



Иванов Д.С., Нуждин Д.О.,
Овчинников М.Ю., Ткачев С.С.

Система дистанционного
управления стендом для
проведения лабораторных
работ по изучению
движения тела,
подвешенного на струне

Рекомендуемая форма библиографической ссылки: Система дистанционного управления стендом для проведения лабораторных работ по изучению движения тела, подвешенного на струне / Д.С.Иванов [и др.] // Препринты ИПМ им. М.В.Келдыша. 2012. № 59. 32 с. URL: <http://library.keldysh.ru/preprint.asp?id=2012-59>

**Ордена Ленина
ИНСТИТУТ ПРИКЛАДНОЙ МАТЕМАТИКИ
имени М.В. Келдыша
Российской академии наук**

**Д.С. Иванов, Д.О. Нуждин,
М.Ю. Овчинников, С.С. Ткачев**

**Система дистанционного управления стендом
для проведения лабораторных работ
по изучению движения тела,
подвешенного на струне**

Москва — 2012

Д.С. Иванов, Д.О. Нуждин, М.Ю. Овчинников, С.С. Ткачев

Система дистанционного управления стандом для проведения лабораторных работ по изучению движения тела, подвешенного на струне

Описаны система дистанционного управления лабораторным стандом и разработанное программное обеспечение, позволяющее управлять макетом системы ориентации, подвешенным на струне. Приведен пример лабораторных работ, выполняемых с помощью разработанной системы дистанционного управления.

Ключевые слова: дистанционное управление, лабораторный станд, микро-спутник, система управления ориентацией

D.S. Ivanov, D.O. Nuzhdin, M.Yu. Ovchinnikov, S.S. Tkachev

An Installation for Laboratory Facility Remote Operation to Study Controlled Motion of the Body Suspended on the String

Structure of a laboratory facility remote control is developed and presented. To control an attitude motion of the mock-up special software is generated and implemented. Laboratory work using created remote control system is shown as an example.

Key words: remote control, laboratory facility, microsatellite, attitude control system

Работа поддержана Министерством науки и образования РФ и Российским фондом фундаментальных исследований.

Введение

Современный уровень развития телекоммуникационных технологий и средств предоставляет широкие возможности по передаче данных на большие расстояния, созданию распределенных приложений, работающих через сеть, обмену аудио- и видеоинформацией и многое другое [1]. Подобные достижения в последнее десятилетие стали возможными благодаря интеграции компьютерных систем с сетевыми технологиями.

Настоящее время характеризуется стремительным появлением и развитием новых информационных технологий. В частности, технология виртуальных приборов позволяет создавать системы измерения, управления и диагностики различного назначения практически любой производительности и сложности. Ее суть состоит в том, что измерительная и управляющая часть приборов и систем реализуется на аппаратной основе (устройств ввода-вывода аналоговых и цифровых сигналов), а их функциональная часть и пользовательский интерфейс – программными способами.

Преимущество и эффективность виртуальных измерительных технологий состоит в возможности программным путем, опираясь на мощь современной компьютерной техники, создавать разнообразные приборы, измерительные системы и программно-аппаратные комплексы, легко перестраивать их к изменяющимся требованиям, уменьшить материальные затраты и время на разработку. При этом создаваемая измерительная система может быть нужным образом адаптирована для решения поставленных задач с учетом их особенностей.

Использование виртуальных измерительных технологий в современных автоматизированных измерительных системах является устойчивой мировой тенденцией последних лет. Об этом свидетельствует большое количество разработок, а также множество [2–5] отечественных и зарубежных [6–9] публикаций, посвященных решению задач в области автоматизации измерений, контроля и управления техническими и технологическими системами.

При использовании виртуальных измерительных технологий ЭВМ становится неотъемлемым компонентом автоматизированных измерительных и управляющих систем. Это дает возможность аппаратно-программного совмещения измерительных систем с телекоммуникационными сетями и обеспечения дистанционного доступа к измерительному и управляющему оборудованию. Подобная интеграция двух современных информационных технологий, а именно, телекоммуникационных сетевых технологий и технологии виртуальных приборов, качественно и количественно расширяет функциональные возможности систем, построенных на их основе. Это позволяет связывать в единую систему большое число различных измерительных и управляющих устройств, удаленных друг от друга на большие расстояния, а также строить системы дистанционного управления (СДУ) различного назначения.

Благодаря указанным преимуществам, помимо решения промышленных задач автоматизации, виртуальные измерительные технологии находят широ-

кое применение в сфере традиционного и дистанционного (обзору СДУ посвящен раздел 1.2 настоящей работы) инженерного образования. В работе [3] отмечаются следующие преимущества, получаемые за счет использования в учебном процессе виртуальных лабораторий.

- За счет круглосуточной автоматизированной работы дистанционной лаборатории (без преподавателя и лаборанта, нет необходимости в посадочных местах и столах для студентов) достигается экономия учебных площадей, оптимизируется учебное расписание, достигается экономия средств за счет сокращения часов, выделенных на проведение занятий.

- Индивидуализируется и повышается качество обучения, так как студент вынужден выполнять лабораторную работу индивидуально. Появляются качественно новые возможности для самостоятельной работы студентов.

- За счет расширения пространства обучения, не локализованного пределами какого-либо здания, виртуальная лаборатория равнодоступна в любое время для всех кафедр университета, его филиалов, а также индивидуально для студентов, имеющих домашний компьютер и выход в Интернет, что позволяет исключить дублирование лабораторных работ.

- Появляется возможность дистанционного использования виртуальных лабораторий и уникальных установок другими вузами.

В настоящей работе реализована система ДУ на примере лабораторной установки (далее – стенда), созданного в ИПМ им. М.В. Келдыша РАН и установленного на кафедре теоретической механики МФТИ [10], [11]. Разработан и внедрен набор лабораторных работ по управлению макетом системы управления малого спутника, которые апробированы в рамках курса "Управление и динамика сложных информационно-механических систем", используемого для обучения и мотивации молодых исследователей в области теоретической механики и управления движением.

1 Построение телекоммуникационных систем дистанционного управления учебным и научным экспериментом

1.1 Обзор существующих систем дистанционного управления учебными и научными экспериментами

В силу современного уровня развития измерительных, компьютерных и телекоммуникационных технологий и средств, а также их взаимной интеграции, управление реальными физическими объектами дистанционно через телекоммуникационные сети общего пользования (Интернет, ЛВС предприятий) является на сегодняшний день мировой тенденцией. Достаточно большое количество работ в этой области посвящено созданию дистанционных практикумов по инженерным общетехническим специальностям, являющихся неотъемлемой частью дистанционного инженерного образования. Рассмотрим некоторые из них.

1.1.1 AIM- Lab, LAB-on-Web, Cyber Lab

В работе [12] представлены результаты совместных проектов Норвежского университета науки и технологии (NTNU) и Политехнического института города Хьеллер. Первая версия их дистанционной лаборатории *AIM- Lab (Automated Internet Measurement Laboratory)* предназначена для исследования характеристик полевых транзисторов. В этой работе измерительная система построена на базе GPIB-устройств, подключаемых к серверу дистанционной измерительной системы.

Дистанционная лаборатория второй версии (LAB-on-WEB) ориентирована на исследование параметров и характеристик различных радиоэлектронных схем и элементов, реализованных посредством специализированной интегральной схемы Alpha-chip. Эта микросхема была разработана в рамках программы обмена между университетами Европы и Латинской Америки. Она содержит диоды, конденсаторы, транзисторы, инверторы и т.д.

Для обеих версий представленных систем предусмотрена возможность реализации пользовательского интерфейса посредством отдельного клиентского приложения либо в стандартном интернет-обозревателе посредством загрузки и исполнения соответствующего Java-скрипта. В последнем случае представление результатов измерения на стороне удаленного пользователя осуществляется посредством CGI-приложения.

Реализация клиентской части в виде отдельного приложения, а также CGI-приложения на сервере осуществлялась в среде LabVIEW [13].

К недостаткам этой работы можно отнести ограниченную гибкость измерительной системы, связанную с невозможностью задействования дополнительных лабораторий без изменения конфигурации и доработки ПО измерительной системы. Также не приводятся никаких данных о средствах и способах реализации многопользовательского режима работы системы.

В работе [6] представлена система дистанционного управления, разработанная в рамках совместного проекта университетов г. Падуя и г. Турина (Италия). Характерной для этой системы является возможность многопользовательского режима работы, определяющего ее пригодность к полноценному использованию как в научно-технической, так и в образовательной сферах. Эта система представляет собой многоуровневую структуру (рис. 1.1). Многопользовательский режим обеспечивается за счет организации очереди запросов и наличия механизмов ее обработки. Использование TCP/IP протокола для передачи данных между сервером системы и УП, обеспечивает возможность расположения последних в различных географических точках независимо от расстояния.



Рис. 1.1. Многоуровневая структура системы дистанционного управления, разработанная университетами г. Падуя и г. Турина (Италия)

В работе [5] представлена система дистанционных лабораторий *Cyber Lab*, разработанная в Стенфордском университете (США). Для тестирования и исследования эффективности этой системы в техническом и образовательном планах была использована «пилотная» лабораторная работа по исследованию оптических явлений. Экспериментальная часть данной лабораторной работы реализована на основе специализированной лазерной установки.

