследований, обеспечившему финансирование настоящего издания.