

ЯЗЫК UML В ЗАДАЧАХ ЦИФРОВОЙ ОБРАБОТКИ СИГНАЛОВ ПЕРСПЕКТИВНЫХ РАДИОНАВИГАЦИОННЫХ СИСТЕМ

Сенченко Я. И.

Научный руководитель – канд. техн. наук, ст. науч. сотр. Кузьмин Е. В.

Сибирский федеральный университет, г. Красноярск

В настоящее время унифицированный язык моделирования – UML, является одной из современных технологий в области программной инженерии. UML позволяет системным архитекторам представлять свое видение системы в виде набора стандартных диаграмм. Сложную систему на языке UML можно представить в виде набора небольших моделей-диаграмм, каждая из которых фокусируется на каком-то определенном аспекте функционирования системы и выражает разный уровень абстракции. Одна диаграмма может описывать взаимодействие пользователя с системой, другая – изменение состояний системы в процессе ее работы, третья – взаимодействие между собой элементов системы и т. д. [1].

Цель: применение унифицированного языка моделирования UML для решения задачи реализации цифровой системы фазовой синхронизации приемоиндикатора перспективной радионавигационной системы.

Диаграммы UML помогают разработчикам определить границы контекста моделируемой предметной области на ранних этапах проектирования, сформировать общие требования к поведению проектируемой системы, разработать концептуальную модель системы для ее последующей детализации, подготовить документацию для взаимодействия с заказчиками и пользователями системы. При построении диаграмм UML используются следующие элементы: фигуры, линии, значки, надписи. Фигуры используются "плоские" – прямоугольники, эллипсы, ромбы и т. д. Линии своими концами должны соединяться с фигурами. Применяется два типа линий – сплошная и пунктирная. Любые системы проектируются с учетом того, что в процессе своей работы они будут использоваться людьми и/или взаимодействовать с другими системами. Диаграммы прецедентов представляют динамические или поведенческие аспекты системы [1].

На рисунке 1 показана диаграмма прецедентов при разработке блока цифровой обработки сигналов (ЦОС) приемоиндикатора перспективной наземной радионавигационной системы [2]. На ней представлены пять действующих лиц. Научный консультант предоставляет техническое задание (ТЗ) на разработку блока ЦОС, составляет сроки разработки, корректирует их при необходимости и консультирует инженеров на каждом этапе разработки. Каждый из трех инженеров занимается разработкой определенного модуля. Далее все разработанные модули объединяются в один блок ЦОС. На диаграмме прецедентов это показано пунктирными линиями с помощью отношения включения («include»). Отношение «include», устанавливаемое между прецедентами, позволяет включить поведение одного прецедента в поток другого прецедента, а отношение «extend» предоставляет возможность ввести новое поведение в существующий прецедент [3]. Затем разработчики вместе проверяют результаты их общей разработки и докладывают о них научному консультанту, который, в свою очередь, представляет опытный образец заказчику для внедрения разработки в производство.

Центральное место в объектно-ориентированном подходе занимает диаграмма классов, концентрируя в себе большой диапазон понятий моделирования. Диаграмма классов описывает типы объектов системы и различного рода статические отношения, которые существуют между ними [4]. Диаграммы классов могут применяться и при

прямом проектировании, то есть в процессе разработки новой системы, и при обратном проектировании – описании существующих и используемых систем. Таким образом, диаграмма классов – конечный результат проектирования и отправная точка процесса разработки. Они представляют собой описание совокупности объектов с общими атрибутами, операциями, отношениями и семантикой. Класс на диаграмме изображается в виде прямоугольника, разделенного горизонтальной линией на две части [1].



Рисунок 1 – Диаграмма прецедентов при разработке блока ЦОС

На рисунке 2 представлена диаграмма классов. Прочитав диаграмму, не трудно понять, что есть большой класс «Человек», который включает в себя трех инженеров. Они, используя «Инструмент» (компьютер) с программным обеспечением (ПО) Xilinx System Generator for DSP, разрабатывают каждый свой модуль. Еще один большой класс «Приемоиндикатор» включает в себя аналоговую и цифровую части. Модуль поиска сигнала, система кодовой синхронизации (СКС) и система фазовой синхронизации (СФС) относятся к классу «Цифровая часть». Такое отношение называют композитным агрегированием.

Диаграмма кооперации акцентирует внимание на организации объектов, принимающие участие во взаимодействии и предназначена для моделирования структурной организации потоков управления. Основное внимание при этом уделяется моделированию структурных отношений между взаимодействующими экземплярами, вдоль которых передаются сообщения. Диаграммы кооперации подходят для визуализации сложных итераций, ветвлений и параллельных потоков управления [5]. На диаграмме кооперации объекты обозначаются прямоугольниками с подчеркнутыми именами (чтобы отличить их от классов), ассоциации между объектами указываются в виде соединяющих их линий, над ними может быть изображена стрелка с указанием названия сообщения и его порядкового номера.