Большое внимание в работе уделяется методическому обеспечению лабораторных практикумов, контролю уровня знаний студентов перед допуском к выполнению работы, а также вопросам индивидуализации их работы. Одним из концептуальных подходов, применявшихся авторами при разработке и создании *Cyber Lab*, было использование возможностей мультимедийных и Web-технологий, что в сочетании с реальным экспериментированием могло бы в значительной степени повысить эффективность и качество обучения. На рис. 1.2 представлен пользовательский интерфейс созданной дистанционной лаборатории системы *Cyber Lab*, доступный студентам через Интернет.

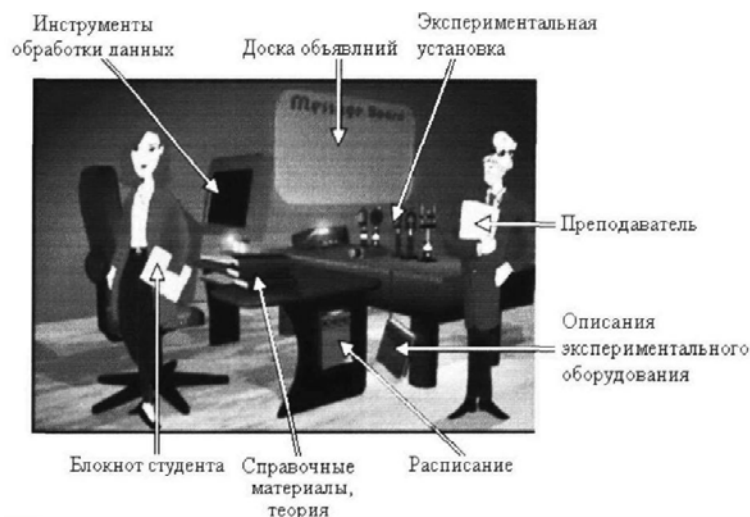


Рис. 1.2. Пользовательский интерфейс дистанционной лаборатории системы *Cyber Lab* с указанием составных компонентов

В работе [14] представлена система дистанционного управления *Remote Lab*, созданная при сотрудничестве Центра инновационного дистанционного образования университета г.Атлантики и колледжа Loyola г.Мэрилэнда (США).

Система *Remote Lab* ориентирована главным образом на использование в дистанционном инженерном образовании для прохождения лабораторных практикумов.

Особенностью системы *Remote Lab* является реализация ПО серверной и клиентской частей посредством языка программирования Java. Это обеспечивает гибкость работы клиентского ПО: для работы удаленного пользователя необходим только стандартный Интернет-обозреватель, осуществляющий загрузку (с сервера системы) и исполнение Java-апплета, реализующего пользовательский интерфейс для дистанционного управления лабораторным экспериментом. Кроме этого, использование Java обеспечивает независимость функциональности клиентского ПО от операционной системы.

ПО сервера данной системы включает в себя несколько блоков *Remote Object*, являющихся программными модулями, взаимодействующими с соответствующими физическими объектами. Блоки *Stab* служат для приема/передачи данных между сервером и удаленным пользователем, в то время как блоки *Skeleton* – между сервером и программным модулем, управляющим экспериментом (*Remote Object*).

К недостаткам рассмотренной системы можно отнести потенциально невысокую функциональность клиентского ПО, что обусловлено спецификой Java-апплетов.

В работе [15], выполненной в Департаменте информатики Университета г. Берлина, представлена система, имеющая то же название, что и в предыдущем рассмотренном случае – *RemoteLab*, предназначенная для дистанционного управления движением робота *Khepera*. Эта система ориентирована на использование в инженерном дистанционном образовании. При ее реализации особое внимание было уделено средствам защиты от некорректных действий студента, ведущих к сбоям либо выходу экспериментального оборудования из строя.

1.1.2 *PEARL, Remote Lab, RVlabX*

В работе [16], выполненной в Trinity College (Ирландия), представлена распределенная система дистанционного управления *PEARL* (Practical Experimentation by Accessible Remote Learning), предназначенная для выполнения экспериментов в дистанционных лабораторных практикумах. В системе *PEARL*, так же как и в ряде рассмотренных ранее случаев, в качестве клиентского ПО используется стандартный Интернет-обозреватель, посредством которого загружается и исполняется соответствующий Java-апплет.

Дистанционная лаборатория (ДЛ) системы *PEARL* строится на основе лабораторного сервера, контролирующего экспериментальное оборудование. Для контроля доступа УП к ресурсам ДЛ используется отдельная серверная машина, выполняющая роль Web-сервера. Использование в СДУ отдельной машины главного сервера обеспечивает возможность создания распределенной измерительной системы, состоящей из нескольких ДЛ. Кроме этого достигается структурирование и централизация системы, а также разделение выполняемых функ-

ций между соответствующими узлами. Функции измерения и управления выполняются лабораторными серверами, а функции контроля доступа УП, а также функции информационной и методической поддержки – главным сервером (ГС).

Организация взаимодействия ГС с ДЛ осуществляется на основании архитектуры CORBA. В отличие от системы *Remote Lab* (г.Берлин, Германия), где канал CORBA был использован для организации взаимодействия УП с непосредственно ДЛ, в системе *PEARL* CORBA используется лишь на участке ГС - ДЛ. При этом передача данных между УП и ГС осуществляется по протоколу HTTP. Такое разделение сетевой архитектуры системы на уровни, использующие различные протоколы (протокол CORBA – участок ГС-ЛС и протокол HTTP – участок УП-ГС) позволяет решить проблему, возникающую при передаче данных в случае наличия в сети сетевых фильтров (firewall) и прокси-серверов, а также участков с малой пропускной способностью.



Рис. 1.3. Структурная схема лабораторной установки дистанционного лабораторного практикума, реализованного в системе PEARL

Первая дистанционная лабораторная работа, реализованная на основе системы *PEARL*, посвящена изучению работы системы технического зрения. Согласно рис. 1.3, основными компонентами лабораторной установки являются: видеокамера; заданное количество независимых источников освещения; стол с ХУ-системой координат, служащий для размещения исследуемых объектов.

Система дистанционного управления *RVLabX* [17], разработанная в университете LICEF-CIRTA г. Монреаля (Канада), так же как и ряд систем, рассмотренных выше, ориентирована на использование в сфере дистанционного инженерного образования.

Отличительной особенностью системы дистанционного управления *RVLabX* является возможность создания удаленными пользователями (как преподавателями, так и студентами) собственных пользовательских интерфейсов, служащих для выполнения дистанционных экспериментов. Ключевыми компонентами *RVLabX*, обеспечивающими основную функциональность данной системы, являются:

- *Interface Designer* - приложение разработки пользовательских интерфейсов;

- *PolymorphiX* – серверный компонент для реализации созданного ПИ как компонента СДУ;
- средства администрирования.

Разработка пользовательского интерфейса посредством приложения *Interface Designer* осуществляется путем размещения на наборном поле соответствующих средств управления и индикации, извлекаемых из палитр. Предусмотрена возможность выбора места расположения и размеров каждого из объектов создаваемой лицевой панели, а также их свойств, связанных с форматом задаваемых/отображаемых данных.

1.1.3 REMOT

В отличие от рассмотренных ранее СДУ, ориентированных главным образом на применение в сфере инженерного образования, в работе [18] представлена система *REMOT* (Remote Experiment Monitoring and control), ориентированная на обеспечение дистанционного управления научными экспериментами. Данная система создана в рамках программы *TELEMATRIX* Европейского комитета DG XIII.

Система *REMOT*, как и каждая из рассмотренных СДУ, построена по принципу «клиент-сервер». Для реализации распределенной архитектуры системы был применен стандарт *CORBA*.

С целью исследования и демонстрации возможностей разработанная система была использована для дистанционного управления экспериментальным оборудованием двух независимых научных лабораторий: лаборатория астрономии (управление телескопом) и лаборатория физики плазмы, содержащая дорогостоящее экспериментальное оборудование. Структурная схема системы *REMOT* представлена на рис. 1.4.

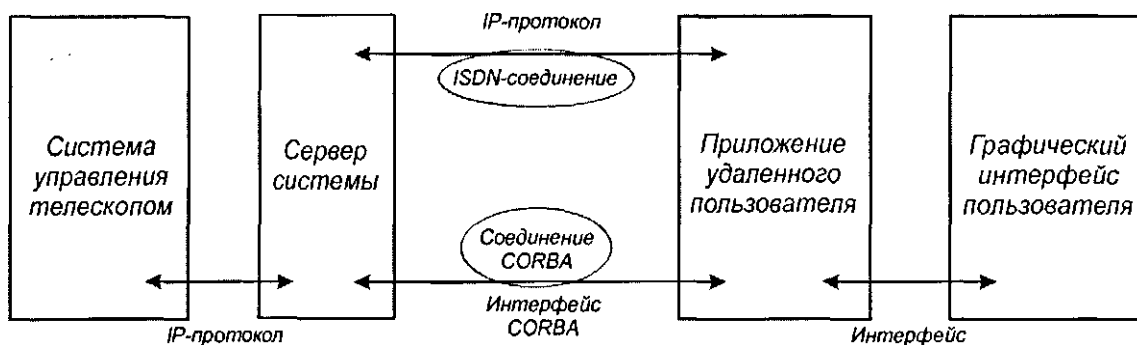


Рис. 1.4. Структурная схема СДУ научными экспериментами *REMOT*

В связи с ориентированием данной системы на использование в научной сфере обеспечиваются лишь средства дистанционного доступа и управления экспериментом, а также представление результатов экспериментов. В работе показаны основные принципы построения системы, однако не приводятся результаты реализации данного проекта, а также описание автоматизированных измерительных систем, применяемых на локальном уровне в лабораториях.

