

**КИБЕРФИЗИЧЕСКИЕ СИСТЕМЫ: ЛИНГВИСТИЧЕСКИЙ АСПЕКТ**

**Аннотация.** Предложен подход к проектированию систем управления техническими объектами, особенностью которого является лингвистически управляемое проектирование. Представлена разработанная на основе этого подхода методика проектирования систем управления, предназначенная для использования на стадии проектирования, которая предшествует построению формальной модели.

**Ключевые слова:** система управления, лингвистически управляемое проектирование, методика проектирования систем управления, киберфизическая система.

**ВВЕДЕНИЕ**

Создание киберфизических систем (КФС), т.е. систем, в которых взаимодействуют физические объекты и компьютерные программы [1, 2], включает проектирование и разработку управляющих составляющих (автоматических или программных).

Отметим несколько значимых аспектов создания систем и их функционирования.

1. Очень важен первоначальный шаг проектирования системы, связанный со сбором, анализом и разработкой требований к ней. Негативные последствия упущений при выполнении этого шага (затраты на переделки, уязвимости готового продукта и др.) отмечали в разное время разные авторы (см., например, [3–5]).

Сформировалась специальная область – инженерия требований [6, 7], в которой рассматриваются методы перехода от требований заказчика, как правило, представленных на естественных языках, к модели, а затем и к действующей системе, удовлетворяющей пожеланиям заказчика. Однако потребность в разработке, развитии и совершенствовании методов и средств работы с требованиями все еще существует.

2. Жизненный цикл системы (период от ее проектирования до вывода из эксплуатации) включает ее сопровождение, а иногда и модификацию. Наличие документации с обоснованием выборов и решений, сделанных и принятых на этапе разработки системы, полезно, когда требуется изменить существующую систему (улучшить её рабочие характеристики или наделить новыми возможностями, не лишив уже имеющихся преимуществ).

3. При создании систем в настоящее время используются формальные методы верификации и валидации [8, 9], что требует построения формальной модели проектируемой системы. Первоначально требования к системе, как правило, формулируются на естественном языке (ЕЯ). Переход от этих требований к формальной модели системы — это отдельная задача, создание методов и средств решения которой выделилось в специальное направление исследований: отслеживание требований [10, 11].

Целесообразно так вести процесс проектирования, чтобы уже на его ранних стадиях создавалась модель проектируемого объекта, которая верифицируется в ходе построения, сравнительно легко формализуется на последующих этапах проектирования и является основой для создания документации (пригодной для использования при модификации, реинжиниринге).

В данной работе предлагается подход лингвистически управляемой разработки требований для выполнения начальной стадии проектирования такой части КФС как система управления (СУ). Этот подход предполагает:

- сбор и формирование набора требований к системе на ЕЯ;
- построение различного вида моделей СУ и ее составляющих на ЕЯ на основе использования языковых смысловых связей;
- фиксирование и обоснование введения в рассмотрение сущностей и решений, принятых в ходе формирования требований и построения моделей;
- формирование рабочей ЕЯ-лексики как связующего звена между этапами разработки требований, с одной стороны, и построения формальной модели системы и документирования, с другой.

Воплощением этого подхода в настоящее время является методика проектирования СУ техническим объектом (объектами), которая:

- обеспечивает (частичную) верификацию уже на первых этапах проектирования (которые осуществляются в рамках ЕЯ);
- базируется на понятиях из предметной области «системы управления»;
- дает возможность подготовить (промежуточный) текст проекта, автоматически (автоматизированно) преобразующийся к формальному виду и являющийся основой для создания документации, пригодной для использования при реинжиниринге.

В данной работе развита намеченная в [12] идея упорядоченного поэтапного построения представленных на ЕЯ функциональных спецификаций проектируемой системы.

Изложение материала организовано следующим образом. В разд. 1 дано общее описание подхода к проектированию СУ, сформулированы принципы подхода. В разд. 2 приведена методика проектирования СУ, развитая на основе предложенного подхода. Она охватывает те этапы проектирования, на которых основным языком представления данных и знаний является ЕЯ. Приведена краткая иллюстрация выполнения этапов проектирования согласно методике. В заключении обобщаются итоги работы, дальнейшие задачи и возможные направления развития.

## 1. ОБЩИЙ ПОДХОД К ПОСТРОЕНИЮ МЕТОДИКИ ПРОЕКТИРОВАНИЯ СУ

Предположим, у разработчика проекта СУ (Разработчика) имеется либо некоторый исходный текст требований (к готовой системе), либо краткая формулировка задания. Назовем это первоначальным текстом (ПТ).

Разработчик на основе ПТ создает спецификацию системы управления объектом на ЕЯ, которая предназначается для использования при верификации, валидации разрабатываемой СУ, а также как часть документации программы (программной системы), реализующей данную СУ.

Чтобы осуществлять верификацию, валидацию разрабатываемой СУ, требуется сначала преобразовать спецификацию этой системы к некоторому формальному виду. Поэтому целесообразно спецификацию писать на языке Разработчика (**ЕЯ предметной области с использованием устойчивых словосочетаний**) так, чтобы последующую формализацию этой спецификации можно было осуществить автоматически или автоматизированно.

Принципы предлагаемого подхода к проектированию СУ следующие.

**1.1. Многоаспектное моделирование.** ЕЯ-спецификации СУ объектом создаются на языке Разработчика поэтапно, для чего строится ряд моделей как самого объекта управления, так и его СУ, которые отражают различные аспекты управления (функциональный, структурный, поведенческий). Эти модели уточняются в ходе проектирования.

**Функциональная модель** описывает совокупность функций СУ. Отражает такие аспекты: управление объектом разными категориями субъектов, а также управление объектом в нормальных условиях и в аварийных ситуациях.

**Структурная модель** описывает подсистемы СУ. При ее построении анализируется структура объекта управления, и выделяются те его компоненты, для которых требуются подсистемы управления.

**Информационная модель** описывает потоки данных между компонентами объекта управления и подсистемами СУ. Для ее построения выделяются состояния компонентов объекта управления и вводятся их атрибуты (признаки, характеристики, существенные для построения формальной модели СУ).

**Модель поведения** описывает взаимодействие СУ и объекта управления, а также взаимосвязи подсистем СУ. Для ее построения выделяются процессы, в которых задействованы объект управления и его компоненты, ситуации протекания процессов и виды переходов из состояния в состояние, определяются стратегии управления.

**1.2. Обоснованность решений.** Принимаемые в ходе проектирования решения, вносимые изменения и уточнения фиксируются и обосновываются (т.е. объясняются и аргументируются). Сохраняется возможность при необходимости вернуться и изменить ранее принятые решения, объяснив причины изменения и обосновав новые решения.

**1.3. Полнота построений.** При введении множеств состояний средств управления, объекта управления или компонентов объекта, а также множеств ситуаций протекания управляемого процесса и видов переходов из состояния в состояние обосновывается полнота каждого множества. (Множества состояний различного вида и ситуации возникают при создании моделей разных типов (подразд. 1.1).)

