

## РАЗРАБОТКА КОМПЬЮТЕРНЫХ ТРЕНАЖЕРОВ-СИМУЛЯТОРОВ

Вагин Р.Н.,

научный руководитель к. т. н., доцент Осипов А.С.

*Сибирский федеральный университет*

### 1. Введение

Р-330 «Мандат» — автоматизированный комплекс радиоэлектронного подавления (РЭП). Предназначен для радиоразведки и радиоподавления линий радиосвязи противника в тактическом и оперативно-тактическом звеньях управления в диапазоне от 1,5 до 100 МГц. Состоит на вооружении подразделений, частей и соединений радиоэлектронной борьбы (РЭБ) Сухопутных войск России, Украины и Белоруссии.

В зависимости от организационно-штатной структуры, количества и типов применяемых средств, решаемых задач, «Мандат» при боевом применении может иметь трех или двухуровневое построение. Трехуровневое построение включает: пункт управления батальона (ПУБ), ротные пункты управления (РПУ), автоматизированные станции помех (АСП). Двухуровневое построение включает: пункт управления батальона (роты) и АСП.

При достаточном количестве пунктов управления (ПУ) и станций радиопомех, а также в зависимости от поставленной задачи, часть средств (один ПУ и несколько АСП) могут выводиться из состава комплекса для выполнения самостоятельной задачи на отдельном направлении, образуя так называемую выделенную группу двухуровневого построения.

Управление комплексом осуществляется посредством автоматизированных пунктов управления (АПУ) Р-330К. АПУ рассчитан для работы в трех режимах:

- средствами комплекса;
- тактический;
- выделенная группа.

Все три режима реализованы программным способом и устанавливаются при вводе исходных данных. Каких-либо переключений, изменения состава аппаратуры не требуется.

В автоматизированных системах управления (АСУ) обмен информацией между человеком-оператором и вычислительными средствами осуществляется с помощью устройств ввода и отображения информации. Роль такого устройства в АПУ Р-330К выполняет устройство диалогового отображения данных (УДОД). С помощью УДОД осуществляется информационное обеспечение лиц боевого расчета ПУ — отображение хода и результатов боевой работы.

### 2. Разработка компьютерного тренажера-симулятора УДОД

Компьютерный процедурный тренажер-симулятор УДОД предназначен для отработки обучающимися процедур подготовки АПУ Р-330К к боевой работе и ее ведения.

В соответствии ГОСТ 21036-75 тренажер — это техническое средство «профессиональной подготовки человека-оператора, предназначенное для формирования и совершенствования у обучаемых профессиональных навыков и умений, необходимых им для управления материальным объектом, путем

многократного повторения обучаемыми действий, свойственных управлению реальным объектом».

Тренажеры применяются везде, где использование реальной системы для целей обучения оказывается неэффективным, дорогостоящим, опасным, неудобным или невозможным делом. Достоинствами компьютерных тренажеров (КТ) также являются: возможность контроля знаний обучаемых, компактность, устойчивость к ошибкам, возможность одновременного охвата большого количества обучаемых, возможность обучения на дому.

По принципу внутреннего устройства и функционирования современные КТ можно условно разделить на несколько разновидностей: электронные экзаменаторы, статические и динамические тренажеры.

Наиболее интересны динамические КТ. В основе они имеют математические модели физических процессов. Они учат понимать влияние различных управляющих воздействий на технологические процессы и потому позволяют обучать более качественно. Разумеется создание такого тренажера процесс трудоемкий и нетривиальный.

Компьютерный симулятор — это программное средство имитирующее (воспроизводящее) реальный объект, отображая часть реальных явлений и свойств в виртуальной среде.

В разработке тренажеров присутствуют три субъекта: производитель, преподаватель и обучаемый. Все они имеют свои цели, и каждый источник целей создает предпосылки для реализации в тренажере тех или иных качеств.

Цели преподавателя:

- высокое качество результата;
- эффективность учебного процесса;
- удобство организации учебного процесса;
- большое количество учебно-тренировочных задач;
- индивидуальное и групповое обучение;
- удаленное обучение;
- контроль знаний и навыков;
- минимум требований к используемой вычислительной технике (ВТ).

Цели обучаемого:

- высокое качество результата;
- максимальная правдоподобность и реалистичность;
- понятность процесса обучения;
- интерес к процессу обучения.

Цели разработчика:

- простота разработки;
- удобство сопровождения;
- масштабируемость;
- привлекательность для рынка.

Первым и самым важным этапом создания тренажера является выбор средств разработки. Средства разработки программного обеспечения (ПО) — совокупность приемов, методов, методик, а также набор инструментальных программ, используемых разработчиком для создания программы, отвечающей заданным требованиям.

