

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
им. Н.Э.Баумана

АВТОМАТИЗИРОВАННАЯ СИСТЕМА МОНИТОРИНГА КРУПНОГАБАРИТНЫХ ЭНЕРГЕТИЧЕСКИХ КОМПЛЕКСОВ

Разработал:
Афанасьев А.В.
ИУ4-52

Научный руководитель:
доцент, к.т.н Власов А.И.

Москва
2003

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ	2
СПИСОК УСЛОВНЫХ ОБОЗНАЧЕНИЙ, СОКРАЩЕНИЙ И ТЕРМИНОВ.....	3
ВВЕДЕНИЕ	4
1. ИССЛЕДОВАНИЕ ЗАДАЧ И ОБЪЕКТА МОНИТОРИНГА И ДИАГНОСТИКИ.....	6
1.1. Задачи анализа по вибрации.	6
1.2. Объекты и назначение анализа по вибрации.....	6
2. АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС МОНИТОРИНГА КРУПНОГАБАРИТНЫХ ЭНЕРГЕТИЧЕСКИХ КОМПЛЕКСОВ.....	11
2.1. Постановка задачи	11
2.2. Структурно-функциональное построение комплекса	12
2.3. Анализ методов мониторинга и диагностики крупногабаритных энергетических комплексов.	14
2.4. Программное обеспечение	19
2.4.1. Ядро.....	19
2.4.2 Модули расширения	21
2.5 Лингвистическое обеспечение	26
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	31

СПИСОК УСЛОВНЫХ ОБОЗНАЧЕНИЙ, СОКРАЩЕНИЙ И ТЕРМИНОВ.

АПК	аппаратно-программный комплекс;
АСУ	автоматическая система управления;
АЧХ	амплитудно-частотная характеристика;
ИТО	исследуемый технический объект;
КЗ	короткое замыкание;
МКР	метод конечных разностей;
МКЭ	метод конечных элементов;
МГЭ	метод граничных элементов;
ОС	операционная система;
ПО	программное обеспечение;
САПР	система автоматического проектирования;
ЭВМ	электронная вычислительная машина;
ЭП	Электромеханические преобразователи;
CAD	Computer Aided Design – автоматизированное проектирование;
DDE	Dynamic Data Exchange –
DSP	Digital Signal Processing – цифровой сигнальный процессор (ЦСП);
DXF	Data eXchange Format – формат передачи данных;;
FEA	Finite Element Analyse – конечноэлементный анализ;
OLE	Object Linking and Embedding;
OpenGL	Open Graphic Library – графическая библиотека, графический стандарт;
UDP	User Datagram Protocol – транспортный протокол без установления соединения;

Типы представления данных.

int тип данных (диапазон -2147483648...2147483647, размер 4 байта);
long тип данных (диапазон -2147483648...2147483647, размер 4 байта);
float тип данных (диапазон -3.4E38... 3.4E38, размер 4 байта)
double тип данных (диапазон -1.7E308.....7E308, размер 8 байт);

ВВЕДЕНИЕ

В настоящее время затраты на обслуживание и ремонт являются одним из важнейших эксплуатационных показателей любой технической системы. Их минимизация в тех случаях, когда система является ремонтпригодной, практически невозможна без эффективного контроля состояния системы. В современных средствах контроля и диагностики, основным видом анализируемых процессов становится вибрация, активно вытесняя многие другие процессы, в том числе и тепловые. Эффективность, относительная дешевизна и простота установки комплексов виброанализа – одни из главных причин такого лидирования вибрации в качестве объекта контроля. Возможность начать диагностику по вибрации в любое время, в том числе и через несколько лет эксплуатации оборудования, когда затраты на профилактические работы и ремонт превысят экономически оправданную величину, является экономическим фактором бурного развития комплексов виброанализа у нас в стране и в мире в целом.