**1.4. Формирование рабочей лексики.** Введение атрибутов при построении информационной модели сопровождается пояснением их смысла. Для этого используются фразы ЕЯ. Набор таких фраз образует рабочую лексику (набор клише). При составлении детальной модели СУ и ее подсистем используются клише. Они являются связующим звеном при переходе от ЕЯ-модели к формальной модели. (О происхождении и использовании клише при проектировании систем см. [12, 13].)

**1.5. Лингвистически управляемое проектирование.** План построения и детализации моделей (упомянутых ранее) основан на систематическом использовании ряда существенных (в области проектирования СУ) понятий.

Далее приведен перечень сущностей и связанных с ними понятий, которые следует принимать во внимание при разработке СУ некоторым объектом.

Объясним причины выделения этих сущностей, показывая, как используемый язык «подсказывает», на что необходимо обратить внимание (здесь описывается лингвистический аспект предлагаемого подхода).

Исходим из понимания (носителями языка) смысла слов **объект и управление**, а также учитываем, что управление всегда осуществляется с определенной целью. Особо отметим, что при формировании приводимого далее перечня ставилась цель объяснить, почему та или иная сущность включается в него, а не дать определение понятия, соответствующего этой сущности.

**Объект управления.** Эту сущность проигнорировать нельзя, поскольку разрабатывается **система управления**. Управление осуществляется некоторым объектом (иногда разрабатываемым наряду с СУ). Следовательно, разрабатывая СУ каким-либо объектом, следует определить (назвать, объяснить, описать, сослаться на готовое описание), что это за объект.

**Функция объекта управления.** Проектируя СУ некоторым объектом, необходимо учитывать, что именно делает объект управления, какую функцию он выполняет.

**Предназначение объекта управления.** Объект управления создан (или проектируется вместе с СУ) с определенной целью, поэтому при разработке СУ этим объектом необходимо уяснить, зачем он нужен.

**Категории пользователей объектом.** Поскольку объект управления выполняет некоторую функцию, следует выяснить, для кого (для чего) нужны объект и результаты его функционирования.

**Среда объекта управления.** Объект управления находится в некоторой среде. Взаимовлияния этой среды и объекта управления следует учитывать при проектировании СУ. Пользователи также являются частью среды.

**Субъекты управления объектом и их категории.** Если имеется объект управления, то следует выяснить, что (кто) есть субъекты управления и их количество. Субъекты управления также являются частью среды объекта управления.

**Компоненты объекта.** Поскольку объект может быть таким, что для некоторых его частей понадобятся отдельные СУ, следует проанализировать структуру объекта и выяснить, повлияет ли она (и как именно) на разрабатываемую СУ.

**Система управления объектом.** Это то, что разрабатывается (нельзя обойти молчанием). Следует уяснить, какова роль СУ в выполнении функции объекта управления, а также является ли СУ простой или составной.

**Подсистемы управления.** Если для некоторых составляющих объекта управления необходимы отдельные СУ, то и проектируемая СУ становится составной, включающей соответствующие подсистемы.

**Нормальные условия** (функционирования объекта, СУ). Следует определить и описать, при каких обстоятельствах функционирование объекта и СУ безопасно для объекта управления и пользователей.

**Аварийные ситуации** (функционирования объекта, СУ). Следует определить и описать, при каких условиях функционирование объекта, СУ опасно для объекта управления или пользователей.

**Состояния** (средств управления, объекта управления, его составляющих, среды объекта управления). Поскольку проверки (верификация, тестирование и т.д.) разрабатываемой СУ проводятся на этапе проектирования, требуется построить модель поведения объекта управления, СУ, а для этого необходимо знать множество состояний объекта управления, СУ.

**Полнота множества состояний.** Обосновывать полноту множества состояний (объекта управления, СУ) целесообразно для повышения качества модели поведения (объекта управления, СУ). Если в модели не учтены какие-либо состояния объекта управления или СУ, то модель будет неполной. Множество  $S$  состояний (объекта  $X$ ) называем полным, если каждое состояние объекта  $X$  входит в  $S$  и каждый элемент из  $S$  является состоянием объекта  $X$ .

**Виды переходов из состояния в состояние.** При изучении или моделировании составных объектов приходится осуществлять их декомпозицию. Каждый компонент имеет свое множество состояний. Состояние объекта или его компонента наблюдается в определенных условиях (в нормальных условиях или в аварийной ситуации), и это следует учитывать.

**Полнота множества переходов.** Обосновывать полноту множества переходов (объекта управления, СУ) из состояния в состояние целесообразно для повышения качества модели поведения (объекта управления, СУ). Если не учтены какие-либо переходы, то для части состояний поведение объекта, СУ не будет описано, а модель будет попадать в тупики.

**Детерминизм переходов.** Это важное свойство поведения объекта управления, СУ. Если желательно детерминированное поведение СУ, то это следует обеспечить при проектировании. При разработке СУ некоторым объектом необходимо знать, является ли поведение объекта детерминированным.

**Стратегии управления объектом или компонентом объекта.** Разработка стратегии управления объектом — важный этап проектирования СУ. Управление объектом преследует определенную цель. При проектировании СУ приходится разрабатывать и общий план достижения этой цели (т.е. стратегию управления).

**Обоснование полноты и корректности стратегии.** При разработке (или использовании имеющейся) стратегии важно исследовать (или знать) ее свойства: полноту и корректность.

**Режим управления.** Условия, в которых осуществляется управление, могут изменяться, поэтому следует выяснить, какое управление требуется для тех или иных условий.

**Процессы.** СУ разрабатывается для того, чтобы регулировать и направлять процессы, в которых задействован объект управления, поэтому необходимо выяснить, что это за процессы.

**Начало процесса, протекание процесса между началом и завершением, завершение процесса.** Это существенные фазы процесса. Следует учитывать их особенности.

**Ситуации протекания процесса.** Процесс протекает в определенных условиях. Наборы этих условий (ситуации) могут быть различными, и следует выяснить, есть ли необходимость в специальном управлении в каждой ситуации.

**Полнота множества ситуаций.** Обосновывать полноту множества ситуаций протекания процесса целесообразно для повышения качества модели поведения (объекта управления, СУ). Если не учитывать ситуации, в которых требуется специальное управление, СУ будет неполной.

**Сценарии.** Взаимодействия подсистем управления в составных СУ, а также взаимодействие пользователя (субъекта управления) с управляемым объектом задаются в виде сценариев.

**Средства наблюдения за объектом управления.** Чтобы управлять и пользоваться объектом, необходимо иметь возможность узнавать о его состоянии, поэтому нужны соответствующие средства.

**Датчики, таймеры, измерительные приборы и т.п.** Это известные средства наблюдения за объектами. Ассортимент средств наблюдения может расширяться.

**Синхронизация процессов.** В системах с несколькими процессами могут понадобиться средства их синхронизации.

**Средства управления объектом или компонентом объекта.** Чтобы оказывать управляющие воздействия на объект или его составляющие, необходимы соответствующие средства.

**Кнопки, тумблеры, переключатели и т.п.** Это известные средства управления объектами. Состав средств управления может расширяться.

**Информационные средства.** Чтобы получать сведения об управляемом (наблюдаемом) объекте непосредственно или узнавать результаты наблюдения за ним, требуются сигнальные (информационные) средства.