В последнее время для разработки простых и средней сложности КТ очень часто используется Flash. Adobe Flash, или просто Flash — мультимедийная платформа компании Adobe для создания веб-приложений или мультимедиа презентаций. Flash позволяет дизайнерам помещать объекты там, где они хотят, не беспокоясь об абсолютном и относительном размещении, проблемами с шаблонами, z-индексами,

разрешениях экрана и так далее. Это важное свойство значительно упрощает создание интерфейса тренажера.

Интерактивность, обработку данных и многое другое во flash-приложениях обеспечивает ActionScript — объектно-ориентированный язык программирования. ActionScript исполняется виртуальной машиной, которая является составной частью Flash Player. ActionScript компилируется в байт-код, который включается в SWF-файл.

Adobe Flash Player — это бесплатная межплатформенная среда выполнения на основе браузера для доставки динамических приложений, контента и видеоматериалов на различные экраны и браузеры.

Рассмотрим две основные проблемы, возникающие при разработке такого КТ как тренажер-симулятор УДОД, и методы их решения.

Проблема 1 — как реализовать физический объект в виде программного кода?

Первый важный шаг на пути решения проблемы — использование схемы проектирования Model-view-controller (MVC). Модель данных приложения, пользовательский интерфейс и взаимодействие с пользователем разделяются на три отдельных компонента так, что модификация одного из компонентов оказывает минимальное воздействие на остальные.

Следующий шаг — абстракция. В процессе моделирования необходимо отвлечься от несущественных сторон, свойств и связей объекта. Вовсе не важно, как в реальном устройстве соединены провода, какие там токи, как устроены блоки памяти и т. п. Но нам очень важно знать, какие действия произойдут, если нажать тот или иной тумблер, клавишу или кнопку. Необходимо выделить логические единицы (память, клавиатура, устройство отображения и т. д.) и создать их абстрактные модели.

Применение модульности станет третьим важным шагом. Модульность в языках программирования — принцип, согласно которому программное средство (ПС) разделяется на отдельные именованные сущности, называемые модулями. То есть приложение должно состоять из нескольких независимых модулей, каждый из которых возможно модифицировать отдельно.

Простота разработки, достигается большим объемом повторно используемого кода. Эта цель осуществима путем декомпозиции функций (разделения их на составные части) и применения объектно-ориентированного программирования (ООП). ООП — парадигма программирования, в которой основными концепциями являются понятия объектов и классов. Класс — структура данных, представляющая модель объекта.

Проблема 2 — как реализовать выполнение процедур?

Управление вычислительными средствами ПУ Р-330К осуществляется путем ввода текстовых команд с клавиатуры УДОД. При этом команды выполняются последовательно, и по завершении работы процедуры на экране снова появляется приглашение к вводу новой команды. Это называется процедурным подходом.

В реальном объекте человек-оператор не сможет ввести новую команду, пока не завершится предыдущая. Устройство в любой момент времени находится в том или ином состоянии, которое определяет ее поведение.

Реализация ввода команд в КТ не тривиальная задача. Непонятно, как и когда нужно выполнять те или иные процедуры. Один из способов решения данной задачи — создать переменные состояния и в зависимости от их значений принимать решения. Но это значит, что необходимо постоянно проверять их значения. Такой подход далеко не идеальный и довольно сложно реализуемый, если система имеет множество параметров.

Другой способ основан на событийно-управляемом программировании. Всякий раз, когда мы интерактивно взаимодействуем с приложением, происходит некоторое событие. Например, если мы щелкаем мышью по командной кнопке, имеющейся в

нашем приложении, то она выступает источником события и подает сигнал о том, что оно произошло. Широковещательное распространение этого сигнала называется генерацией события. Существует также возможность программной генерации событий.

Любой объект, способный генерировать события, является отправителем событий, также называемым источником событий.

Обработчик событий — это процедура, вызываемая при возникновении соответствующего события. Функции не могут быть использованы в качестве обработчиков событий, так как возвращение значения в источник события с ее помощью невозможно.

Слушатель событий — это объект, который слушает или ожидает когда произойдет определенное событие. Если событие было сгенерировано, то слушатель передаёт параметры события обработчику.

Исходя из вышесказанного, решение проблемы кроется в динамическом связывании событий и их обработчиков. Т. е. необходимо создать точку входа в процесс (первый слушатель событий) и когда будет сгенерировано событие, которое этот он ожидает, управление будет передано обработчику. Первое действие, которое должен выполнить обработчик — удалить слушатель событий, который его вызвал. По окончании работы процедуры необходимо создавать новый слушатель событий. Таким образом, не требуется постоянно проверять состояние системы, работа программы ускоряется, и реализация сложной системы упрощается.

### **3. Выводы**

Для разработки простых и средней сложности КТ идеально подходит технология Flash. Она позволяет использовать принципы модульности, ООП, событийно-управляемое программирование, быстро и легко создавать интерфейсы любой сложности.

Использование схемы проектирования MVC может значительно упростить разработку КТ, позволив логически и физически разделить данные, управление и интерфейс.

Лучшим методом реализации выполнения процедур является применение динамического связывания событий и их обработчиков.