Возможности вибрационной диагностики не следует отождествлять с возможностями аварийной защиты оборудования по вибрации или вибрационного контроля. Средства аварийной защиты – простейшие измерители мощности вибрации, обнаруживающие предаварийное состояние объекта и отключающие его до или на начальном этапе аварии. Средства вибрационного контроля (мониторинга) – это измерители мощности составляющих вибрации, обнаруживающие ее изменения в процессе эксплуатации оборудования. Эти средства не предназначены для обнаружения зарождающихся дефектов и поэтому в ряде случаев регистрируют изменение состояния лишь после появления цепочки развитых дефектов, последний из которых наиболее сильно влияет на вибрацию. Такие цепочки дефектов появляются незадолго до аварии, поэтому высокую эффективность имеют только системы мониторинга, измеряющие вибрацию практически непрерывно.

В последнее десятилетие были разработаны несколько эффективных методов обнаружения основных дефектов машин и оборудования по вибрации на этапе их зарождения. Естественно, что они основаны, в основном, на анализе высокочастотной вибрации, для возбуждения которой не нужны большие колебательные силы, но и проявляется она только в месте их действия, быстро затухая при распространении. Такие методы стали использоваться диагностами многих стран для перехода от вибрационного мониторинга к глубокой диагностике.

Параллельно развивались методы автоматизации алгоритмов диагностики, что позволило ряду производителей диагностических систем заменить программным обеспечением высококлассных экспертов при решении типовых диагностических задач. Доля таких задач очень высока и превышает девяносто процентов от всех задач, решаемых путем анализа сигналов вибрации. Первые автоматические системы вибрационной диагностики были разработаны в 1991-1992 годах и постоянно совершенствовались.

В 1998 году появилось новое поколение систем автоматической диагностики машин по вибрации, разработанное совместно специалистами России и США, и объединившее лучшие из свойств как систем мониторинга, так и систем диагностики. Краткий анализ возможностей таких систем и особенностей их построения приводится ниже.

Как показывает практика, наиболее экономически эффективным является способ вывода техники из технологического цикла и передачу ее в ремонт по результатам диагностики и прогноза состояния. Целью данной работы является создание аппаратно-программного комплекса (АПК) вибрационной диагностики и мониторинга различных конструктивных элементов крупногабаритных энергетических комплексов с возможностью работы как в полуавтоматическом режиме – вибродиагностика конструкций по запросу оператора, так и в полностью автоматическом – вибромониторинг конструкций и автоматическое исключение их из технологического цикла, если экспертная система комплекса дала заключение, что появились определенные дефекты в элементах (микротрещины, асимметрия рабочего цикла ротора и проч.). Таким образом, с помощью разрабатываемого АПК появляется возможность организовать технологический цикл таким образом, чтобы простои из-за внезапного отказа различных частей оборудования были минимизированы или вообще исключены.

1. ИССЛЕДОВАНИЕ ЗАДАЧ И ОБЪЕКТА МОНИТОРИНГА И ДИАГНОСТИКИ

1.1. Задачи анализа по вибрации.

Мониторинг машин и оборудования является лишь одной из ряда технических задач, в которых для диагностики используются измерения и анализ вибрации машины и оборудования. Такого рода задачи можно разделить на семь основных групп, в том числе:

- вибрационный мониторинг машин и оборудования;
- вибрационная диагностика;
- балансировка роторов по вибрации;
- обнаружение источников вибрации (шума);
- обнаружение источников акустической эмиссии;
- вибрационный модальный анализ;
- ультразвуковая дефектоскопия.

В каждой группе задач имеются свои ограничения как на объекты диагностики, так и на основные свойства вибрации, используемые для достижения оптимальных результатов. Для их описания вибрацию следует разделить, во-первых, по частотному признаку, а, во-вторых, по способу ее формирования.

По частоте, как правило, она делится на четыре области - низкочастотную, среднечастотную, высокочастотную и ультразвуковую.

По способу формирования - на вибрацию естественного происхождения (в машинах и т.п.) и искусственного, возбуждаемую специальным источником - вибратором.

Вибрация естественного происхождения используется для решения первых пяти групп задач, искусственного - для последних двух групп.

1.2. Объекты и назначение анализа по вибрации.

Вибрационный мониторинг.