**Табло, индикаторы и т.п.** Это известные информационные средства, позволяющие узнавать о состоянии управляемого (наблюдаемого) объекта или его составляющих. Состав информационных средств может расширяться.

**Управляющие воздействия и их результаты.** Средства управления предназначены для воздействия на объект управления. Предполагается, что результаты этих действий известны при разработке СУ. Их следует проверять для готовой модели СУ.

**Средства интерфейса.** Для связи между компонентами составных СУ, а также для связи субъектов управления с СУ нужны соответствующие средства.

**Сообщения, признаки и т.п.** Это известные средства, с помощью которых осуществляется интерфейс. Состав средств интерфейса может расширяться.

**Количественные и качественные характеристики (разрабатываемой СУ).** При разработке любой системы рассматриваются нефункциональные требования к ее качеству и количественные показатели ее работы и эксплуатации.

**Количественные и качественные характеристики разработки (СУ).** Разработка системы (в том числе и СУ) осуществляется в рамках количественных характеристик (сроки, затраты и т.д.), а также с учетом качественных требований и ограничений.

Определение каждой сущности и каждого понятия из приведенного перечня либо задано наперед (уже известно), либо формулируется в ходе разработки СУ. Важно, чтобы было известно, что означает то или иное используемое при проектировании СУ понятие (термин).

**1.6. Согласование частей.** Общая модель СУ строится на основе разработанных отдельных моделей с использованием рабочей лексики, которая включает как отдельные слова, так и устойчивые многословные фразы (клише). Модель СУ тесно связана с моделью объекта управления. Если объект сложный (составной), то и СУ имеет несколько подсистем, работу которых надо согласовывать.

## 2. МЕТОДИКА ПРОЕКТИРОВАНИЯ СУ ТЕХНИЧЕСКИМ ОБЪЕКТОМ

**2.1. Общий план построения методика.** Опишем, как на основе изложенных принципов строится методика проектирования систем управления техническими объектами. Начинать проектирование СУ естественно с выяснения того, что следует разработать и с какой целью, что собой представляет объект управления, выяснения того, каковы требования заказчика и его видение функционирования готовой СУ, а также какие ресурсы доступны для ее разработки.

Затем можно приступать к разработке ряда моделей различных степеней детализации (многоаспектное моделирование). Разработка моделей осуществляется на основе принципа лингвистически управляемого проектирования: план построения моделей определяется перечнем сущностей и связанных с ними понятий, приведенном в разд. 1, подразд. 1.5. Построение моделей сопровождается введением в рассмотрение требуемых для их описания множеств (например, множеств составляющих и состояний (чего-либо) и т.п.). Согласно методике следует предоставлять обоснования полноты вводимых множеств (принцип полноты построений). По существу, обоснование полноты множества состоит в том, чтобы показать, что при его построении ничего не упущено и не попало ничего «лишнего».

Методика состоит из ряда этапов, каждый из них содержит задания, которые следует выполнить. Если выполнение задания приходится отложить (например, вследствие необходимости уточнений) и вернуться к нему в дальнейшем, то оно попадает в специальный список. На каждом этапе методика допускается внесение уточнений.

Этап имеет такой формат:

- номер;
- название, характеризующее род деятельности Разработчика на данном этапе;
- перечень заданий для Разработчика;

— пункт об уточнениях, связанных с данным этапом, возникающих после выполнения заданий этого этапа (контролирует выполнение необходимых корректировок), обеспечивает возможность возврата к данному этапу, когда уточнения возникают во время выполнения других этапов;

— пункт об отложенных работах (контролирует выполнение всех заданий данного этапа, а также новых задач, которые возникли при выполнении заданий этого этапа).

Элемент перечня заданий имеет номер, название, формулировку задания и/или вопросы, ответы на которые требуются для его выполнения, а также может содержать пояснения к заданию.

Задания для Разработчика сформированы в соответствии со списком сущностей и связанных с ними понятий (см. разд. 1, подразд. 1.5). Таким образом, методика построена на лингвистической основе. Остальные аспекты методики определяются принципами: обоснование сделанных выборов и принятых решений, формирование рабочей лексики, согласование частей.

На этапе 1 работы по методике формулируется постановка задачи. На этапах 2–5 подготавливаются данные, необходимые для построения модели проектируемой СУ. На этапах 6–8 строятся модели СУ. На этапе 9 обеспечивается взаимодействие составляющих СУ. На этапе 10 завершается стадия проектирования СУ, предшествующая формализации.

**2.2. Методика.** Далее приводится методика проектирования СУ техническим объектом.

**Этап 1.** Уяснение задачи разработки.

1. Формулирование задания и цели разработки. Выяснить и сформулировать, что следует разработать (какую именно СУ), какую функцию должна выполнять проектируемая СУ.

В ходе разработки проекта следовать правилу: для всех вводимых в рассмотрение сущностей приводить толкования, смысл каждого используемого понятия (термина) либо известен, либо определяется.

2. Объект управления. Что понимается под объектом (классом объектов) управления? На этом шаге разработки проекта следует сформулировать, что именно является объектом управления для разрабатываемой СУ. Что делает и для чего нужен объект управления?

3. Вид системы управления и ее пользователи. Какой характер общности разрабатываемой СУ? Кто будет пользоваться СУ (и управляемым ею объектом)? Следует уточнить, насколько общий характер должна иметь разрабатываемая СУ (например, разрабатывается ли она для управления уникальным объектом, классом серийных или похожих объектов либо для других целей). От этой информации может зависеть выбор формализма для построения формальной модели поведения (или в какую форму будут преобразованы функциональные спецификации (ФС), описывающие поведение СУ на ЕЯ). Выбор формализма можно отложить и на последующие стадии проектирования.

4. Ограничения и приоритеты. Каковы желаемые характеристики разрабатываемой СУ? (Например, стоимость, не превышающая определенной величины, скорость работы, производительность и др.) Каковы количественные и качественные характеристики разработки СУ? Следует составить перечень количественных и качественных показателей, которым должна соответствовать готовая СУ, а также выяснить, какие ресурсы и средства требуются и доступны для ее разработки. Следует выяснить и привести перечень неизменных свойств готовой СУ. (Например, безопасность, надежность, беступиковость и др.) Следует выяснить и сформулировать приоритетность обеспечения характеристик и

свойств разрабатываемой СУ. (Например, в первую очередь, обеспечивается безопасность (указывается, что понимается под безопасностью), во вторую очередь обеспечивается удобство пользования (указывается, что считать удобством пользования) и т.д. для каждого желаемого свойства.)

5. Сценарии пользователя. Следует разработать и описать ориентированные на пользователя сценарии его взаимодействия с управляемым объектом и СУ.

6. Уточнения, касающиеся объекта управления, вида системы управления, ограничений, свойств и приоритетов, сценариев пользователя. Внести (если требуется) уточнения в данные, собранные при выполнении пп. 1–5. Обосновать и зафиксировать принятые решения. Обновить описания объекта управления, вида системы управления, ограничений и приоритетов, сценариев пользователя согласно внесенным изменениям.

