

# Система удалённого управления учебным робототехническим комплексом

Автор: студ. гр. У-104 Евсяков Артём Сергеевич  
Научный руководитель: доцент, к.т.н. Багаев Дмитрий Викторович

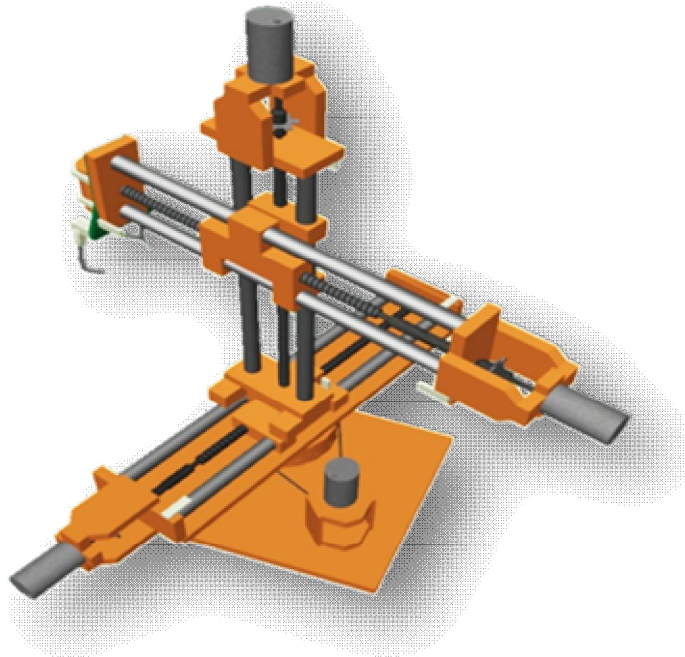
Необходимость удаленного управления обусловлена следующими соображениями. В процессе работы оператор системы должен следить за информацией о ходе технологического процесса, текущими координатами приводов, сообщениями об ошибках в системе управления и т.д. Между тем современные технологические линии нередко имеют протяженность, превышающую сотню метров. Традиционное решение состоит в том, что оператор непрерывно перемещается в зоне оборудования, обращаясь к специальным пультам, распределенным по длине рабочего участка. Подобный подход требует значительных затрат и увеличивает площадь рабочего пространства оператора. Альтернативой служит применение в качестве удаленных терминалов портативных компьютеров notebook или карманных компьютеров типа Palm или современных сотовых телефонов, которые могут быть переносными. При этом оператор может выбрать для себя наиболее комфортную позицию.

Существуют ситуации, когда построение распределенных систем создает дополнительные удобства, а также ситуации, когда без этого обойтись невозможно. В обоих случаях применяют удаленный терминал, который дублирует основной терминал системы или заменяет его. Особенность удаленного терминала состоит в том, что он может использовать иную платформу, его средства визуализации и управления более лаконичны. При этом должен осуществляться доступ к основным функциям ядра через локальную или корпоративную сети, а быть может, через Интернет.

Целью данной работы была разработка системы удалённого управления учебным робототехническим комплексом (УРТК).

Учебный робототехнический комплекс является макетом системы управления реального робототехнического комплекса. Он состоит из 3 мехатронных моделей линейного перемещения - степени свободы X, Y и Z и поворотного модуля – степень W. Смена инструменты с захвата на сверло и обратно осуществляется поворотом в необходимую сторону по степени F. Определение текущего положения производится семью концевыми датчиками, величина линейного перемещения по X, Y и Z определяется с помощью импульсных датчиков положения. Общий вид УРТК представлен на рисунке 1.

Робот подключен к блоку управления, который в свою очередь подключается к компьютеру через порт LPT. Существуют также и другие способы подключения УРТК к компьютеру, например блок управления с USB интерфейсом, что было учтено при разработке.



**Рис. 1. Общий вид УРТК**

Система удалённого управления представляет собой программный комплекс, реализующий следующие функции:

- возможность ручного и программного управления по координатам X, Y, Z, W и F с компьютера, непосредственно подключённого к УРТК (с сервера);
- возможность ручного и программного управления при подключении к серверу через компьютерную сеть;
- отображение на экране удалённого терминала и сервера текущих параметров УРТК: состояния концевых датчиков, значения координат и направлений движений;
- возможность создания управляющей программы методом ПОКАЗ-И-ОБУЧЕНИЕ.