Его объектами являются, прежде всего, машины и оборудование – источники вибрации. Отличительной особенностью таких объектов можно считать наличие в них колебательных сил, возникающих, например, при движении отдельных узлов или потоков жидкости (газа), при действии переменных электромагнитных полей. Только в редких случаях объектами мониторинга может

быть оборудование, не являющееся источником колебательных сил и вибрации, но по которому распространяется вибрация от другого источника.

Назначением вибрационного мониторинга является обнаружение изменений вибрационного состояния контролируемого объекта в процессе эксплуатации, причинами которых во многих случаях являются дефекты.

Мониторинг машин и оборудования проводится прежде всего по низкочастотной и среднечастотной вибрации, которая хорошо распространяется от места формирования до точек ее контроля. Число таких точек может быть сведено к минимуму, до одной – двух на каждый объект мониторинга, имеющий общий корпус, а измерения вибрации могут проводиться без изменения режима работы объекта. В системе мониторинга, если она не включена в систему быстродействующей аварийной защиты, может использоваться аппаратура с одним каналом измерения вибрации, к которому последовательно подключаются все используемые датчики вибрации. Эти меры позволяют существенно снизить стоимость системы мониторинга без снижения достоверности получаемых результатов.

Вибрационная диагностика.

Ее объектами являются те же машины и оборудование, которые охватываются системами вибрационного мониторинга. Вибрационная диагностика чаще всего используется или для выходного контроля качества изготовления (ремонта) и сборки машин, или для их предремонтной дефектации, или для обнаружения дефектов и слежения за их развитием в процессе эксплуатации. Для каждого из перечисленных случаев могут использоваться разные методы диагностики. В последнем случае диагностические измерения могут проводиться без смены режима работы объекта диагностики.

В отличие от мониторинга назначением вибрационной диагностики в процессе эксплуатации оборудования является обнаружение изменений и прогноз развития не вибрационного, а технического состояния, причем каждого из его элементов, для которого существует реальная вероятность отказа в период между ремонтами. Для этого измеряется не только низкочастотная и среднечастотная, но и высокочастотная вибрация, а также используются более сложные, чем при мониторинге, методы анализа вибрации, позволяющие получать полный объем диагностической информации. Вибрация измеряется на каждом диагностируемом узле или, по крайней мере, в точках перехода высокочастотной вибрации от

диагностируемого к другим узлам объекта, доступным для измерения вибрации. Используемая аппаратура также может иметь только один канал измерения и анализа вибрации.

Балансировка роторов.

В процессе эксплуатации машин их вибрация на частоте вращения ротора может расти и для ее снижения приходится балансировать ротор непосредственно на месте эксплуатации каждой машины. Объектами балансировки на месте эксплуатации, как правило, являются машины, в которых есть доступ к плоскостям балансировки, т.е. к местам, в которых на вращающихся частях можно закреплять балансировочные массы.

Основным назначением балансировки является уравнивание ротора и, тем самым, снижение низкочастотной вибрации машины. Но вибрация на частоте вращения ротора далеко не всегда определяется центробежными силами, поэтому добиться значительных и стабильных результатов в большинстве случаев удастся лишь с помощью средств, определяющих причины роста вибрации в каждом случае, т.е. решающих и диагностические задачи. Балансировка на месте эксплуатации машины проводится по вибрации и с помощью тех же приборов, что и вибрационная диагностика.

Средства измерения вибрации для балансировки ротора должны иметь, как минимум два канала измерения, один из которых определяет амплитуду вибрации на частоте вращения, а другой (канал с датчиком оборотов) - ее фазу относительно выбранной метки на роторе. При поиске дефектов, ограничивающих возможности балансировки, дополнительно к низкочастотной может измеряться и высокочастотная вибрация опор ротора или корпуса машины. Вибрация измеряется в типовых режимах работы машины, на которых необходимо производить балансировку.

Обнаружение источников вибрации (шума).

Подобная задача на производствах возникает прежде всего при поиске утечек, особенно в протяженных трубопроводах, недоступных для визуального осмотра. Решается она, как правило, во время проведения различного рода испытаний с повышением давления в трубопроводах или после обнаружения изменения состояния объекта контроля средствами мониторинга.