**Замечание 1.** Требование обоснования принимаемых решений поддерживает «верификационный» стиль проектирования.

7. Требуемые уточнения и/или отложенные работы. Следует проверить, завершено ли выполнение пп. 1–6, зафиксировать требующиеся и еще не сделанные уточнения, отложенные работы (если такие имеются).

Для данного этапа методики и последующих отметим, что если уточнений и отложенных работ на каком-либо этапе применения методики нет, то констатируется их отсутствие, иначе следует перечислить требуемые уточнения и/или отложенные работы.

**Этап 2.** Объект управления и среда его функционирования.

1. Структурная модель объекта управления. Проанализировать структуру объекта управления. Выделить его составляющие и для каждой из них привести определение (толкование). Какой принцип положен в основу выделения составляющих? Имеются ли такие составляющие объекта управления, для которых требуется разработать подсистемы управления как части всей проектируемой СУ?

2. Полнота набора составляющих объекта управления. Дать обоснование.

3. Среда функционирования объекта управления. Проанализировать структуру среды объекта управления. Выделить ее составляющие. Какой принцип положен в основу выделения составляющих? Что представляет собой каждая из них?

4. Полнота набора составляющих среды функционирования объекта управления. Дать обоснование.

5. Уточнения, касающиеся составляющих объекта управления и среды функционирования объекта управления. Внести, если необходимо, соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания составляющих объекта управления и среды его функционирования.

6. Требуемые уточнения и/или отложенные работы.

**Этап 3.** Субъекты управления и режимы управления.

1. Субъекты управления. Проанализировать состав совокупности субъектов управления и выяснить, каковы их виды. Сколько видов субъектов управления? Следует провести классификацию субъектов управления и дать названия выделенным видам (например, субъект управления 1, ..., субъект управления  $n$  ( $n$  — количество выделенных видов субъектов управления)).

2. Понятие субъекта (субъектов) управления. Следует сформулировать, что представляет собой субъект управления 1, ..., субъект управления  $n$ .

3. Режимы управления. Стандартными режимами управления будем называть управление объектом: в нормальных условиях, в ненормальных, но не опасных условиях, в опасных условиях. И ненормальные, и опасные условия будем считать аварийной ситуацией. Следует выяснить и сформулировать, что такое для объекта управления и СУ нормальные (ненормальные, но не опасные, опас-

ные) условия функционирования. Какие режимы управления будут доступны каждому из видов субъектов управления? Следует выяснить, субъекты каких видов смогут управлять объектом управления в нормальных (ненормальных, но не опасных, опасных) условиях.

4. Встроенные (автоматические) системы защиты. Выяснить, имеются ли какие-либо встроенные (автоматические) системы (средства) защиты объекта управления, и если таковые существуют, то назвать их и описать. Определить, требуется ли разрабатывать какие-либо системы защиты.

5. Полнота набора функций системы управления. Дать обоснование полноты набора субъектов управления и наборов режимов управления.

6. Уточнения, касающиеся субъектов и режимов управления. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания наборов субъектов и режимов управления согласно уточнениям.

7. Требуемые уточнения и/или отложенные работы.

**Этап 4.** Средства управления объектом управления и наблюдения за ним.

1. Состав средств управления и наблюдения. Определить состав средств управления. Что представляет собой каждое из них? Это какие-либо стандартные средства (кнопки, тумблеры, переключатели и т.д.) или требуется искать либо разрабатывать новые?

Определить состав средств наблюдения за объектом управления. Что представляет собой каждое из них? Это какие-либо доступные стандартные средства (датчики, измерительные приборы, имеющиеся в наличии или в продаже) или требуется разрабатывать новые?

Следует выбрать и зафиксировать, какие средства управления и наблюдения будут использоваться в разрабатываемой СУ, а также обосновать их выбор.

2. Полнота набора средств управления и наблюдения. Дать обоснование.

3. Управляющие воздействия и их результаты. Описать, какие результаты должно обеспечить использование каждого из средств управления.

4. Состояния средств управления. Для каждого вида средств управления определить множество состояний.

5. Полнота множеств состояний. Обосновать полноту построенного множества состояний каждого вида средств управления.

6. Уточнения, касающиеся средств управления и наблюдения. Внести соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания средств управления.

7. Требуемые уточнения и/или отложенные работы.

**Этап 5.** Состояния и процессы, связанные с объектом управления и его средой.

1. Состояния составляющих объекта управления. Определить множество состояний для каждой выделенной составляющей объекта управления. В каких ситуациях наблюдается каждое состояние? Обосновать полноту набора ситуаций. Какие требуются информационные средства наблюдения за состояниями объекта управления? Что представляет собой каждое из них? Это стандартные средства (табло, индикаторы и пр.) или требуется разрабатывать новые?

2. Полнота множеств состояний. Обосновать полноту множества состояний каждой составляющей объекта управления.

3. Процессы, в которых задействованы составляющие объекта управления. Следует выявить и описать эти процессы. Обосновать полноту набора процессов для каждой составляющей объекта управления. Требуется ли средства синхронизации процессов? Каковы ситуации протекания процессов? Обосновать полноту множеств ситуаций.

4. Описание фаз протекания процессов. Выделить фазы протекания каждого процесса и описать связи процессов (их фаз) и управляющих воздействий.

5. Сценарии взаимодействия процессов. Описать соответствующие сценарии.

6. Уточнения, касающиеся множеств состояний и наборов процессов, связанных с составляющими объекта управления. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания множеств состояний и наборов процессов, связанных с составляющими объекта управления.

7. Требуемые уточнения и/или отложенные работы.

**Этап 6.** Структурная модель СУ.

1. Составляющие СУ. Выделить составляющие СУ. Составляющая (подсистема) СУ определяется тройкой (субъект управления, режим управления, составляющая объекта управления).

2. Способы и виды связи между составляющими СУ. Выяснить, какие подсистемы СУ взаимозависимы. Для каких совокупностей подсистем управления требуются средства, которые обеспечивают их согласованную работу? Определить каналы входов и выходов подсистем СУ. Как «общаются» между собой компоненты СУ (с помощью сообщений, «классной доски» или иным способом)? (Это необходимо для установления информационных связей между компонентами СУ.)

3. Уточнения, касающиеся составляющих СУ и видов связи между ними. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания наборов составляющих СУ и видов связи между ними.

4. Требуемые уточнения и/или отложенные работы.

**Этап 7.** Информационная модель СУ.

1. Определение атрибутов СУ. Определить множество глобальных атрибутов и множество атрибутов для каждой подсистемы управления (множества локальных атрибутов). Объяснить смысл каждого атрибута, используя краткие точные недвусмысленные фразы. Совокупность этих фраз образует рабочую лексику.

2. Обоснование выбора множеств глобальных и локальных атрибутов. Объяснить причины выбора атрибутов.

3. Уточнения, касающиеся множеств глобальных атрибутов и множеств локальных атрибутов. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения. Обновить описания атрибутов.

4. Требуемые уточнения и/или отложенные работы.

**Этап 8.** Модель поведения.

1. Виды переходов из состояния в состояние. Описать соответствующие функции переходов, при этом использовать рабочую лексику.

2. Полнота множества видов переходов. Дать обоснование.