При разработке программного комплекса было решено использовать язык программирования Java [5]. Что нам это дало:

- мощный объектно-ориентированный язык с поддержкой многопоточности и удобной реализации сетевых взаимодействий;
- возможность использовать в качестве клиентской части мобильные устройства (мобильные телефоны, смартфоны или КПК с поддержкой Java 2 Micro Edition [3]);
- кроссплатформенность (применение на компьютерах с различными операционными системами – в основном семейства Windows и Linux);
- наличие множества бесплатных средств разработки и готовых библиотек, что значительно удешевляет и упрощает разработку.

Так же есть и минусы при использовании платформы Java . Она не позволяет производить прямой доступ к аппаратным средствам (портам компьютера, ускорению видео), но это легко решается с помощью JNI (Java Native Interface) позволяющему подключать к программам, написанным на Java библиотеки на C/C++, что решает проблему доступа к аппаратному обеспечению, однако ухудшает переносимость – библиотека должна быть написана и скомпилирована под конкретную конфигурацию системы.

Для решения задачи удалённого управления была выбрана клиент-серверная архитектура. Сервер и удалённый терминал соединяются между собой с помощью компьютерной сети или сети Интернет. Сервером в нашем случае является компьютер, который непосредственно подключён к УРТК. Удалённым терминалом может являться персональный компьютер или мобильное устройство поддерживающее платформу Java 2 Micro Edition.

Система удалённого управления состоит из двух частей – серверное программное обеспечение и клиентское программное обеспечение, устанавливаемое на удалённом терминале.

На рисунке 2 показан интерфейс пользователя серверной программы.