В UML диаграмма состояний используется для описания динамических свойств системы и показывает, в каких состояниях может находиться система, при каких усло-

виях эти состояния сменяют друг друга и по каким законам меняются переменные состояния в конкретных состояниях. Диаграммы состояний также показывают, как протекают параллельные процессы в моделируемом объекте, если они существуют [6]. Диаграммы состояний применяются для того, чтобы объяснить, каким образом работают сложные объекты. Диаграмма состояний показывает, как объект переходит из одного состояния в другое.

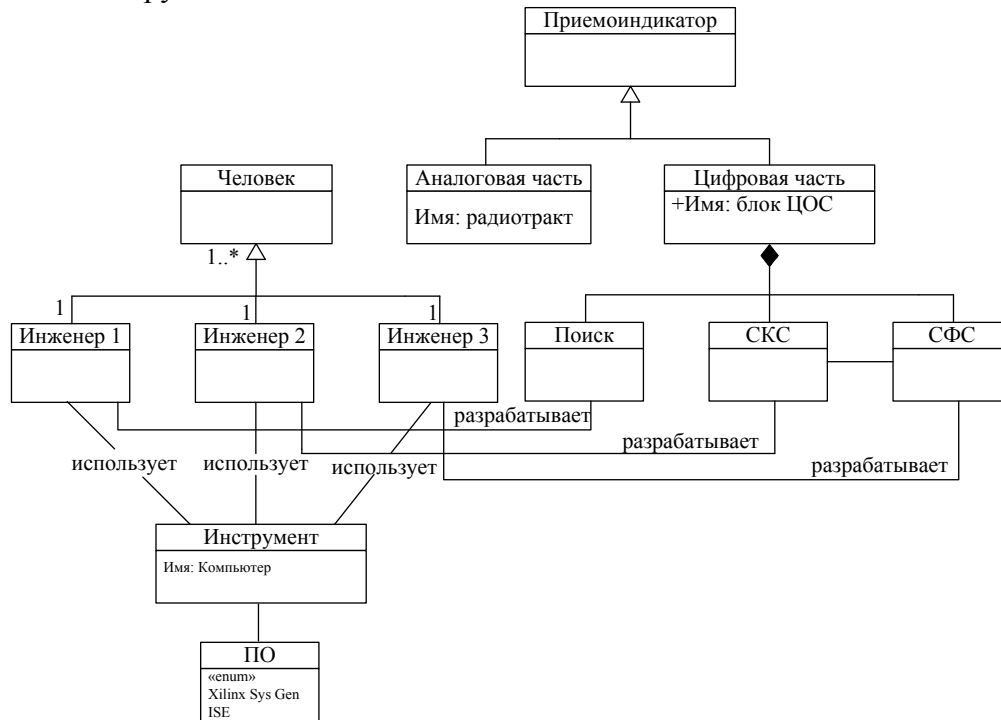


Рисунок 2 – Диаграмма классов при разработке блока ЦОС

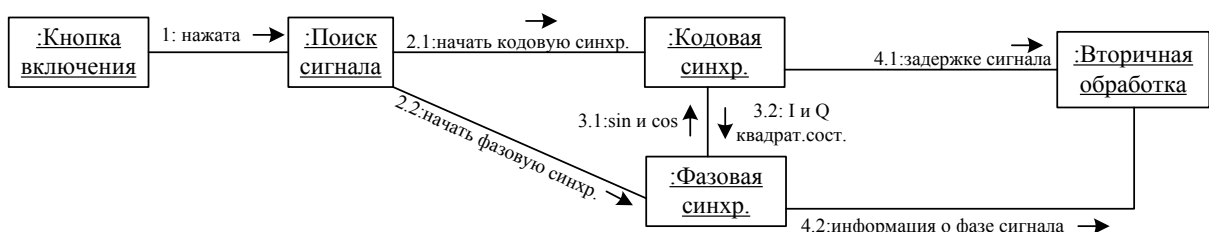


Рисунок 3 – Диаграмма кооперации при разработке блока ЦОС

Очевидно, что диаграммы состояний служат для моделирования динамических аспектов системы. Диаграмма состояний полезна при моделировании жизненного цикла объекта. Отличительной особенностью диаграммы состояний является то, что она описывает процесс изменения состояний только одного экземпляра определенного класса - одного объекта, причем объекта реактивного, то есть, поведение которого характеризуется его реакцией на внешние события. Понятие жизненного цикла применимо как раз к реактивным объектам, настоящее состояние (и поведение) которых обусловлено их прошлым состоянием. Но диаграммы состояний важны не только для описания динамики отдельного объекта. Они могут использоваться для конструирования исполняемых систем путем прямого и обратного проектирования.