3. Стратегии управления. Выбрать и описать стратегии управления для СУ и ее подсистем, а также обосновать соответствие требованиям заказчика, полноту и корректность каждой стратегии. Выяснить, детерминировано ли поведение СУ.

4. Уточнения, касающиеся процессов, ситуаций, видов переходов, стратегий управления. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения и обновить описания.

5. Требуемые уточнения и/или отложенные работы.

**Этап 9.** Интерфейс подсистем управления, а также СУ и субъектов управления.

1. Согласование работы всех подсистем СУ. Проверить, обеспечены ли все средства, которые требуются для согласованной работы подсистем СУ.

2. Обоснование согласованности функционирования всех подсистем СУ. Обосновать, что все подсистемы СУ работают согласованно.

3. Интерфейс СУ и субъектов управления. Проверить, обеспечены ли в достаточной мере информационные средства для получения сведений об объекте управления, необходимые субъектам управления. Разработка интерфейса частично осуществляется в ходе разработки моделей подсистем управления, так как требуется рассматривать входные и выходные сигналы для каждой из подсистем управления.

4. Уточнения, касающиеся интерфейса подсистем управления. Внести (если требуется) соответствующие уточнения. Обосновать и зафиксировать принятые решения и обновить описания.

5. Требуемые уточнения и/или отложенные работы.

**Этап 10.** Завершение стадии проектирования, предшествующей формализации.

1. Сбор и упорядочение уточненных, обоснованных данных, полученных в ходе проектирования СУ. Собрать и упорядочить итоговые данные, выделив постановку задачи, принятые в ходе проектирования решения (с их обоснованием), которые повлияли на окончательный результат. Составить описание спроектированной системы.

2. Подготовка текста документации. Предоставить итоговые материалы по спроектированной СУ в установленной форме.

Представим краткую иллюстрацию выполнения этапов проектирования по описанной методике. Проследим на простом примере, как связаны задания этапов с сущностями и понятиями из перечня разд. 1, подразд. 1.5.

**Этап 1.** Уяснение задачи разработки.

Разработать СУ лифтом для пятиэтажного жилого здания с одним лифтом.

Функция СУ: обеспечить управление лифтом в соответствии с его назначением.

**Объект управления** (исходим из распространенного представления о лифте в жилом здании): устройство с кабиной для пассажиров (грузов), которая имеет дверь и с помощью движущего механизма перемещается между этажами.

**Функция лифта:** доставка пассажиров (грузов) с этажа на этаж данного здания.

**Вид СУ:** для объектов одной серии, т.е. для пятиэтажного здания с одним лифтом. **Пользователи:** люди.

**Приоритетным** при разработке СУ считать безопасность пользователей лифта и здания, в котором он находится.

**Сценарии** взаимодействия пользователей с лифтом (согласовываются с заказчиком и ориентированы на пользователя) приводятся далее.

Пользователь может вызвать лифт с лифтовой площадки каждого из этажей здания. Для этого на каждой площадке имеются две кнопки: ВВЕРХ и ВНИЗ. На самом нижнем этаже кнопка ВНИЗ заблокирована, как и кнопка ВВЕРХ на самом верхнем этаже. Для удобства пользователей на каждой лифтовой площадке имеется световой индикатор текущего положения кабины, показывающий направление ее движения или номер этажа, на котором она стоит.

Любой пользователь, находясь в кабине, может сделать заявку на обслуживание (доставку его на нужный этаж). Для этого в кабине есть панель с пронумерованными кнопками для каждого из этажей здания. Имеется также световой индикатор положения кабины.

Когда кабина останавливается на этаже либо по вызову пользователя, либо для того, чтобы пользователь вышел из кабины, ее двери автоматически открываются на фиксированное время и автоматически закрываются. Если в ходе закрывания встречается помеха, двери снова открываются. Также пользователь, находящийся в кабине, может ускорить закрывание двери, для чего на панели в кабине есть кнопка «Закреть двери».

Если имеются вызовы лифта, когда кабина с одним или несколькими пользователями находится на некотором этаже, она не перегружена и ее дверь закрыта, то лифт начинает движение по вызову. Если кабина перегружена, то лифт с помощью индикатора перегрузки, имеющегося в кабине, сигнализирует об этом и ожидает, пока весовая нагрузка не уменьшится.

Пользователь может рассчитывать, что его вызов, поступивший во время обслуживания другого, более раннего вызова, запоминается и ставится в очередь на обслуживание.

Для удобства пользователя в кабине есть кнопка «Вызов диспетчера».

Рассмотрев сценарии взаимодействия пользователей с лифтом, отметим, что целесообразно динамически переупорядочивать очередь вызовов так, чтобы уменьшить расходы на движение кабины. Зафиксируем это как отложенное задание.

**Этап 2.** Объект управления и среда его функционирования.

**Составляющие лифта:** кабина с дверью, тросы (или другие средства), с помощью которых перемещается кабина, мотор, приводящий в движение средства перемещения кабины, блок питания мотора, оборудование, обеспечивающее связь мотора и средств перемещения кабины, а также кабины и средств ее перемещения.

**Полнота набора составляющих.** Все приведенные составляющие необходимы для работы лифта. О составляющих, не вошедших в представленный набор, следует узнать у эксперта по лифтам данного вида.

Рассмотрим модель **среды функционирования** объекта.

**Составляющие среды:** шахта лифта в пятиэтажном здании; лифтовые площадки (на каждом этаже одна).

Основой для выделения составляющих среды является предназначение лифта: перемещение грузов, пассажиров с этажа на этаж. Для движения лифта необходимо оборудованное соответствующим образом пространство (шахта). На каждом этаже здания должна быть площадка для входа в кабину и выхода из нее.

На лифтовой площадке на уровне межэтажного перекрытия имеется проем, перекрытый дверью. При необходимости следует описать структуру шахты лифта.

**Этап 3.** Субъекты управления и режимы управления.

**Субъекты управления:** пассажиры, обслуживающий персонал.

Пассажиром (пользователем лифта) будем называть того, кто находится в кабине лифта или ожидает его на лифтовой площадке в целях перемещения с этажа на этаж. Представителем обслуживающего персонала (специалистом) назовем члена подразделения обслуживания лифтов.

Поскольку есть два вида субъектов управления и два **стандартных режима управления**, заключаем, что СУ лифтом должна обеспечивать управление движением лифта специалистом (пассажиром) в нормальных условиях и в аварийных ситуациях.

**Нормальные условия** функционирования лифта:

- среда функционирования лифта в нормальном состоянии;
- система питания и мотор лифта работают нормально;
- средства для перемещения кабины и система крепления исправны;
- кабина исправна;
- лифт включен;
- двери кабины и на лифтовых площадках исправны;
- кабина не перегружена;

- движение кабины вверх/вниз осуществляется со скоростью, которая находится в заданных пределах;
- движение кабины происходит с закрытой дверью;
- на всех лифтовых площадках во время движения лифта двери закрыты;
- дверь на лифтовой площадке некоторого этажа открывается, только если кабина лифта стоит на уровне этого этажа;
- пока кабина лифта стоит на уровне некоторого этажа, двери на лифтовых площадках остальных этажей закрыты;
- если кабина лифта стоит на уровне некоторого этажа и ее дверь открыта, то через определенное время она закрывается.