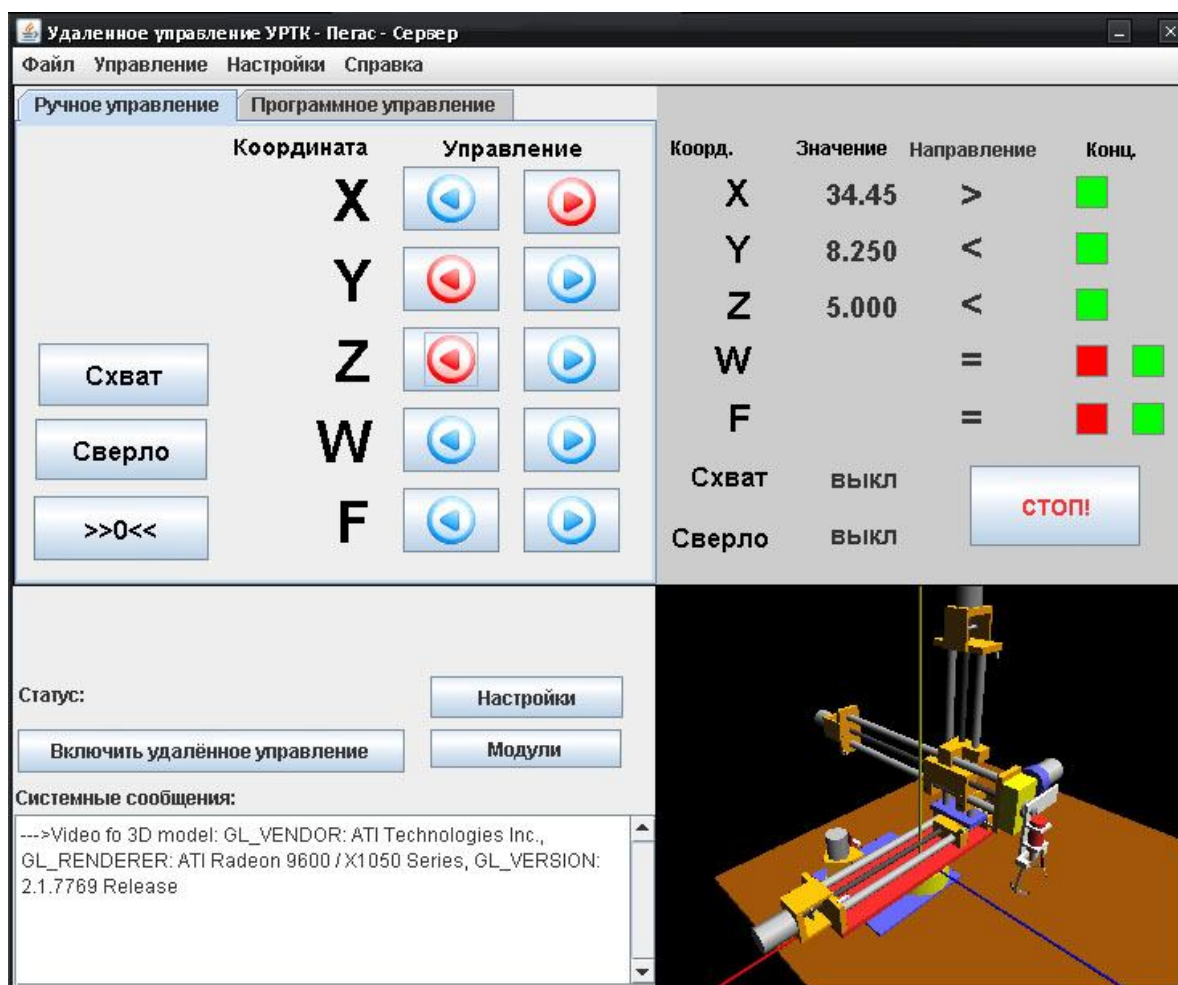


Рис. 2. Интерфейс пользователя серверной программы в режиме ручного управления

В интерфейсе пользователя можно выделить 4 части.

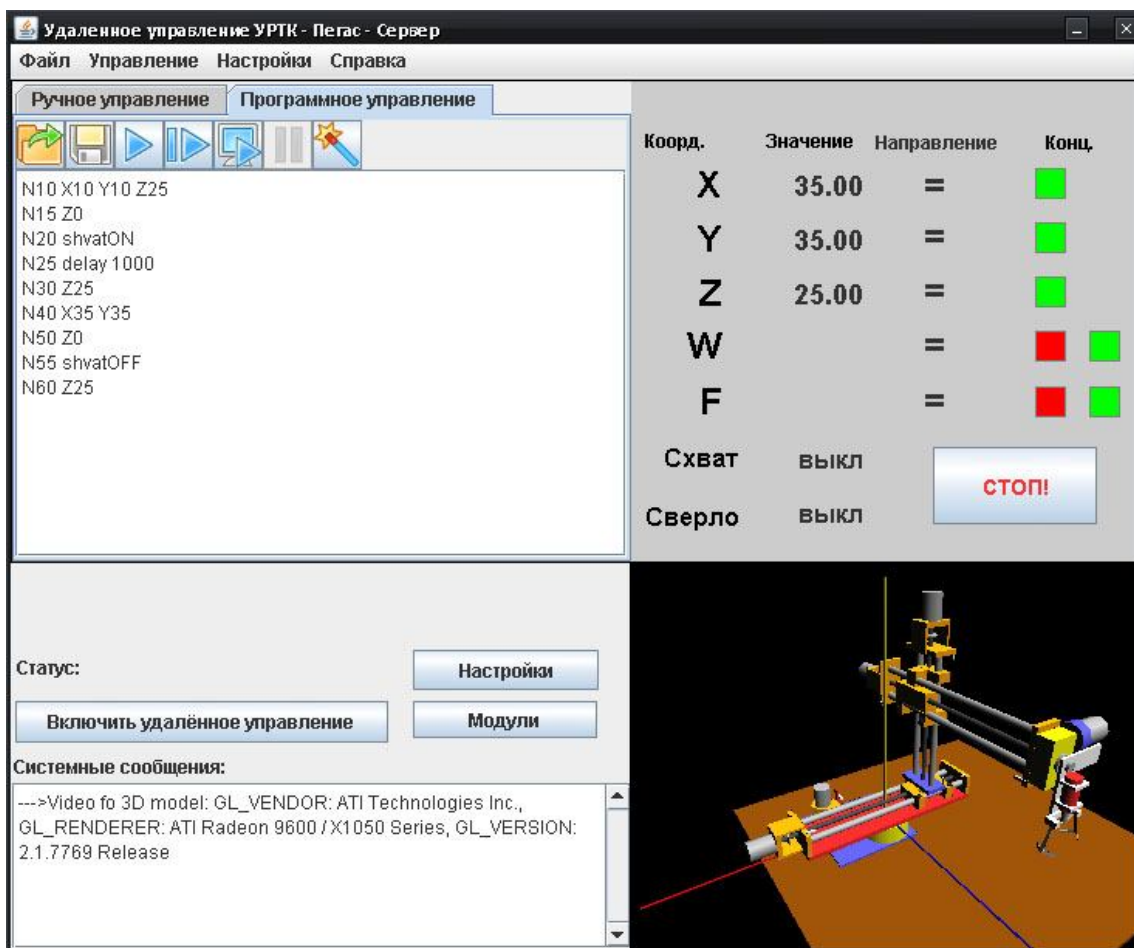
В верхнем левом углу расположены элементы управления. Управление перемещением осуществляется в двух режимах – ручном и программном. Для переключения между режимами управления необходимо переключиться между вкладками. Ручное управление производится с помощью экранных кнопок или нажатием соответствующих клавиш клавиатуры. При этом пользователь может запускать и останавливать движение всем координатам (X, Y, Z, W, и F) и включать и выключать хват и сверло робота. Экранная кнопка с надписью «>>0<<» переводит все координаты в ноль.