1.1.4 uCV-Lab

В работе [2] представлена дистанционная автоматизированная Web-лаборатория *uCV-Lab* для изучения и исследования микроконтроллеров и сигнальных процессоров. Базовая часть автоматизированной измерительной системы этой Web-лаборатории строится по ставшей на сегодняшний день классической схеме ПК + устройство ввода/вывода. В сочетании со специализированным оценочным модулем обеспечивается возможность проектирования программного обеспечения, а также контроля состояния и исследования рабочих процессов микроконтроллерных устройств и построенных на их основе программно-аппаратных средств.

Клиентское ПО реализовано как ActiveX-объект, встроенный в *html*-страницу. Загрузка этого ПО осуществляется через стандартный Интернет-обозреватель по протоколу HTTP. Обмен данными между серверным и клиентским ПО осуществляется по протоколу DSTP. Компонент «монитор», входящий в состав серверного ПО, выполняет функции управления оценочным модулем.

На основании представленной структуры Web-лаборатории *uCV-Lab* создан центр коллективного пользования, представляющий собой сеть дистанционных лабораторий.

Следует отметить достаточно тщательную проработку вопросов, касающихся методического обеспечения дистанционных работ. Так, например, согласно [2], при подключении к главному серверу центра (диспетчеру) студент выбирает нужную дисциплину, после чего загружается страница с изображением лаборатории со стеллажами, где размещены все доступные для работы измерительные приборы, а также объекты исследований. После выбора требуемой дисциплины посредством указанной выше страницы выполняется компоновка рабочего места путем перемещения на лабораторный стол необходимого оборудования.

К недостаткам системы можно отнести ограничения многопользовательского режима работы, связанные с достаточно небольшим количеством (учитывая востребованность представленных лабораторных дисциплин) удаленных пользователей (не более восьми), способных одновременно получать доступ к ресурсам лабораторий.

1.1.5 Distant Lab 1.0

Одним из наиболее интересных решений представляется система, созданная в Казанском государственном университете [1]. Центральным компонентом системы является главный сервер, обеспечивающий доступ УП к ресурсам ДЛ, а также управляющий работой распределенной измерительной системы, построенной на основе дистанционных лабораторий. Взаимодействие главного сервера с измерительными средствами ДЛ осуществляется по ЛВС. Передача данных от установок к серверу и далее клиенту использует в работе Data-Socket сервер. Передатчик запросов ГС и передатчик результатов ИС осуществляют

запись сформированных для передачи данных в соответствующие разделы Data Socket–сервера. Приемники результатов и запросов соответственно ГС и ИС необходимы для чтения данных, публикуемых в соответствующих разделах Data Socket–сервера, и передачи их клиентской стороне.

Для выполнения работ дистанционных лабораторных практикумов используется клиентское ПО Distant Lab 1.0. Особенностью среды Distant Lab 1.0 является то, что количество доступных лабораторных работ не является фиксированным – непосредственно «тело» лабораторной работы (ее функциональность, математические модели исследуемых схем и устройств, их графические изображения и пр.) не является внедренным в программный код среды Distant Lab 1.0, а реализуется посредством отдельной Jb-библиотеки, которая инициализируется в Distant Lab 1.0 перед началом работы. Указанная особенность позволяет легче увеличивать количество доступных дистанционных работ (каждой лабораторной работе соответствует отдельная Jb-библиотека), а также при необходимости модернизировать ранее созданные, исключая трудоемкие модификации исходной среды Distant Lab 1.0 и, как следствие, обновление ее новой версией. Другой важной особенностью Distant Lab 1.0 является гибкость – расположение на лицевой панели виртуальных электронно-измерительных приборов, таких, как многоканальный осциллограф и функциональный генератор, вызываемых из специальной библиотеки, их подключение к исследуемой схеме не являются фиксированными и производятся студентом в соответствии с методикой лабораторной работы. Стоит заметить, что описанная система совмещает в себе средства моделирования с лабораторными установками, которые задействуются по необходимости.

В табл. 1 приведены основные данные по рассмотренным системам дистанционного управления учебными и научными экспериментами.

Таблица 1

Системы СДУ

	Название СДУ	Расположение ДЛ	Используемые телекоммуникац. технологии	Ориентированность	Особенности
1	<i>AIM-Lab</i>	г.Хьеллер, Норвегия	Средства Java, Lab VIEW. CGI, HTML.	Изучение свойств схем на полевом транзисторе	Возможность создания только одной дистанционной лаборатории вследствие неразделенности серверного и измерительного уровней системы
2	<i>LAB-on-WEB</i>	г.Хьеллер, Норвегия	Технологии ActiveX/DCOM	Изучение характеристик и параметров различных радиоэлектронных схем	
3	<i>без названия</i>	г.Турин, г. Падуя, Италия	Протокол TCP/IP	Представлена общая концепция создания распределенных измерительных систем	

4	<i>Cyber Lab</i>	г.Стенфорд, США	Описание технологий не приведено, ориентировочно ActiveX, HTTP	Изучение оптических явлений на основе лазерной установки	Уделяется большое внимание вопросам методической и информационной поддержки, а также контролю знаний удаленных пользователей
5	<i>Remote Lab</i>	г.Мэриленд, США	RMI, Java, HTTP	Дистанционная работа с реальными объектами (для дистанционного инженерного образования)	Делается акцент на возможностях использования технологии ИМ1 при построении системы дистанционного управления
6	<i>Remote Lab</i>	г.Берлин, Германия	HTTP, CORBA	Управление движением робота	Возможность загрузки на сервер системы собственных алгоритмов управления роботом. Обеспечивается режим защиты оборудования от некорректных действий пользователей
7	<i>PEARL</i>	г.Дублин, Ирландия	HTTP, Java, CORBA	Дистанционная работа с реальными объектами (для дистанционного инженерного образования)	Возможность создания набора дистанционных лабораторий вуза. Решение проблем присутствия проху-серверов и сетевых фильтров
8	<i>RVLabX</i>	г.Монреаль, Канада	HTTP, ActiveX, XML	Дистанционная работа с реальными объектами (для дистанционного инженерного образования)	Возможность быстрого создания собственного пользовательского интерфейса для дистанционных лаб. практикумов
9	<i>REMOT</i>	При сотрудничестве вузов Европы	HTTP, CORBA	Доступ и дистанционное управление оборудованием обсерватории и лаборатории физики плазмы	Ориентированность только на научные исследования

10	<i>uCV-Lab</i>	г.Новосибирск, Россия	HTTP, DSTP, ActiveX, средства LabVIEW	Исследование микропроцессорных систем	Использование измерительной системы, позволяющей работать со сложными устройствами. Возможность увеличения числа дистанционных лабораторий
11	<i>Distant Lab 1.0</i>	г.Казань, Россия	LabVIEW, HTTP, Data Socket	Дистанционные лаборатории по электронике	Простота увеличения количества работ и числа дистанционных лабораторий

1.2 Вывод по результатам обзора

По результатам исследования различных систем для проведения дистанционных лабораторных работ были выделены основные требования к создаваемой системе. В число этих требований входят:

- управление индивидуальным, не имеющим аналогов, объектом управления;
- непосредственный доступ к объекту управления;
- несколько режимов наблюдения (Web-камера, трехмерное моделирование);
- обработка алгоритмов непосредственно на объекте управления;
- удаленное управление;
- многопользовательский режим доступа с возможностью выбора роли и разграничением прав;
- возможность работы через сеть Internet.

В соответствии с этими требованиями разработана система удаленного управления стендом на кафедре теоретической механики МФТИ. Разработанная система позволяет выполнять весь комплекс задач лабораторных работ по управлению ориентацией макета системы управления. Описание разработанной системы дистанционного управления приведено в главе 2 настоящей работы.

1.3 Реализация необходимой программной среды для лаборатории в МФТИ

В этом разделе описана структура СДУ и описан алгоритм её работы. Так как в настоящей работе СДУ рассматривается главным образом применительно к научно-техническому образованию, то автоматизированную измерительную систему в совокупности с лабораторным макетом можно назвать дистанционной лабораторией (ДЛ).

1.3.1 Принцип работы СДУ

Структурная схема СДУ приведена на рис. 1.5. Она имеет двухуровневую архитектуру и реализуется по модели клиент-сервер. В качестве сервера выступает сам лабораторный макет с бортовым компьютером, к которому подключаются пользователи. Такой подход позволяет исключить отдельный сервер для управления макетом, но ограничивает количество одновременно подключенных клиентов. Поскольку лабораторная работа направлена на индивидуальное обучение, ограниченность одновременно подключенных пользователей можно не принимать во внимание.

Взаимодействие клиента с сервером осуществляется по сети Интернет, локальной сети или сети Wi-Fi в зависимости от места расположения пользователя.

Наблюдение за прохождением эксперимента реализуется двумя способами. Первый способ осуществляется с использованием независимой Web-камеры. Пользователь может подключиться к камере, которая установлена в лаборатории, и наблюдать за движением макета в реальном времени по видеоканалу. Независимость заключается в том, что она физически никак не связана с лабораторным макетом. Таким образом, даже при выключенном бортовом компьютере существует возможность внешнего наблюдения за макетом.