Невыполнение хотя бы одного из перечисленных условий означает, что лифт находится **в аварийной ситуации**.

При составлении перечня нормальных условий работы лифта принято во внимание, что приоритетным требованием является безопасность пользователей, а также учитывалась ранее определенная структура лифта и среды его функционирования. При этом рассматривались среда функционирования лифта, его составляющие, лифт в движении и в покое.

Отметим, что детальные описания условий нормальной работы составляющих лифта и нормального состояния среды либо считаются уже известными (есть соответствующие нормативы), либо будут разрабатываться. Важно, что предложенная методика «побуждает» заниматься этими вопросами.

**Этап 4.** Средства управления объектом управления и наблюдения за ним.

С учетом определенных ранее видов субъектов управления и режимов управления (см. этап 3) следует рассмотреть и определить **средства управления лифтом** пассажиром (специалистом) в нормальных условиях и в аварийных ситуациях.

Рассмотрим средства управления лифтом пассажиром в нормальных условиях. Состав этих средств, а также их управляющие воздействия определим на основании сценариев взаимодействия пользователей с лифтом, сформированных на этапе 1.

**Кнопка** (с номером этажа в кабине, вызова лифта на площадке, «Закрыть двери», «Вызов диспетчера») может быть: в нерабочем состоянии (заблокирована); в рабочем состоянии (не заблокирована) и нажата; в рабочем состоянии и не нажата. Кнопка (из перечисленных ранее) может находиться в нерабочем состоянии: в результате внешнего воздействия, не зависящего от лифта; в опасной ситуации, когда лифтом пользоваться нельзя; постоянно (кнопки ВНИЗ на площадке нижнего этажа и ВВЕРХ на площадке верхнего этажа заблокированы).

Для регулирования работы дверей кабины и на лифтовой площадке используется **таймер**.

**Этап 5.** Состояния и процессы, связанные с объектом управления и его средой.

Лифт либо исправен, либо не исправен. Исправный (неисправный) лифт либо включен, либо выключен. В каждом из этих четырех случаев нарушения в состоянии среды либо наблюдаются, либо нет. Отсюда имеем восемь состояний лифта.

Для примера рассмотрим **состояния** кабины лифта и двери в нормальных условиях (лифт исправен, включен, среда в нормальном состоянии). Кабина лифта: движется вверх; движется вниз; находится в покое (в состоянии нормальной остановки). Дверь: закрыта; не закрыта, открывается; не закрыта, закрывается; открыта.

**Процессы**, в которых задействована кабина лифта: движение (ожидание) в нормальных условиях либо в аварийной ситуации.

Существуют два вида **ситуаций протекания процессов**: нормальные условия и аварийная ситуация.

Если кабина движется в нормальных условиях для обслуживания вызова на некоторый этаж (назовем его и направление движения к нему целевыми), то это движение либо не прерывается (кабина без остановок прибывает на этаж согласно вызову), либо прерывается, чтобы обслужить «попутный» вызов, поступивший в процессе движения кабины к ранее намеченному этажу.

Рассмотрим случай, когда во время движения кабины по вызову поступает новый вызов на этаж, находящийся на пути к целевому этажу. Если кабина не успевает затормозить на требуемом во вновь поступившем вызове этаже, то вызов не обслуживается. С учетом этого обстоятельства будем различать состояния движущейся между этажами кабины и **фазы** ее движения:

— кабина движется в зоне торможения перед очередным этажом (т.е. не успевает затормозить для остановки на ближайшем по ходу движения этаже);

— кабина движется не в зоне торможения перед очередным этажом.

Возникает задача: необходимы средства, которые дают возможность определить, в какой зоне движения между этажами находится кабина.

**Этап 6.** Структурная модель СУ.

**Подсистемы управления** определяются с учетом составляющих лифта, условий его функционирования, субъектов управления лифтом.

На данном уровне детализации проекта представим составляющую системы управления в виде упорядоченной тройки вида:

(субъект управления, режим управления, составляющая объекта управления).

Учитывая, что пользователь имеет средства управления двумя составляющими лифта, а режимов управления два, то имеем четыре тройки, первым компонентом в которых есть «пользователь». Для специалиста ограничения на воздействия на составляющие лифта не вводились. Так как составляющих шесть (поскольку для дверей кабины и на лифтовой площадке есть средство управления, к тому же они должны работать согласованно, выделяется такая составляющая, как двойные двери), имеем 12 троек с первым компонентом «специалист». Всего 16 подсистем СУ. В процессе разработки проекта СУ может оказаться, что не все эти подсистемы следует разрабатывать. Однако для полноты построений подсчитываем все.

**Этап 7.** Информационная модель СУ.

Предлагается такой порядок построения множества атрибутов. Для введения и описания атрибутов СУ рассматриваем:

— средства управления, вспомогательные и информационные средства для удобства пассажиров;

— связи средств управления с компонентами объекта управления;

— состояния средств управления, вспомогательных и информационных средств;

— состояния каждой составляющей объекта управления, связанной со средствами управления.

Введем атрибуты: средств управления; вспомогательных и информационных средств для удобства пассажиров; компонентов объекта управления, связанных со средствами управления.

Так, например, связаны кнопки вызова лифта и направления его на этажи и кабина, кабина и световые индикаторы.

Пусть действия пользователей (нажатия кнопок) моделируются с помощью сообщений вида:

‘нажата кнопка в кабине направления лифта на этаж <номер этажа>’;

‘нажата кнопка ВВЕРХ на этаже <номер этажа>’;

‘нажата кнопка ВНИЗ на этаже <номер этажа>’.

Здесь <номер этажа> — целое число в диапазоне MD:MU, где MD — номер самого нижнего этажа, MU — номер самого верхнего этажа, MD, MU — целые неотрицательные числа.

Для краткого обозначения фраз (клише) «нажата кнопка в кабине направления лифта на этаж», «нажата кнопка ВВЕРХ на этаже», «нажата кнопка ВНИЗ на этаже» введем последовательности символов bC, bUp, bDn соответственно.

Для сообщений введем глобальный атрибут перечислимого типа Mes со значениями:

(bC, <номер этажа>);

(bUp, <номер этажа>);

(bDn, <номер этажа>);

(E,0) (означает, что сообщений нет).

При включении лифта сообщений нет. После передачи сообщения поля атрибута Mes очищаются.

Каждой кнопке и световому индикатору на лифтовой площадке соответствует номер этажа  $i$ , на котором находится эта площадка;  $i$  — целое число, принимающее значения от MD до MU.

Для описания состояний кнопок для направления лифта на этажи введем одномерный массив Pan\_Cab со значениями bl, pr, nf:

Pan\_Cab [ $i$ ] = bl означает, что заблокирована кнопка направления лифта на этаж  $i$ ;

Pan\_Cab [ $i$ ] = pr означает, что не заблокирована и нажата кнопка направления лифта на этаж  $i$ ;

Pan\_Cab [ $i$ ] = nf означает, что не заблокирована и не нажата кнопка направления лифта на этаж  $i$ .