В том случае, когда необходимо найти источник повышенной вибрации среди большого количества объектов, недоступных для установки датчиков вибрации, обнаружение проводится по шуму, излучаемому этим источником. Для выполнения такой работы, назначением которой является и обнаружение координат источника, используется измерительная аппаратура, по крайней мере с двумя каналами измерения либо шума в газообразной или жидкой среде, либо высокочастотной вибрации, например, на стенках трубопроводов.

Обнаружение источников акустической эмиссии.

Обнаружение утечек в сосудах высокого давления и трубопроводах происходит тогда, когда дефект уже настолько значителен, что требует срочного вмешательства. На ранней стадии развития дефекты, приводящие к утечкам, обнаруживаются по акустической эмиссии дефектных участков нагруженных оболочек и конструкций, в результате которой во время формирования микротрещин возникает ультразвуковая вибрация. Она хорошо распространяется по однородному материалу, не имеющему сварных и других видов соединений, и регистрируется высокочастотными датчиками вибрации даже на значительном расстоянии от источника эмиссии.

По акустической эмиссии обнаруживаются зарождающиеся дефекты не только сосудов, находящихся под давлением, но и различных металлических строительных конструкций, мостов, кранов и т. д. Для повышения чувствительности средств обнаружения и для определения места возникновения дефектов используются многоканальные средства одновременного измерения высокочастотной вибрации, отличающиеся повышенной сложностью. Методы обнаружения акустической эмиссии эффективны тогда, когда в объекте контроля отсутствует высокочастотная вибрация, возбуждаемая другими источниками, например, потоками газа или жидкости, как это имеет место в трубопроводах под давлением. В последних случаях приходится принимать специальные меры для разделения источников вибрации, которые далеко не всегда оказываются достаточными.

Вибрационный модальный анализ.

Его объектами являются не только машины, оборудование и их узлы, но и составные части различных конструкций, сооружений и т. д. Используется этот вид

анализа для определения таких механических свойств объектов, как частоты резонансов, формы колебаний и других.

Назначением модального анализа может быть как отработка новых конструкций машин и оборудования, так и выходной контроль серийно выпускаемых изделий, и прежде всего по значениям резонансных частот либо объекта в целом, либо отдельных его узлов. Для решения диагностических задач в процессе эксплуатации машин и оборудования системы модального анализа используются нечасто, и в первую очередь из-за сложности собственно систем и их обслуживания. Составными частями таких систем являются источник вибрации и несколько (не менее двух) каналов для ее измерения и анализа. В качестве источника вибрации непрерывного действия используются специальные машины - вибраторы, в качестве источников импульсной вибрации - молотки с встроенными устройствами измерения ударных импульсов.

Ультразвуковая дефектоскопия.

Объектами дефектоскопии являются отдельные элементы машин, оборудования, конструкций и сооружений, как правило, находящихся в стадии изготовления или восстановления. Средства дефектоскопии, использующие внешние источники ультразвуковой вибрации, по своей структуре и назначению похожи на средства модального анализа “в миниатюре”, но в них есть и другие отличительные черты, кроме области частот измеряемой вибрации.

Так, дефектоскопия использует волновые свойства вибрации, в частности ее отражение от различных неоднородностей и потери при распространении. Это позволяет обнаружить и локализовать дефектные участки внутри деталей или их заготовок, что и является основным назначением средств ультразвуковой дефектоскопии. Подобные средства, как и средства модального анализа, весьма редко используются для диагностики машин в процессе эксплуатации. Одной из причин этого является высокая эффективность методов и средств дефектоскопии, использующих другие виды излучений, например электромагнитное, рентгеновское и т. д.