Второй способ использует трехмерную модель объекта и данные, поступающие с бортового компьютера макета. При соединении с макетом по получаемым значениям угла и угловой скорости клиентская программа определяет необходимый угол поворота встроенной трехмерной модели макета и отображает ориентацию макета. Этот способ – лишь наглядная визуализация получаемых данных, но это значительно облегчает процесс наблюдения.

Подключение пользователя к макету возможно в двух режимах: в режиме «Администратора» и в режиме «Наблюдателя». Такое разграничение введено для совместного одновременного проведения эксперимента несколькими студентами. В режиме «Администратора» доступны функции управления проведением эксперимента. «Администратор» управляет включением алгоритмов и настройкой параметров макета. К макету может быть подключен только один «Администратор», при попытке подключения второго «Администратора» система предложит подключиться в режиме «Наблюдатель» либо отключиться. В режиме «Наблюдатель» пользователь может только наблюдать за проведением эксперимента и получать данные. Таким образом, один подключенный пользователь не сможет мешать другому при проведении эксперимента.

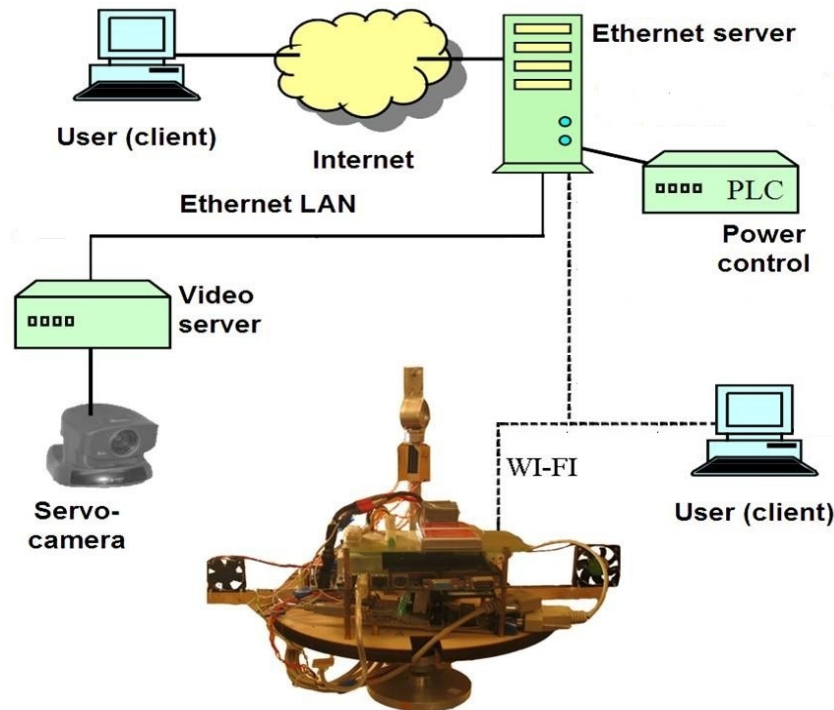


Рис. 1.5. Структурная схема СДУ

1.3.2 Программная модель СДУ

Выбор среды программирования

Качество программного обеспечения определяется главным образом рациональностью программных решений, применяемых в процессе его разработки. Эта рациональность существенным образом может зависеть от качества совокупного решения тех или иных задач, а также от качества интеграции компонентов различных программных сред. Потенциально наибольшая оптимальность может быть достигнута при реализации всех поставленных задач в одной программной среде. Таким образом, для реализации программной модели и непосредственно самой СДУ следует выбрать одну среду программирования. При разработке и создании систем дистанционного управления, как правило, решаются следующие задачи:

- автоматизация измерений на локальном уровне;
- математическая обработка и представление информации;
- создание гибкого пользовательского интерфейса;
- организация передачи данных по телекоммуникационным сетям (запросов на измерения и результатов измерений);
- организация работы с базами данных.

Одной из сред программирования, позволяющей оптимально решать указанные задачи, является среда Borland C++ Builder.

Программное обеспечение

Как уже упоминалось ранее, в разработанном программном обеспечении используется модель клиент-сервер. В роли сервера выступает бортовой компьютер лабораторного макета. В роли клиента – пользовательская клиентская программа, соответственно. Связь осуществляется напрямую, посредством протокола TCP/IP.

При передаче данных между клиентом и сервером существует два вида пакетов: данные и управление. Пакеты с данными пересылаются от сервера клиенту во время выполнения какого-либо активного алгоритма. В такой пакет входят 20 чисел:

- 0 – datatype (тип пакета);
- 1 – state (состояние макета/выполняемый алгоритм);
- 2 – FanStatus (состояние вентиляторов);
- 3 – RotorStatus (состояние маховика);
- 10 – t (время с момента начала выполнения алгоритма);
- 11 – alpha (deg) (угол поворота макета);
- 12 – omega (deg/s) (угловая скорость макета);
- 13 – DegKal (оценка угла, полученная с использованием фильтра Калмана, основанного на модели перманентного вращения);
- 14 – RotKal (оценка угловой скорости, полученная с использованием фильтра Калмана, основанного на модели перманентного вращения);
- 15 – DegKal2 (оценка угла, полученная с использованием фильтра Калмана, основанного на модели крутильных колебаний);
- 16 – RotKal2 (оценка угловой скорости, полученная с использованием фильтра Калмана, основанного на модели крутильных колебаний).

Пропущенные номера зарезервированы для данных, необходимых для новых лабораторных работ.

Пакеты с управлением пересылаются от клиента к серверу при необходимости настройки или смены режима и от сервера к клиенту для обновления настроек клиента.

При проведении эксперимента все входящие данные сохраняются на клиентские компьютеры, а при окончании эксперимента можно выбрать путь, по которому будут сохранены результаты.

При изменении настроек с клиента отправляется пакет с настройками, которые применяются на серверной программе, после чего настройки рассылаются всем остальным клиентам.

2 Система для дистанционного проведения лабораторных работ

2.1 Описание стенда

Рассмотрим схему макета системы ориентации на лопастных двигателях, созданного на кафедре теоретической механики в МФТИ. Макет представляет собой автономный аппарат, включающий в себя систему электропитания, сис-

тему определения ориентации, систему управления ориентацией и систему связи. Схема макета изображена на рис. 2.1, внешний вид макета – на рис. 2.2.

Система электропитания включает в себя два аккумулятора и преобразователь напряжения. Преобразователь из 14 вольт от аккумуляторов формирует 5 вольт, необходимых для питания бортового компьютера и 12 вольт для питания управляющей платы.

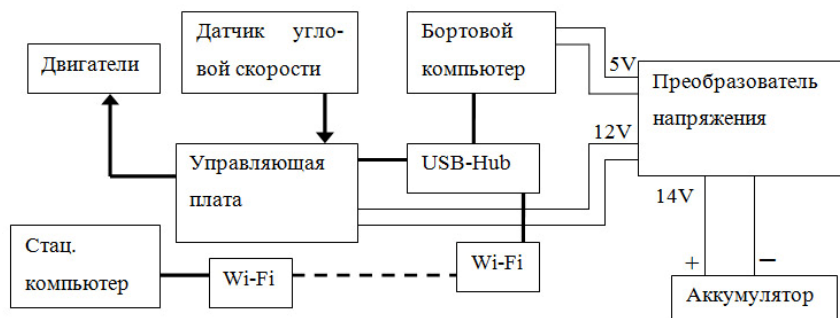


Рис. 2.1. Схема макета

Система определения ориентации состоит из бортового компьютера JReX CE (далее БК) и оптоволоконного датчика угловой скорости VG-910D. БК производит считывание информации об угловой скорости с датчика угловой скорости, обрабатывает её в режиме реального времени и передает данные. Передача информации производится через USB порты БК, используется разветвитель USB-Hub.

Система управления ориентацией состоит из двух лопастных двигателей, с помощью которых создается управляющий момент. БК выдает на управляющую плату команду на включение одного из двигателей, управляющая плата преобразует команду в напряжение, которое подается на лопастные двигатели.

Система связи состоит из устройства Wi-Fi и БК. Устройство Wi-Fi позволяет организовать доступ к БК через удалённый рабочий стол и подключиться клиентской программе. Управление программным обеспечением происходит со стационарного компьютера через удалённый рабочий стол БК или через клиентскую программу.

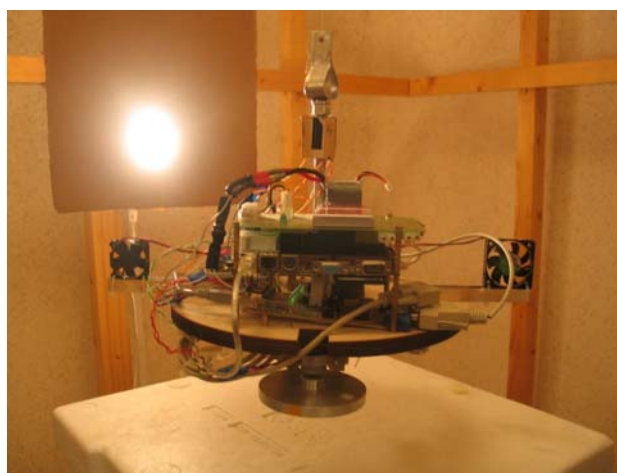


Рис. 2.2. Внешний вид макета

2.2 Программное обеспечение макета

Программное обеспечение – это комплекс из двух программ. Первая из них, установленная на макете (серверная часть), осуществляет считывание информации с датчиков макета, её обработку и передачу результатов второй программе. Вторая программа (клиентская часть) установлена на пользовательском компьютере, осуществляет прием данных, их визуализацию и сохранение.