Для хранения информации о целевом направлении движения кабины лифта введем глобальный атрибут перечислимого типа DD со значениями:

Up означает, что целевым направлением движения кабины лифта является движение вверх;

Down означает, что целевым направлением движения кабины лифта является движение вниз;

Undef означает, что целевое направление движения кабины лифта не определено.

При включении лифта целевое направление движения лифта не определено.

Для хранения информации о номере целевого этажа введем глобальный атрибут DF перечислимого типа со значениями:

(<признак>, <номер этажа>), где <признак> принимает булевы значения;

(1, <номер этажа>) означает, что номер целевого этажа определен и равен <номер этажа>;

(0,0) означает, что номер целевого этажа не определен.

Кнопки вызова лифта и направления на этажи могут нажимать разные пользователи (а кабина одна). В результате нажатия кнопок вызова лифта и направления его на этажи независимо несколькими пользователями возникает совокупность вызовов, из которых образуется очередь. Следовательно, нужны средства представления и обслуживания этой очереди.

Для моделирования очереди вызовов введем атрибут QC типа «список». При включении лифта очередь QC пуста. Если очередь QC пуста, то целевое направление движения кабины лифта и целевой этаж не определены.

Кроме очереди вызовов, требуется блок ее обслуживания. Этот блок динамически упорядочивает вызовы, определяет целевое направление движения кабины и целевой этаж. Для моделирования такого блока введем функцию callQC с аргументами ⟨номер этажа⟩, ⟨вид кнопки⟩, DD, DF, QC для пополнения очереди вызовов QC вызовом кабины на этаж ⟨номер этажа⟩ вида ⟨вид кнопки⟩ и для переопределения целевого направления движения кабины лифта DD и/или целевого этажа DF, если это требуется. Здесь ⟨вид кнопки⟩ — атрибут перечислимого типа с символьными значениями bC, bUp, bDn (их смысл пояснен при описании атрибута Mes), которые дают возможность определить, какая кнопка управления движением кабины лифта нажата.

Для хранения информации о состоянии кабины лифта введем атрибут перечислимого типа LiftCab со значениями, которые соответствуют состояниям кабины и условиям, в которых она находится. Приведем некоторые его значения:

(CUp, ⟨номер этажа⟩) — нормальные условия, кабина лифта движется вверх от этажа ⟨номер этажа⟩ (миновала этаж ⟨номер этажа⟩, но не доехала до следующего) и не находится в зоне торможения перед очередным этажом;

(CUpS, ⟨номер этажа⟩) — нормальные условия, кабина лифта движется вверх в нормальном режиме от этажа ⟨номер этажа⟩ (миновала этаж ⟨номер этажа⟩, но не доехала до следующего) и находится в зоне торможения перед очередным этажом.

#### **Этап 8.** Модель поведения.

Модель поведения подсистемы СУ ⟨пользователь, нормальные условия, кабина⟩ описывает для кнопок переходы из состояния *нажата* в состояние *не нажата* и наоборот, а также **результаты управляющих воздействий** их нажатия.

Для построения модели поведения выделяются виды условий (ситуации) для кнопок, когда происходит переход кнопки в состояние *нажата*, а также рассматриваются условия, когда нажатую кнопку следует перевести в состояние *не нажата*.

**Ситуации**, в которых наблюдается текущее состояние кнопки каждого вида, когда поступает вызов на некоторый этаж  $j$ :

— кабина лифта находится в состоянии нормальной остановки на этаже  $i$  ( $i = j$ );

— кабина лифта находится в состоянии нормальной остановки на этаже  $i$  ( $i \neq j$ );

— кабина лифта движется в нормальных условиях.

Приведем фрагмент модели для указанной подсистемы СУ, записанный на ЕЯ с использованием фраз рабочей лексики.

Управляющее воздействие: нажатие кнопки с номером этажа в кабине лифта.

Ситуация (состояние кабины, фаза процесса, в котором задействована кабина): кабина лифта движется и не находится в зоне торможения перед следующим этажом.

Условия изменения состояния кнопки в заданной ситуации при данном воздействии:

— если кабина лифта движется вверх от этажа  $i$  и принято сообщение ‘нажата кнопка направления лифта на этаж  $j$ ,  $j > i + 1$ , то пополняется очередь вызовов QC, определяется целевое направление и целевой этаж, не заблокирована и нажата кнопка направления лифта на этаж  $j$ ;

— если кабина лифта движется вверх от этажа  $i$ , принято сообщение ‘нажата кнопка направления лифта на этаж  $j$ ’,  $j = i + 1$ , то пополняется очередь вызовов QC, целевым этажом становится этаж  $j$ , не заблокирована и нажата кнопка направления лифта на этаж  $j$ ;

— если кабина лифта движется вверх от этажа  $i$  и принято сообщение ‘нажата кнопка направления лифта на этаж  $j$ ’,  $j < i + 1$ , то не заблокирована и не нажата кнопка направления лифта на этаж  $j$ .

Эти три условия можно представить в виде

⟨условие⟩ ---- > ⟨постусловие⟩

согласно описанию атрибутов и указанной фазе процесса следующим образом:

LiftCab(CUp,  $i$ ) & Mes = (bC,  $j$ ) & ( $j > i + 1$ ) ---- > callQC( $j$ , bC, DD, DF, QC);  
Pan\_Cab[ $j$ ] := np; Mes := (E, 0).

LiftCab(CUp,  $i$ ) & Mes = (bC,  $j$ ) & ( $j = i + 1$ ) ---- > callQC( $j$ , bC, DD, DF, QC);  
DF :=  $j$ ; Pan\_Cab[ $j$ ] := np; Mes := (E, 0).

LiftCab(CUp,  $i$ ) & Mes = (bC,  $j$ ) & ( $j < i + 1$ ) ---- > Pan\_Cab[ $j$ ] := nf; Mes := (E, 0).

Этапы 9 и 10 здесь не рассматриваются, поскольку переходить к ним следует после выполнения всех заданий предыдущих этапов.

#### ЗАКЛЮЧЕНИЕ

В данной работе предложен общий подход к построению методики проектирования СУ, основанный на принципах, среди которых принцип лингвистически управляемого проектирования. В соответствии с этим принципом построение и детализация моделей, отражающих различные аспекты СУ, осуществляется на основе систематического использования ряда существенных (в области проектирования СУ) понятий. Приведен перечень этих понятий. Он построен на основе языковых смысловых связей сущности «система управления» с другими сущностями, и эти связи и сущности выявляются последовательно.

Этот перечень использован для построения методики проектирования СУ. Он является средством, которое «направляет» процесс построения ЕЯ-спецификации системы управления объектом на языке Разработчика, «подсказывает» Разработчику, какие этапы разработки следует пройти и в каком порядке, «помогает» Разработчику избежать упущений, «напоминает» о необходимости обосновать решения. В результате получилось средство разработки СУ, управляемой «такого рода перечнем». Если развить этот перечень, используя его в качестве задела для концептуализации, в онтологию (предметной области «Проектирование СУ»), предложенная методика проектирования будет частично автоматизирована.

Отметим, что описанная методика сочетается с agile-технологией разработки систем [14, 15], с техникой спецификацией примерами (SBE) [16], пользовательских историй (user stories) [17–19].