2. АППАРАТНО-ПРОГРАММНЫЙ КОМПЛЕКС МОНИТОРИНГА КРУПНОГАБАРИТНЫХ ЭНЕРГЕТИЧЕСКИХ КОМПЛЕКСОВ

2.1. Постановка задачи

На мировом рынке аппаратного и аппаратно-программного обеспечения существует довольно много систем, осуществляющих измерение виброакустических и вибромеханических полей. Значительно меньше продуктов, ориентирующихся на комплексном подходе к созданию автоматизированных систем, т.е. систем мониторинга и виброанализа конструктивных элементов машин в автоматическом режиме для обнаружения дефектов на ранних стадиях их зарождения. В таблице 2.1. приведены сравнительные характеристики существующих на рынке виброизмерительных систем.

Таблица 1. Сравнительные характеристики виброизмерительных систем

Название	Производитель	Диапазон измерения, Гц	Виброскорости, мм/с	Тип	Цена, \$
Vibro Vision	ООО «Вибро-Центр»	10 000	100	Аналоговый	560
Лазерный виброметр LV-2	ООО «ЛАЗЕРНАЯ ТЕХНИКА»		200	Аналоговый	3000
Викинг-2	ООО «Аурис»	0.1 - 23000		Программно-аппаратный	10000
Атлант-8	ООО «Вибро-Центр»	5 - 5000		Программно-аппаратный	7200
Корсар+	ООО «Вибро-Центр»	10 - 1000	0,3 - 100	Программно-аппаратный	1200
Диана-С	ООО «Вибро-Центр»	1000		Программно-аппаратный	2000

Целью работы является создание аппаратно-программного комплекса, интегрирующего в себе виброизмерительную подсистему, подсистему предварительной обработки полученного ряда дискретных измерений, алгоритмическое обеспечение различных математических моделей анализа измерений, а также экспертную систему с возможностью передачи полученного решения как оператору через визуализацию на контрольном дисплее, так и передачи этого решения на подсистему управления технологическим процессом. Анализ существующих систем подобного рода, а также растущие потребности в улучшении контроля производственных циклов, показывает обоснованность и экономическую целесообразность разработки комплекса такого класса.

Настоящий АПК ориентирован на постоянный мониторинг состояния крупногабаритных энергетических установок путем анализа вибрационного спектра, генерируемого вращающимися механическими элементами конструкции. Основной задачей АПК является постоянный съем, анализ и обработка данных с вибродатчиков, а также визуализация результатов анализа на дисплее.

В рамках работы решаются следующие задачи:

- создание программной базы комплекса мониторинга с поддержкой модульной структуры для возможного наращивания комплекса;
- разработка множества модулей для съема данных датчиков (с помощью АЦП), для генерации полей (с помощью ЦАП), а также модулей визуализации (визуализация экспертной оценки, визуализация спектра и проч.)
- разработка модулей алгоритмического обеспечения математической модели, выбранной для анализа вибро-процессов, объекта мониторинга.

2.2. Структурно-функциональное построение комплекса

Для повышения эффективности разрабатываемого программного обеспечения комплекса, а также уменьшения вероятности возникновения возможных сбоев и ошибок, а также обеспечения легкости их устранения во время работы комплекса в сложном технологическом цикле, структура комплекса является модульной. Таким образом повышается проблемостойкость комплекса при выполнении намеченных задач.

Поскольку разрабатываемое программное обеспечение должно обеспечивать адекватную работу АПК в различных режимах (полуавтоматическая диагностика, мониторинг), необходима возможность неодинакового наполнения комплекса различными по функциональности компонентами (модулями). В различных задачах, может быть общим, например, сбор данных (т.е. измерения производятся аналогичным образом, используя одинаковые аналого-цифровые преобразователи), а отличаться обработкой данных и экспертной оценкой. В этом случае нет необходимости каждый раз разрабатывать заново программное

обеспечение целиком, а сконцентрировать усилия только на самом необходимом, т.е. на реализации алгоритмического обеспечения выбранной математической модели обработки дискретных векторов данных и экспертной оценке. На рисунке 2.2.1. представлена предложенная схема модели построения модульности комплекса.