2.2.1 Программа на макете (серверная часть)

Работа программы

С датчика поступает цифровой сигнал, который умножением на некоторый коэффициент и смещением (для калибровки) преобразуется в угловую скорость. Значения угловой скорости за несколько измерений суммируются, после чего эта сумма делится на количество измерений. Таким образом, на выходе получаем усреднённое значение угловой скорости. Усреднение производится для того, чтобы уменьшить шум измерений и уменьшить число вычислений программы в единицу времени. У программы есть верхний предел по частоте выходных данных ≈ 100 Гц. При частоте выше этого значения программа не успевает обрабатывать поступающие с датчика данные, и работа в режиме реального времени становится невозможной. Поэтому при установке параметров работы программы нужно отдельно обратить внимание на усреднение: зная частоту подачи данных, организовывать такое усреднение (то есть выбрать количество измерений для усреднения), чтобы частота выходных данных программы не превышала установленного предела.

Усреднённая угловая скорость поступает на вход сразу трём программным модулям: интегрирование кинематического уравнения, Фильтр Калмана (ФК) модели перманентного вращения, ФК модели крутильных колебаний. На выходе соответственно имеем: результат интегрирования, апостериори оценки угла отклонения макета от положения равновесия и угловой скорости двух ФК. Все эти значения, а также усреднённая угловая скорость и угол передаются клиентской программе.

Интерфейс программы

Главное окно программы содержит три области (рис. 2.3). Рассмотрим их.

Port settings – установка рабочего датчика, выбор порта, через который будет поступать информация, установка порта управления и частоты отправки данных клиентским программам. Задав эти параметры программы, можно нажать кнопку “Open port” – программа подключится к датчикам и автоматически начнется процесс вычисления.

Control Settings – управление алгоритмом действия макета (эти кнопки дублируют аналогичные на клиенте).

Some information – содержит информацию о выполняемом алгоритме, числе принятых байт в одной посылке с датчика, текущем времени эксперимента, текущей угловой скорости, количестве присоединенных пользователей и сведений об администраторе.

Главное окно содержит меню со следующими разделами:

File – выход из программы;

Sensor calibration – установка калибровочных коэффициентов для датчиков VG-910D и ADIS16100;

Initial conditions – задание начального вектора состояния макета;

Model of measuring – задание параметров модели измерений (квадрата ошибки измерений σ^2 и значений элементов начальной матрицы ошибки для ФК) и размер выборки N для алгоритма “Уточнение ковариации”. Если N=0, то “Уточнение ковариации” отключено;

Dynamic model – задание периода крутильных колебаний.

Управление программным обеспечением

Управление программным обеспечением осуществляется через интерфейс программы (рис.2.3). Перед запуском программы необходимо произвести следующие действия:

- в области Port settings в выпадающем окошке Choose sensor выбрать датчик, с которого будет производиться чтение (VG-9, волоконно-оптический датчик или Mems, микромеханический датчик);
- в окошке Sensor port задать COM-порт, откуда будет считываться информация;
- в окошке Actuator port задать COM-порт, через который будет осуществляться управление;
- установить значение усреднения Averaging;
- установить частоту отправки данных SendDataFreq;
- запустить программу, нажав на кнопку Open port. Программное обеспечение производит вычисления и отправляет данные подключенным клиентам.

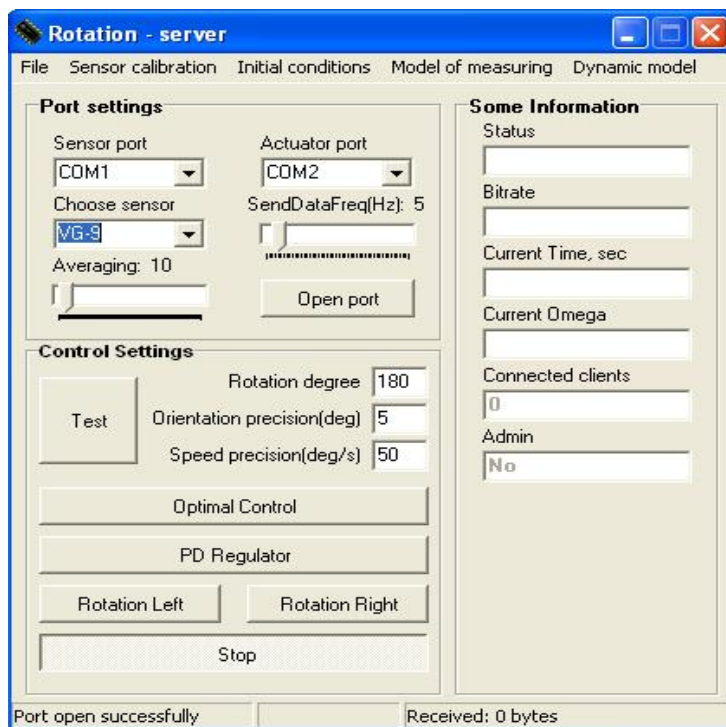


Рис.2.3.Интерфейс серверной части

2.2.2 Пользовательская программа (клиентская часть)

Работа программы

Все входные значения клиентской программы записываются в два вида файлов в папке Results, которая создается в директории клиентской программы. Первый вид файла с названием “Дата-Время_AllSessionReceiveddata.txt”, где «Дата» – это текущая дата, а «Время» – время запуска клиентской программы, содержит все измерения, полученные за то время, пока программа была запущена. Второй вид файла с названием “Дата-Время_ТипПрограммы-Receiveddata.txt”, где дата и время аналогичны первому файлу, а «ТипПрограммы» – это тип выполняемого алгоритма управления, создается при изменении типа работы, то есть фактически при выполнении каждого отдельного эксперимента, это позволяет получить сразу результаты каждого эксперимента отдельно. Каждый блок значений записывается в отдельной строке.

Также значения обрабатываются и выводятся в виде информации о выполняемом эксперименте, в виде графиков результатов в реальном времени и значений в таблицах. Также программа позволяет удаленно наблюдать за макетом с помощью Web-камеры либо наблюдать за движением трехмерной модели макета, отражающей получаемые данные угла и угловой скорости.

2.2.3 Интерфейс программы

Главное окно программы содержит четыре области (рис.2.4):

Connect properties – настройка соединения с макетом (серверной частью программы). Здесь настраивается адрес сервера и режим «администратора» для управления, либо в режиме «клиента» для просмотра поступающих данных.

Status information – содержит информацию о выполняемом алгоритме, текущем времени эксперимента, текущем угле поворота макета и текущей угловой скорости.

Lab1, Lab2, Lab3 – вкладки управления режимами макета, с помощью которых можно задавать режимы работы макета.

VideoCam, 3DModel – вкладки наблюдения за макетом в режиме реального времени через Web-камеру, либо в режиме 3D модели.

Главное окно содержит меню со следующими разделами:

Device Settings – установка рабочего датчика, выбор порта, через который будет поступать информация, и выбор порта управления.

Device calibration – установка калибровочных коэффициентов для датчиков угловой скорости и вентиляторов.

Initial conditions – задание начального вектора состояния макета.

Dynamic model – задание периода крутильных колебаний.

Models settings – задание начального вектора состояния макета, периода крутильных колебаний и параметров модели измерений (квадрата ошибки измерений σ^2 и значений элементов матрицы ошибки для ФК).

Graphics – вывод в режиме реального времени графиков включения вентиляторов, угловой скорости и угла отклонения от положения равновесия. Графики отображают все выходные данные программы.

Output values – вывод на экран принимаемых данных, выходных значений двух фильтров и результата интегрирования.

File – выход из программы.