В верхнем правом углу программы находится панель, на которой отображаются все текущие параметры робота, а также кнопка аварийной остановки, при нажатии на которую останавливаются все движения. К текущим параметрам относятся значения линейных перемещений по координатам X, Y, Z, состояния концевых датчиков, текущие направления движения и состояния хвата и сверла.

В правом нижнем углу отображается 3D модель УРТК построенная исходя из текущих параметров. Она позволяет легче анализировать показания датчиков и текущее состояние УРТК.

В правом нижнем углу отображается окно с системными сообщениями, сообщения об ошибках, кнопка включения удалённого управления и прочие дополнительные функции.

На рисунке 3 показан интерфейс пользователя серверной программы в режиме программного управления.



**Рис. 3. Интерфейс пользователя серверной программы в режиме программного управления**

Панель инструментов в режиме программного управления позволяет открывать ранее сохранённые программы, сохранять программы, запускать программы на выполнение, пошагово выполнять программы, запускать программы в эмуляторе, останавливать выполнение программы и вызывать окно записи программы ПОКАЗ-И-ОБУЧЕНИЕ.

Управляющая программа представляет собой обычный текст и сохраняется в простом текстовом файле. При запуске программы текущее место выполнения программы подсвечивается красным цветом.

Язык программирования, на котором пишется управляющая программа, был специально разработан.

Список команд языка используемых при составлении управляющей программы с их описанием приведен в таблице 1.

Таблица 1 Список команд программного управления

Команда	Описание
;	начало комментария, текст начиная с символа ; до конца строки будет игнорироваться
[координата][значение]	запускается движение по координате до достижения необходимого значения, координата обозначается символом X , Y , Z , W или F

<b>N[текст]</b>	метка кадра, используется оператором goto и для отделения кадров программы друг от друга
<b>delay [значение]</b>	пауза при выполнении программы, значение указывается в миллисекундах
<b>if ([значение] ==get_signal([address])) [действие]</b>	условный оператор, если значение (не)равно значению, возвращаемому функций get_signal(address) то выполняется действие идущее после оператора, чаще всего этим действием является переход на метку кадра goto. Функция get_signal(address) – функция получения данных с внешних источников, она реализуется в классе интерфейса подключения робота
<b>goto [метка кадра]</b>	переход выполнения программы на метку кадр с такой меткой
<b>shvatON, shvatOFF, sverloON, sverloOFF</b>	включение и выключение схвата и сверла

Управляющая программа состоит из кадров, каждый из которых имеет свою метку. Каждый кадр пишется с новой строки и содержит информацию о выполнении каких либо действий.

Пример программы:

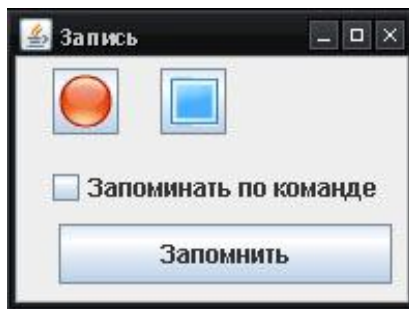
```

N10 X10 Y10 Z25
N15 Z0
N20 shvatON
N25 delay 1000
N30 Z25
N40 X35 Y35
N50 Z0
N55 shvatOFF
N60 Z25

```

УРТК при выполнении данной программы захватывает предмет находящийся в координатах X10 Y10 , поднимает его и перемещает в точку с координатами X35 Y35.