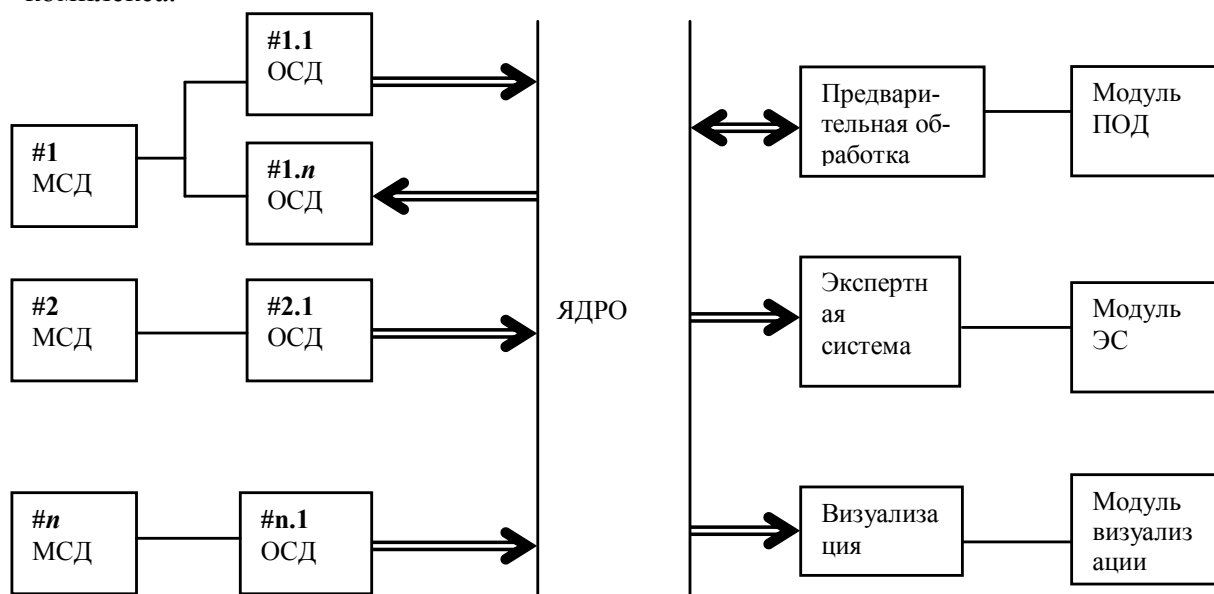


Рис.2.2.1. Модульная модель ПО АПК

Пояснение:

МСД – модуль сбора данных;

ОСД – объект сбора данных;

ЯДРО – «сердце» АПК, осуществляющее взаимодействие всех частей комплекса.

Под модулями на схеме понимается отдельно разработанный модуль программы подключаемый определенным образом к комплексу. Все ОСД, «Предварительная обработка данных», «Экспертная система», «Визуализация» являются конкретными объектами в памяти комплекса, настроенные на работу с определенным устройством в определенном режиме. Например, если модуль драйвера АЦП/ЦАП «Lcard 7xx» предоставляет программе интерфейс для работы с этой платой, то объект представляет собой сущность, настроенную на работу с выбранной платой АЦП/ЦАП установленной в системе в режиме, например, сбора данных. Тонкими линиями показана своего рода наследственность объектов от модулей комплекса при данном режиме работы с комплексом. Жирными линиями со стрелками показано направление потока данных от объектов и/или к объектам. Объекты могут получать данные и/или их отдавать ядру. Ядро комплекса устанавливает взаимодействия потоков данных между различными объектами.

Выбор данной схемы построения модульности комплекса позволяет практически неограниченно расширять возможности комплекса по измерению дописывая отдельные модули и библиотеки модулей и встраивая их в комплекс.

В качестве основных компонентов можно выделить ядро, которое будет управлять всеми остальными процессами, препроцессор, ответственный за ввод информации и представлении ее в стандартном унифицированном формате; процессор, предназначенный непосредственно для обработки данных и вывода результатов в стандартном формате, и постпроцессор, предназначенный для преобразования стандартных результатов в необходимый формат.

2.3. Анализ методов мониторинга и диагностики крупногабаритных энергетических комплексов.