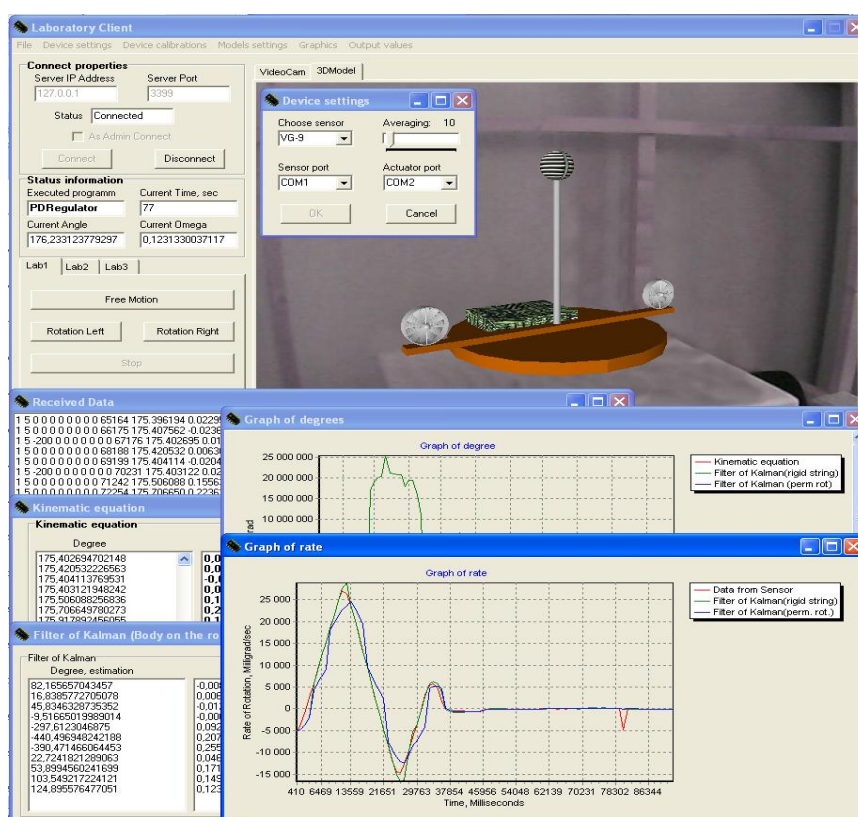


Рис.2.4. Интерфейс клиентского ПО

Управление программным обеспечением

Управление клиентским программным обеспечением осуществляется при помощи интерфейса программы (рис. 2.4). Перед запуском лабораторной работы необходимо выполнить следующие операции:

- В области *Connect Properties* ввести адрес макета и порт 3399. Если планируется управление проведением лабораторной работы, нужно подключиться в режиме «Администратор». Для этого выделить метку «As Admin Connect», если планируется просто наблюдать, метку нужно оставить невыделенной (к серверу может подключиться только один управляющий, остальные подключаются наблюдателями). Далее нужно нажать кнопку *Connect* и дождаться состояния «Connected» в поле *Status*.

- При подключении в режиме «Администратор» во вкладке главного меню *Device Settings* можно изменить параметр *Sensor port* – COM-порт, откуда будет считываться информация, в окне *Actuator port* задать COM-порт, через который будет осуществляться управление. Также в пункте *Choose sensor* вы-

бирается тип подключенного ДУС (VG-9 или MEMS), а ползунок *Averaging* устанавливает усреднение для принимаемых данных на серверной части. Эти настройки дублируют и меняют настройки области *Port Settings* на серверной части программы.

- При подключении в режиме «Администратор» во вкладке главного меню *Device Calibrations* проверить настройки ДУСов и вентиляторов.

- При подключении в режиме «Администратор» во вкладке главного меню *Models Settings* в разделе меню *Initial conditions* проверить начальный угол отклонения от положения равновесия в градусах (если макет закручен по часовой стрелке, то добавляется знак минус). В разделе *Dynamic model* проверить период крутильных колебаний. В разделе *Model of measuring* проверить значение квадрата ошибки σ^2 для заданного усреднения датчика и размер выборки N для алгоритма “Уточнение ковариации”. Если $N=0$, то “Уточнение ковариации” отключено.

- Выбрать режим наблюдения через Web-камеру или 3D-модель и выбрать соответствующую вкладку *VideoCam* или *3DModel*. При выборе режима веб-камеры, нужно подключить ее, настроив необходимые параметры и нажав клавишу *Play/Stop*, находящиеся в области *Camera Settings* под изображением. Если выбран режим 3D-модели, можно включить или отключить сглаживание движений кнопкой *Smoothing*.

- Для начала эксперимента необходимо выбрать соответствующую вкладку *Lab1/Lab2/Lab3* и нажать на кнопку необходимого эксперимента. Программное обеспечение начнет производить вычисления и прием-передачу данных.

Полученные данные, а также результаты вычислений отображаются на графиках раздела меню *Graphics* и в значениях в разделе *Output values*.

3 Проведение лабораторных работ

В качестве примера рассмотрим выполнение лабораторной работы по исследованию алгоритмов управления ориентацией с помощью разработанной системы удаленного управления стендом на кафедре теоретической механики МФТИ.

3.1 Порядок выполнения лабораторных работ

3.1.1 Общие действия при проведении локальной лабораторной работы

1. Проверить, что аккумуляторы надежно соединены проводами с платой питания. Проверить правильность подключения аккумуляторов: черный провод всегда соответствует минусу, поэтому он должен соединяться с черным проводом аккумулятора и входить в разъем платы с подписью “-”; провода с любым другим цветом (кроме белого, который тоже иногда соответствует “-”) обозначают полярность “+” и должны подключаться соответственно.

2. Включить БК. Проверить, что на макете зажглась зеленая лампочка – индикатор включения питания.

3. Включить стационарный компьютер. Подключиться к БК макета и запустить серверную часть программы.

4. Отключиться от БК макета и запустить клиентскую часть программы.

5. Настроить параметры соединения в области *Connect Properties*, выбрать свою роль (администратор – управляет проведением эксперимента или наблюдатель – просто получает данные), соответственно установкой или не установкой метки «As Admin Connect». Далее нужно нажать кнопку «Connect» и дождаться состояния «Connected» в поле «Status».

3.1.2 Общие действия при проведении удаленной лабораторной работы

1. Запустить клиентскую часть программы.

2. Настроить параметры соединения в области *Connect Properties*, выбрать свою роль (администратор или наблюдатель), соответственно, установкой или не установкой метки «As Admin Connect». Далее нужно нажать кнопку «Connect» и дождаться состояния «Connected» в поле «Status».

3. Для наблюдения за экспериментом с помощью Web-камеры выбрать вкладку *VideoCam* в правой части программы, настроить параметры камеры в области *Camera Settings* и нажать клавишу *Play/Stop*.

4. Для наблюдения за экспериментом по 3D модели выбрать вкладку *3DModel* в правой части программы. При наблюдении можно включить или отключить сглаживание движений кнопкой *Smoothing*.

3.1.3 Исследование алгоритмов управления ориентацией

Дальнейшие действия, связанные с запуском экспериментов или управлением макетом, доступны только при подключении в статусе «Администратор». В роли наблюдателя возможно только получение данных и наблюдение за проведением эксперимента.

1. Выбрать вкладку *Lab2*.

2. Задать параметры лопастных двигателей в программе. Для этого нужно зайти в меню *Device Calibrations* и выбрать *Coolers(OptControl)*. Затем нужно задать значения параметров двигателей, соответствующих модели, из раздела 1.1 и нажать кнопку *OK*.

3. Задать разворот на 180 градусов в окошке *Rotation Degree*.

4. Задать точность поддержания угла для оптимального алгоритма в окошке *Orientation Precision(deg)* в градусах.

5. Задать точность поддержания скорости для оптимального алгоритма в окошке *Speed Precision(deg/s)* в градус/секунда.

6. Запустить программу управления нажатием на кнопку *Optimal Control*. В области *Status Information* в окошке *Executed Programm* должна появиться надпись «OptimalControl». А в других окнах должна отображаться информация о

текущем времени эксперимента (*Current Time*), текущем угле поворота (*Current Angle*) и текущей угловой скорости макета (*Current Omega*). Зайти в меню *Graphics* и выбрать *Graph of Coolers*. График будет отображать текущее состояние работы вентиляторов. Зайти в меню *Graphics* и выбрать *Graph of Degree*. График будет отображать текущее значение угла поворота. Зайти в меню *Graphics* и выбрать *Graph of Rate of Rotation*. График будет отображать текущее значение выдаваемой датчиком угловой скорости и оценки угловой скорости при помощи фильтров Калмана. Прекратить программу, когда визуально аппарат выйдет на предельный цикл, нажатием клавиши *Stop*.

7. Задать коэффициенты k_ω и k_α PD-регулятора, соответствующие максимальной степени устойчивости. Для этого нужно зайти в меню *Device Calibrations* и выбрать *Coolers(PDReg)*. Затем нужно задать значения и нажать кнопку *OK*.

8. Задать разворот на 180 градусов в окошке *Rotation Degree*.

9. Запустить программу управления нажатием на кнопку *PD Regulator*. В области *Status Information* в окошке *Executed Program* должна появиться надпись «PDRegulator». А в других окнах должна отображаться информация о текущем времени эксперимента (*Current Time*), текущем угле поворота (*Current Angle*) и текущей угловой скорости макета (*Current Omega*). Зайти в меню *Graphics* и выбрать *Graph of Coolers*. График будет отображать текущее состояние работы вентиляторов. Зайти в меню *Graphics* и выбрать *Graph of Degree*. График будет отображать текущее значение угла поворота. Зайти в меню *Graphics* и выбрать *Graph of Rate of Rotation*. График будет отображать текущее значение выдаваемой датчиком угловой скорости и оценки угловой скорости фильтров Калмана. Прекратить программу, когда визуально аппарат выйдет на предельный цикл, нажатием клавиши *Stop*.

10. Повторить пункты 7-9 с коэффициентами k_ω и k_α PD-регулятора, при которых корни характеристического уравнения будут мнимыми.

11. Построить в Matlab графики траектории движения на фазовой плоскости и определить точность единичного маневра, время маневра и точность поддержания ориентации (определить по размерам предельного цикла) для двух алгоритмов. Для PD-регулятора точность единичного маневра считать в моменты, когда траектория впервые достигнет точек, в которых угловая скорость и рассогласование по углу равны нулю (значения угла и угловой скорости в соответственных точках будут значениями точностей маневра).