Другой немаловажной возможностью разработанной программы является создание программы в режиме ПОКАЗ-И-ОБУЧЕНИЕ. При этом программа сама или по команде пользователя запоминает все действия, а потом создаёт управляющую программу, которая полностью повторяет действия пользователя. Для запуска этого режима необходимо нажать кнопку с изображением волшебной палочки на панели программного управления. При этом на экране откроется окно записи, которое показано на рисунке 4.



**Рис.4. Окно записи в режиме ПОКАЗ-И-ОБУЧЕНИЕ**

При нажатии на кнопку с изображением красного шара – начинается запись. При нажатии на кнопку синем квадратом запись останавливается, создаётся управляющая программа и открывается окно программного управления с полученной программой. Теперь программу можно отредактировать, сохранить или запустить на выполнение.

Помимо режима работы когда программа сервера подключена непосредственно к УРТК, есть режим эмуляции УРТК. В этом режиме работа производится с программной моделью робота. Этот режим очень полезен при обучении, отладке самой программы сервера и отладке управляющих программ.

У программы имеется возможность подключения дополнительных модулей. Отображение списка доступных модулей происходит по нажатию на кнопке «Модули». Дополнительные модули служат для расширения функциональности программы без её перекомпиляции, что позволяет создавать модули сторонним разработчикам.

Через сеть интернет или компьютерную сеть к серверу могут подключаться пользователи удалённых терминалов. Подключение осуществляется по протоколу TCP/IP. Сервер и удалённые терминалы обмениваются сообщениями, синтаксис и семантика которых определяет протокол прикладного уровня [2], который также был разработан в ходе создания комплекса. Программы, установленные на удалённых терминалах, при помощи которых происходит подключение к серверу называются клиентскими программами. Возможно подключение к серверу с нескольких удалённых терминалов, но одновременно только с одного возможно управлять движениями робота. Клиент, который в данный момент управляет движением, называется активным. Он имеет возможность ручного и программного управления. Остальные удалённые терминалы находятся в пассивном режиме, им доступны только команды на получение параметров УРТК. Пассивный режим используется при обучении, для мониторинга параметров и при сопряжении работы нескольких роботов, когда одному роботу необходима информация о состоянии другого.

Интерфейс пользователя клиентской программы удалённого терминала на базе компьютера практически полно повторяет интерфейс пользователя серверной программы. Вместо кнопки «Включить удалённое управление» у

программы удалённого терминала есть кнопка «Подключиться к серверу». При нажатии на неё происходит подключение к серверу IP- адрес которого записан в настройках.

Интерфейс пользователя программы удалённого терминала на базе мобильного устройства (мобильный телефон, смартфон, КПК с поддержкой Java 2 Micro Edition) изображён на рисунке 5.



**Рис. 5. Интерфейс пользователя клиентской программы для мобильного устройства**

Разработанная система удалённого управления учебным робототехническим комплексом является полноценной системой управления с возможностью управления с удалённых терминалов. Такие системы в современном мире имеют широкий спектр применения – от систем охраны и видеонаблюдения до систем «умный дом», проведения удалённых хирургических операций и мониторинга сложных технологических объектов и комплексов.

Применение удалённого управления создаёт дополнительные удобства эксплуатации системы и позволяет вынести органы управления системой из производственной зоны и расположить терминал в удобном для работы месте, даже на очень большом расстоянии от объекта. Это позволяет оградить человека от неблагоприятных производственных факторов, появляется возможность централизованного управления – управления множеством устройств из одного места.

Использование в данной разработке платформы Java позволило снизить стоимость разработки и использовать систему как на компьютерах с установленной операционной системой на базе ядра Linux так и на компьютерах с операционной системой семейства Windows.

## Список использованной литературы

1. Ан. П. Сопряжение ПК с внешними устройствами: Пер. с англ. – М.: ДМК Пресс, 2001
2. Джеймс Ф.Куроуз Компьютерные сети. 2-е изд. – СПб.:Питер, 2004
3. Лекции Жерздев С.В. Java 2 Micro Edition Нижегородский государственный университет им. Н.И. Лобачевского
4. Методическое пособие по лабораторному практикуму «Робототехника» часть –1 Павлов В.Ю. Кемерово 2003.
5. П. Ноутон, Г.Шилдт Java 2 Наиболее полное руководство: Пер. с англ. – СПб.:БХВ-Петербург, 2006
6. Сосонкин В.Л. Мартинов Г.М. Системы числового программного управления: Учеб. пособие. – М.:Лотос, 2005.