Диагностические задачи могут решаться различными методами, часто дублирующими друг друга. В данном разделе работы приводятся основные и наиболее эффективные методы анализа вибрации для постановки диагноза относительно дефектности оборудования.

Основной частью в энергетических комплексах, а также потенциально наиболее дефектоопасной, являются электрические машины, которые представляют собой комплекс вращающихся конструктивных элементов. Для таких объектов в настоящее время диагностические задачи решаются наиболее полно, а также разработаны методики для мониторинга и диагностики.

Основной метод вибрационного мониторинга - наблюдение за изменением энергетических параметров вибрации машины и, прежде всего, мощности (уровня) отдельных компонент вибрации. Особенности любого подхода к решению задач мониторинга определяются тремя главными факторами. Первый - выбор точек измерения вибрации, второй - количества разделяемых компонент и третий - интервалов между измерениями.

В настоящее время активно развиваются два основных направления вибрационного мониторинга машин и оборудования. В рамках одного из них решаются задачи минимизации числа точек контроля вибрации и интервалов между измерениями в первую очередь за счет использования стационарных систем мониторинга непрерывного действия. В рамках другого максимизируются интервалы между измерениями за счет контроля вибрации во многих точках, в том числе на каждом узле оборудования, являющемся источником вибрации. В

результате роста этих интервалов появляется возможность использования переносной аппаратуры для измерения и анализа вибрации.

И в том и в другом случае эффективность мониторинга зависит от количества компонент вибрации, доступных для измерения. Чтобы увеличить их число в машинах с вращающимися узлами производится узкополосный спектральный анализ вибрации, наиболее эффективный для анализа периодических компонент сигнала. В качестве примера на рис. 2.3.1 приведены спектры вибрации машины (редуктора) без дефектов и после появления дефектов шестерен, сопровождающихся ростом уровня (мощности) многих составляющих спектра вибрации.

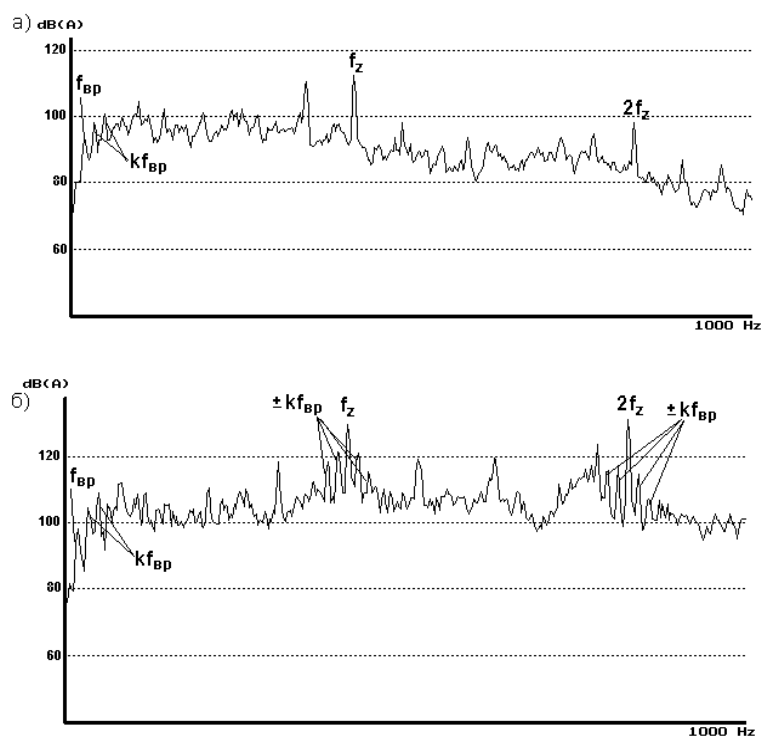


Рис.2.3.1. Спектры вибрации подшипникового узла редуктора.
а) - бездефектный редуктор; б) - редуктор с дефектом шестерни. f_{bp} - частота вращения оси редуктора; f_z - зубцовая частота.

Мониторинг вибрационного состояния машины в этом случае заключается в сравнении уровня отдельных составляющих с указанными на рисунке 2.3.2. порогами и анализ изменений уровня отдельных составляющих за время эксплуатации.