3.2 Результаты проведения лабораторной работы "Исследование алгоритмов управления ориентацией"

При проведении лабораторной работы "Исследование алгоритмов управления ориентацией" преследуется цель получить оценку точности работы алгоритмов управления, сравнить качество работы алгоритма на основе PD-регулятора и оптимального по быстродействию алгоритма. Лабораторная работа выполняется в соответствии с порядком, описанным в разделе 3.1.3.

На рис. 3.1 представлен график траектории на фазовой плоскости при работе оптимального по быстродействию управления, на рис. 3.2 представлен график управления в зависимости от времени. Из-за неучета импульса последодействия в алгоритме траектория выходит на предельный цикл. По размерам предельного цикла можно оценить точность поддержания заданной ориентации, которая составляет около 15° , и по скорости около $7^{\circ}/\text{сек}$.

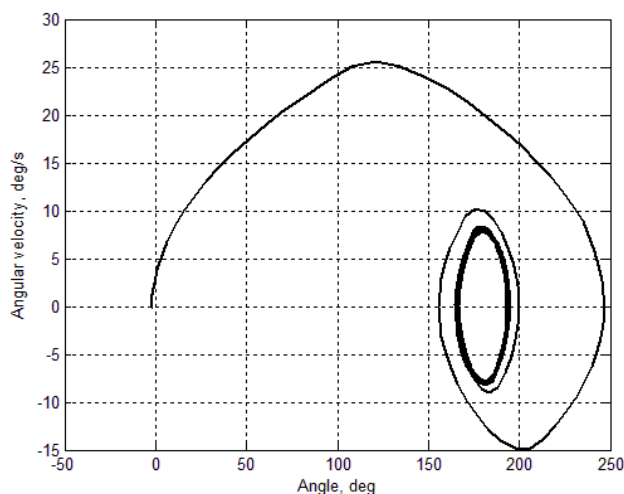


Рис. 3.1. Траектория на фазовой плоскости при работе алгоритма оптимального управления

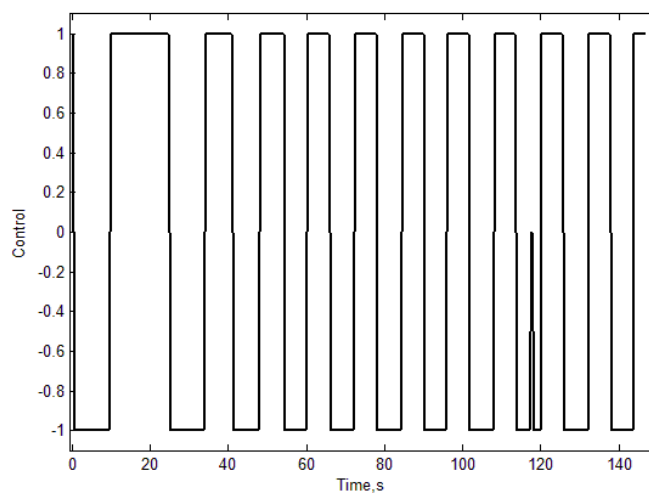


Рис. 3.2. Управление в зависимости от времени при работе алгоритма оптимального управления

На рис. 3.3 представлен график работы PD-регулятора с коэффициентами, обуславливающими затухающие колебания ($k_{\alpha} = 0.04$, $k_{\omega} = 0.2$). На рис. 3.4 представлен график зависимости управления от времени. Вследствие невысокой степени устойчивости траектория макета достаточно продолжительно приходит в окрестность заданного положения в 180° . Но после прихода в нужную точку на фазовой плоскости точность поддержания составляет порядка 3° по углу и $1^{\circ}/\text{с}$ по угловой скорости.

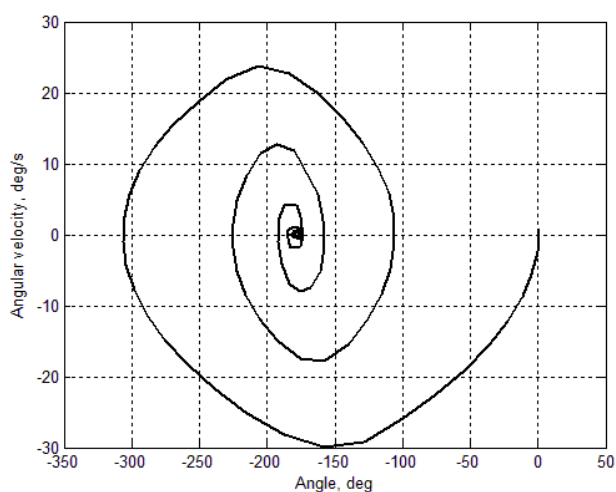


Рис. 3.3. Траектория на фазовой плоскости при работе PD-регулятора с коэффициентами, обеспечивающими затухающие колебания

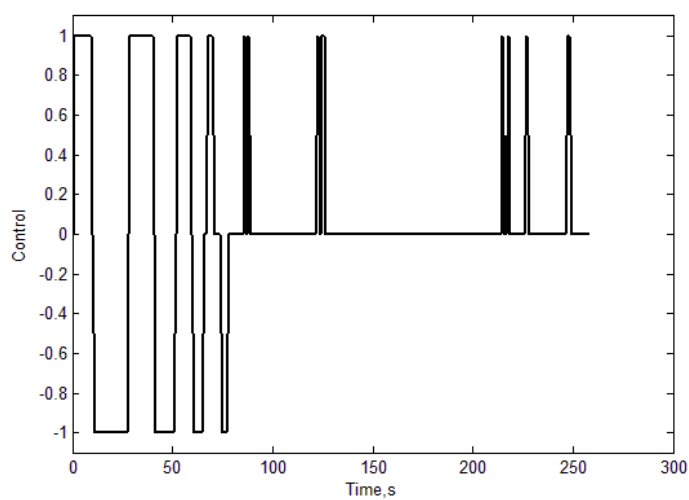


Рис. 3.4. Управление в зависимости от времени при работе PD-регулятора с коэффициентами, обеспечивающими затухающие колебания

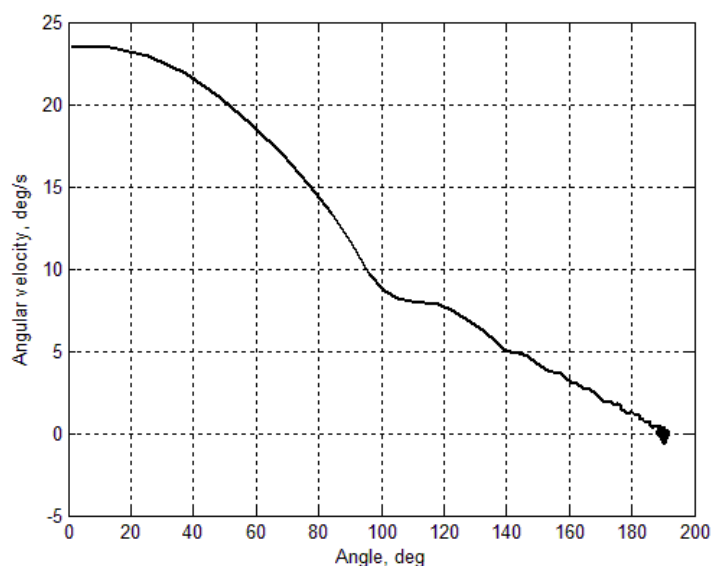


Рис. 3.5. Траектория на фазовой плоскости при работе PD-регулятора с коэффициентами, обеспечивающими максимальную степень устойчивости

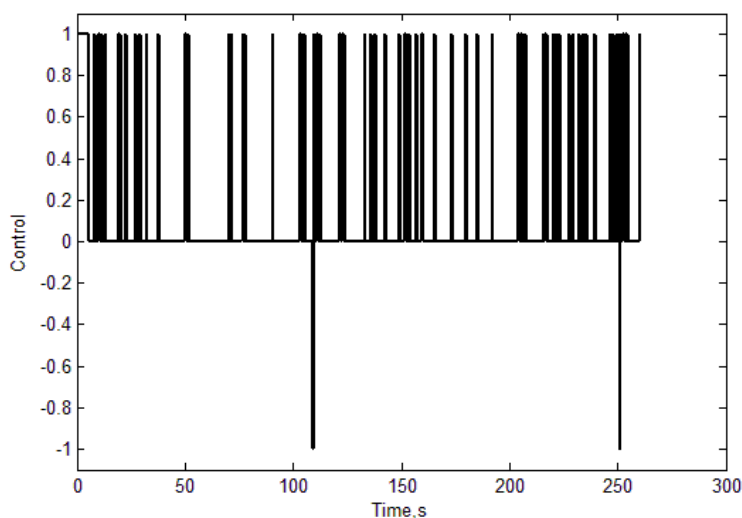


Рис. 3.6. Управление в зависимости от времени при работе PD-регулятора с коэффициентами, обеспечивающими максимальную степень устойчивости

При задании коэффициентов PD-регулятора, обеспечивающих максимальную степень устойчивости, $k_\alpha = 0.01$, $k_\omega = 0.2$ (мнимая часть корней характеристического члена равна нулю, корни имеют только отрицательную действительную часть), точка на фазовой плоскости по траектории, близкой к экспоненциальной, приходит в заданное положение на фазовой плоскости (рис. 3.5) за время приблизительно 50 с. График управления представлен на рис. 3.6. При поддержании положения достигается точность 1.0° по положению и $0.5^\circ/\text{с}$ по угловой скорости.