В ходе применения методики требуется вести учет уточнений, отложенных заданий. Это трудоемкий процесс, которому уделяют внимание и другие авторы [20]. Отметим также, что в случае составной СУ приходится распределять внимание между несколькими подсистемами проектируемой СУ. Принимая во внимание лишь перечисленные трудности, можно сделать вывод, что как для отдельного разработчика, так и для группы разработчиков желательно иметь инструментальные средства поддержки процесса проектирования, которые:

— направляют процесс построения ЕЯ-спецификации системы управления объектом на языке Разработчика, подсказывая Разработчику, какие этапы разработки следует пройти и в каком порядке;

- помогают Разработчику не упустить все, что существенно;
- напоминают о необходимости обосновать решения;
- ведут историю проектирования, что обеспечивает возможность вернуться и изменить решение;
- обновляют описания;
- поддерживают раздел описаний атрибутов, в котором для каждого атрибута есть пояснение его смысла;
- ведут рубрику «Требуемые уточнения и/или отложенные работы»;
- поддерживают процесс создания документации разработанной системы управления объектом.

Отметим, что принципы 1.1–1.4, 1.6 (см. разд. 1) имеют «организационно-регламентирующий» характер (предписывают рассматривать СУ и то, что с ней связано, с различных точек зрения, обосновывать принимаемые решения, заботиться о том, чтобы ничего не упустить, формировать рабочую лексику и пользоваться ею, чтобы облегчить переход от ЕЯ-спецификаций к формализованному представлению, обеспечить целостность построения модели СУ), а принцип 1.5 «порождающий» (именно на основе перечня сущностей и связанных с ними понятий, сконструированного по этому принципу, разрабатывается методика проектирования).

Применение принципов 1.1–1.4, 1.6, как нетрудно заметить, не ограничивается исключительно областью «Проектирование СУ». Принцип 1.5 использован в данной работе для построения перечня существенных понятий в области «Проектирование СУ», но его можно применить и для построения аналогичных перечней для других областей проектирования, а затем строить методики проектирования в этих областях подобно тому, как это сделано в данной работе.

#### СПИСОК ЛИТЕРАТУРЫ

1. Lee E.A., Seshia S.A. Introduction to embedded systems: A cyber-physical systems approach. Sec. Ed. Cambridge, Mass.: MIT Press, 2017. 568 p.
2. Летичевский А.А., Летичевский А.А. мл., Скобелев В.Г., Волков В.А. Кибер-физические системы. *Кибернетика и системный анализ*. 2017. Т. 53, № 6. С. 3–19.
3. Gause D.C., Weinberg G.M. Exploring requirements: Quality before design. Dorset House Pub., 1989. 299 p.
4. Howard M., LeBlanc D., Viega J. 24 deadly sins of software security. McGraw-Hill, 2009. 432 p.
5. Finkel S. How to write good requirements/user stories in Scrum. URL: <https://www.agileaces.net/write-good-requirements-user-stories-scrum>.
6. Alford M.W., Lawson J.T. Software requirements engineering methodology (Development). TRW Defense and Space Systems Group. 1979.
7. Software Requirements Engineering (2nd ed.). Thayer R.H., Dorfman M. (Eds.). US: IEEE Computer Society Press, 1997. 552 p.
8. Колчин А.В., Летичевский А.А., Потиеенко С.В., Песчаненко В.С. Обзор современных систем и методов верификации формальных моделей. *Проблемы програмування*. 2012, № 4. С. 75–88.
9. Kolchin A., Potiyenko S., Weigert T. Challenges for automated model-based test scenario generation. In: Damaševičius R., Vasiljeviene G. (Eds). Information and Software Technologies – 25th International Conference, ICIST 2019, Vilnius, Lithuania, October 10–12, 2019, Proceedings. Springer: *Communications in Computer and Information Science*. 2019. Vol. 1078. P. 182–194.
10. Gotel O., Finkelstein A. An analysis of the requirements traceability problem. In: ICRE IEEE Computer Society. 1994. P. 94–101.
11. Hallerstede S., Jasrtam M., Ladenberger L. A method and tool for tracing requirements into specifications. *Science of Computer Programming*. 2014. Vol. 82. P. 2–21.

12. Годлевский А.Б., Мищенко Н.М., Мороховец М.К., Фелижанко О.Д., Щёголева Н.Н. Методика разработки и средства формализации функциональных спецификаций систем и устройств. *Кибернетика и системный анализ*. 2016. Т. 52, № 5. С. 25-42.
13. Мищенко Н.М., Мороховец М.К., Фелижанко О.Д., Штелик Е.В., Щёголева Н.Н. Новые функциональные возможности системы обработки естественных языковых спецификаций и среда её функционирования. *Кибернетика и системный анализ*. 2018. Т. 54, № 6. С. 37-46.
14. Cockburn A. Agile software development. Boston: Addison-Wesley, 2001. 256 p.
15. Agile Alliance. URL: <http://www.agilemanifesto.org>.
16. Adzic G. Specification by example. How successful teams deliver the right software. New York: Manning Publications, 2011. 295 p.
17. Cockburn A. Writing effective use cases. Addison-Wesley, 2000. 304 p.
18. Jeffries R. Essential XP: Card, conversation, and confirmation. URL: <https://ronjeffries.com/xprog/articles/expcardconversationconfirmation/>.
19. Ralph P. The sensemaking-coevolution-implementation theory of software design. *Science of Computer Programming*. 2015. Vol. 101. P. 21-41.
20. The Scrum product backlog – International Scrum Institute. URL: [https://www.Scrum-institute.org/The\\_Scrum\\_Product\\_Backlog.php](https://www.Scrum-institute.org/The_Scrum_Product_Backlog.php).

Надійшла до редакції 04.06.2020

**О.Б. Годлевський, М.К. Мороховець**  
**КІБЕРФІЗИЧНІ СИСТЕМИ: ЛІНГВІСТИЧНИЙ АСПЕКТ**

**Анотація.** Запропоновано підхід до проектування систем керування технічними об'єктами, особливістю якого є лінгвістично кероване проектування. Подано розроблену на базі цього підходу методику проектування систем керування, призначену для застосування на стадії проектування, що передують побудові формальної моделі.

**Ключові слова:** система керування, лінгвістично кероване проектування, методика проектування систем керування, кіберфізична система.

**A.B. Godlevsky, M.K. Morokhovets**  
**CYBER-PHYSICAL SYSTEMS: LINGUISTIC ASPECT**

**Abstract.** An approach to the design of control systems for engineering objects is proposed. Its peculiarity is a linguistically driven design. The methodology for control systems design developed on the base of this approach is given. It is intended for use at the design stage preceding the construction of a formal model.

**Keywords:** control system, linguistically-driven design, methodology for control systems design, cyber-physical system.

**Годлевский Александр Богуславович,**  
 кандидат физ.-мат. наук, старший научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: [a.godl49@gmail.com](mailto:a.godl49@gmail.com).

**Мороховец Марина Константиновна,**  
 кандидат физ.-мат. наук, старший научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: [marina.morokhovets@gmail.com](mailto:marina.morokhovets@gmail.com).