Стрелками на диаграмме показываются переходы между состояниями, которые вызваны выполнением методов описываемого диаграммой объекта. Существует также два вида псевдосостояний: начальное, в котором находится объект сразу после его соз-

дания (обозначается сплошным кружком), и конечное, которое объект не может покинуть, если перешел в него (обозначается кружком, обведенным окружностью) [1].

На рисунке 4 показана диаграмма состояний разрабатываемого блока ЦОС.

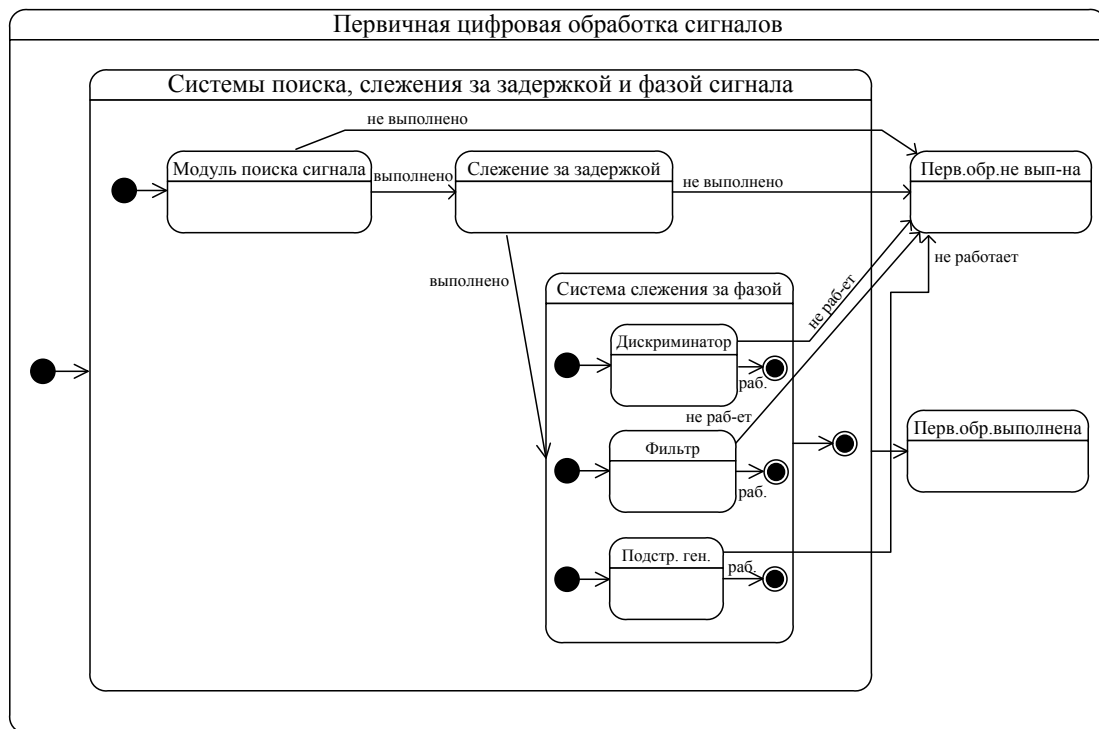


Рисунок 4 – Диаграмма состояний при разработке блока ЦОС

Вывод: применение унифицированного языка моделирования UML для решения задачи реализации цифровой системы фазовой синхронизации приемоиндикатора перспективной радионавигационной системы позволяет определить границы контекста моделируемой предметной области на ранних этапах проектирования, сформировать общие требования к поведению проектируемой системы, разработать концептуальную модель системы для ее последующей детализации.

Работа выполнена при поддержке гранта Российского фонда фундаментальных исследований (РФФИ) № 08-08-00849-а.

Список используемой литературы:

1. [А.В. Бабич](#). UML. Первое знакомство. – Бинوم. Лаборатория знаний, 2008. – 167 с., ил.
2. Реализация и исследование цифровой системы фазовой синхронизации приемоиндикатора широкополосной радионавигационной системы / Е.В. Кузьмин, Я.И. Сенченко // Современные проблемы радиоэлектроники: сб. науч. тр. – Красноярск: Сибирский федеральный ун-т; 2010. – с. 188-192.
3. Арлоу Д., Нейштадт И. UML 2 и Унифицированный процесс. Практический объектно-ориентированный анализ и проектирование, 2-е издание. – Пер. с англ. – СПб: Символ-Плюс, 2007. – 624 с., ил.
4. Фаулер М., Скотт К. UML. Основы. – пер. с англ. – СПб: Символ-Плюс, 2002. – 192 с., ил.
5. Ларман К. Применение UML и шаблонов проектирования. 2-е издание.: пер. с англ. – М.: Издательский дом «Вильямс», 2004. – 624 с.: ил.