Все характеристики алгоритмов управления сведены в табл. 2.

Таблица 2

Характеристики работы алгоритмов управления ориентацией

Параметр	Оптимальный алгоритм	PD-регулятор 1	PD-регулятор 2
Точность единичного маневра поворота на 180° , градусы и угловая скорость	25° , $15^{\circ}/с$	100° , $20^{\circ}/с$	5° , $0.7^{\circ}/с$
Время маневра, с	25	25	50
Точность поддержания положения, градусы и угловая скорость	15° , $7^{\circ}/с$	3° , $1^{\circ}/с$	1.0° , $0.5^{\circ}/с$

Таким образом, оптимальный алгоритм имеет самую низкую точность вследствие того, что в модели вентиляторов не учтен импульс последействия, но у него самое короткое время единичного маневра. При работе PD-регулятора с коэффициентами, обеспечивающими затухание колебаний, точность единичного маневра намного хуже точности, достигаемой при оптимальном управлении при том же времени, но точность поддержания значительно лучше. Если же выбрать коэффициенты, обеспечивающие максимальную степень устойчивости, то PD-регулятор достаточно долго приводит систему в нужное положение, но точность поддержания самая высокая.

Заключение

Описана разработанная система дистанционного управления стендом, который инсталлирован на кафедре теоретической механики МФТИ. Эта система удовлетворяет основным требованиям, предъявляемым к дистанционной лаборатории: позволяет управлять макетом системы ориентации, обеспечивает непосредственный доступ к объекту управления посредством сети Internet. В качестве достоинств разработанной системы можно выделить:

- исследуемые алгоритмы управления выполняются непосредственно на бортовом компьютере макета, а удаленно подается только команда на запуск того или иного алгоритма;
- имеется несколько режимов наблюдения: с помощью Web-камеры и по передаваемым данным об ориентации с помощью построения трехмерного изображения макета;
- лабораторная работа может выполняться удаленно одновременно группой студентов, причем только один в режиме "Администратор" имеет доступ для запуска исследуемого алгоритма, а остальные имеют доступ только для получения данных в режиме "Наблюдатель", что позволяет избежать конфликта интересов при одновременном управлении макетом.

Разработанная система дистанционного управления совмещает в себе все основные преимущества аналогичных систем, описанных в разделе 1.1, и вне-

дряется в курс "Управление и динамика сложных информационно-механических систем" для проведения лабораторных работ по исследованию качества работы алгоритмов управления ориентацией [19].

Список литературы

1. Евдокимов Ю.К., Маливанов Н.Н., Кирсанов А.Ф. Разработка и создание региональной дистанционной автоматизированной учебной лаборатории коллективного пользования по общетехническим дисциплинам для ВУЗов и средних учебных заведений республики Татарстан на базе технологии National Instruments // Сб. тр. Междунар. науч. практич. конф. «Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments». Москва. 17-18 ноября, 2006. С. 5-10.
2. Баран Е.Д., Захаров П.М., Любенко А.Ю. Web-лаборатория «Микроконтроллеры и сигнальные процессоры» // Современные технологии автоматизации. 2005. Т. 1. С. 128-135.
3. Дунаев С.П. Технологии Интернет-программирования. СПб.: БХВ-Петербург, 2001.
4. Кирсанов А.Ю. Методика экспериментального исследования динамики работы системы дистанционного управления экспериментом через информационные сети общего пользования // Электронное приборостроение. Научно-практический сборник. 2005. № 3(44). С. 42–49.
5. Назаров А.А. Исследование компьютерных сетей связи с протоколами случайного множественного доступа // Вестник Томского государственного университета. 2000. № 271. С. 72–73.
6. Balestra A. Remote Control for Galeleo Telescope and the EU // «The Three Galileos Conference". Padova, Italy, 7-10 Jan. 1997.
7. Buckman B. VI-Based Introductory Electrical Engineering Laboratory Course // International Journal of Engineering Education. 2000. Т. 16. № 3. С. 25–30.
8. Liu J. Distributed Distance Measurement for Large-Scale Networks // Computer Networks: The International Journal of Computer and Telecommunications. 2003. Т. 41. № 2. С. 177–192.
9. Ponomarev D.U. Tensor Analysis for Investigation Next Generation Networks // Proceedings of IEEE International Siberian Conference on Control and Communications, Tomsk, 20 november. 2005. С. 53–57.

10. Овчинников М.Ю. и др. Полунатурное моделирование как элемент подготовки специалистов по механике и управлению // Сборник научно-методических статей Теоретическая механика. М.: Изд-во Моск. Ун-та, 2009. С. 137–142.

11. Овчинников М.Ю. От теоретической механики до научной работы // Современные проблемы механики и ее преподавания в ВУЗах: доклады IV Всероссийского совещания-семинара заведующих кафедрами и ведущих преподавателей теоретической механики ВУЗов Российской Федерации. Новочеркасск, 21-24 сентября 2010г. : Юж.-Рос. гос. техн. ун-т. - Новочеркасск: ЮРГТУ, 2010. С. 165–166.

12. Ferreira J.M. The PEARL Digital Electronics Lab: Full Access to the Workbench via the Web // Proceedings of the 13th Annual Conference in Education - European Association for Education in Electrical and Information Engineering. , 2002.

13. Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments. Сб. тр. Междунар. науч.-практич. конф. «Образовательные, научные и инженерные приложения в среде LabVIEW и технологии National Instruments». М.: , 2006.

14. Fjeldly T.A. Lab-on-Web - A Comprehensive Electronic Device Laboratory on a Chip Assembled via Internet // International Conference on Engineering Education. August 18-19. Manchester, U.K.: , 2002. С. 1–5.

15. Schwars J. Remote Lab: A Reliable Tele- Laboratory Environment // International Conference on Internet Computing, January. , 2000.

16. Schafe T. PERL: A Generic Architecture for Live Experiments in a Remote Laboratory // SCS International Conference on Simulation and Multimedia in Engineering Education, ICSEE'03, January 2003.

17. Saliah H.H. Resource Management Strategies for Remote Virtual Laboratory Experimentation // 2000 Frontier in Education Conference: Building a Century of Progress in Engineering Education", October 18-21. Kansas City, USA: , 2000.

18. Rugelj J. The Application Of Information Services In Computer Networks To Education // IEEE Instrumentation and Measurement Technology Conference. May 18-21. St. Paul, Minnesota: , 1998.

19. Определение параметров моделей лопастных двигателей ориентации. Исследование алгоритмов управления ориентацией: Руководство к лабораторной работе / Сост.: Д.С. Иванов, Д.О. Нуждин, С.С. Ткачев. Под ред. М.Ю. Овчинникова — М.: МФТИ, 2012. — 48 с.

Оглавление

ВВЕДЕНИЕ	3
1 ПОСТРОЕНИЕ ТЕЛЕКОММУНИКАЦИОННЫХ СИСТЕМ ДИСТАНЦИОННОГО УПРАВЛЕНИЯ УЧЕБНЫМ И НАУЧНЫМ ЭКСПЕРИМЕНТОМ	4
1.1 ОБЗОР СУЩЕСТВУЮЩИХ СИСТЕМ ДИСТАНЦИОННОГО УПРАВЛЕНИЯ УЧЕБНЫМИ И НАУЧНЫМИ ЭКСПЕРИМЕНТАМИ.....	4
1.1.1 AIM- Lab (<i>Automated Internet Measurement Laboratory</i>) и <i>LAB-on-Web</i>	5
1.1.2 <i>PEARL (Practical Experimentation by Accessible Remote Learning)</i>	7
1.1.3 <i>REMOT (Remote Experiment Monitoring and control)</i>	9
1.1.4 <i>uCV-Lab</i>	10
1.1.5 <i>Distant Lab 1.0</i>	10
1.2 ВЫВОД ПО РЕЗУЛЬТАТАМ ОБЗОРА	13
1.3 РЕАЛИЗАЦИЯ НЕОБХОДИМОЙ ПРОГРАММНОЙ СРЕДЫ ДЛЯ ЛАБОРАТОРИИ В МФТИ.....	13
1.3.1 Принцип работы СДУ.....	14
1.3.2 Программная модель СДУ.....	15
2 СИСТЕМА ДЛЯ ДИСТАНЦИОННОГО ПРОВЕДЕНИЯ ЛАБОРАТОРНЫХ РАБОТ	16
2.1 ОПИСАНИЕ СТЕНДА	16
2.2 ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ МАКЕТА.....	18
2.2.1 Программа на макете (серверная часть)	18
2.2.2 Пользовательская программа (клиентская часть).....	20
2.2.3 Интерфейс программы.....	20
3 ПРОВЕДЕНИЕ ЛАБОРАТОРНЫХ РАБОТ	22
3.1 ПОРЯДОК ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ.....	22
3.1.1 Общие действия при проведении локальной лабораторной работы .	22
3.1.2 Общие действия при проведении удаленной лабораторной работы .	23
3.1.3 Исследование алгоритмов управления ориентацией	23
3.2 РЕЗУЛЬТАТЫ ПРОВЕДЕНИЯ ЛАБОРАТОРНОЙ РАБОТЫ "ИССЛЕДОВАНИЕ АЛГОРИТМОВ УПРАВЛЕНИЯ ОРИЕНТАЦИЕЙ"	24
ЗАКЛЮЧЕНИЕ	28
СПИСОК ЛИТЕРАТУРЫ	29