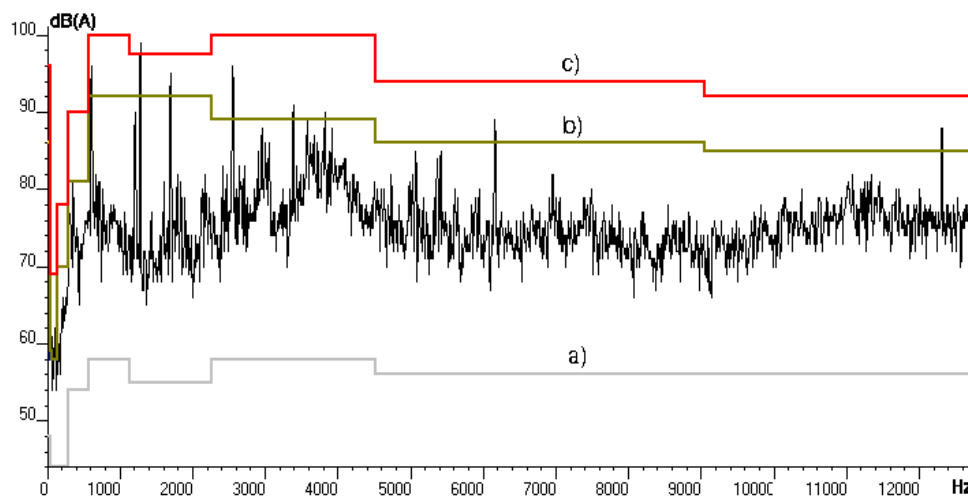


Рис.2.3.2. Мониторинг вибрационного состояния машины.

а) - порог слабого сигнала вибрации; б) - порог среднего сигнала вибрации; в) - порог сильного сигнала вибрации.

В практической диагностике машин по вибрации существует два основных подхода к решению диагностических задач.

В первом случае диагностика выполняется только после обнаружения изменений вибрационного состояния машины средствами мониторинга и ее задачей является интерпретация этих изменений. Однако мониторинг проводится, как правило, по низкочастотной и среднечастотной вибрации, реагирующей на появление, в основном, только развитых дефектов. Именно они приводят к заметным изменениям энергетических параметров вибрации, превышающим их естественные флуктуации при смене режимов работы машины.

Второй подход заключается в использовании тех методов и средств диагностики, которые обнаруживают основные виды дефектов на этапе их зарождения еще до того, как произойдут существенные энергетические изменения в сигнале вибрации машины в целом. Обнаружение дефектов на этапе зарождения дает возможность наблюдать за их развитием и своевременно планировать работы по ремонту и обслуживанию машины. Такой подход достаточно часто называют мониторингом дефектов, и он может осуществляться в течение нескольких лет, пока совокупность имеющихся дефектов как по глубине, так и по количеству, не приведет к ситуации, близкой к аварийноопасной. Естественно, что такой подход возможен лишь в том случае, когда все потенциально опасные дефекты могут быть обнаружены и идентифицированы на ранней стадии их развития. Многие годы такой возможности просто не существовало. Однако в результате интенсивного развития методов и технических средств диагностики, и прежде всего по сигналу

вибрации, в настоящее время системы мониторинга дефектов становятся реальностью и получают все более широкое распространение.

Для обнаружения зарождающихся дефектов во вращающихся узлах используются естественные диагностические признаки, определяемые путем спектрального анализа самого сигнала вибрации или колебаний мощности его компонент (спектральный анализ огибающей).

Так, основным признаком дефектов в подшипниках является изменение свойств сил трения и возбуждаемой ими высокочастотной вибрации. Эти изменения, заключающиеся либо в появлении микроударов, либо в периодическом изменении коэффициента трения при контакте дефектных участков поверхностей трения, легко обнаруживаются при спектральном анализе огибающей высокочастотных составляющих вибрации корпуса подшипникового узла, возбуждаемых силами трения, см. рис. 2.3.3.

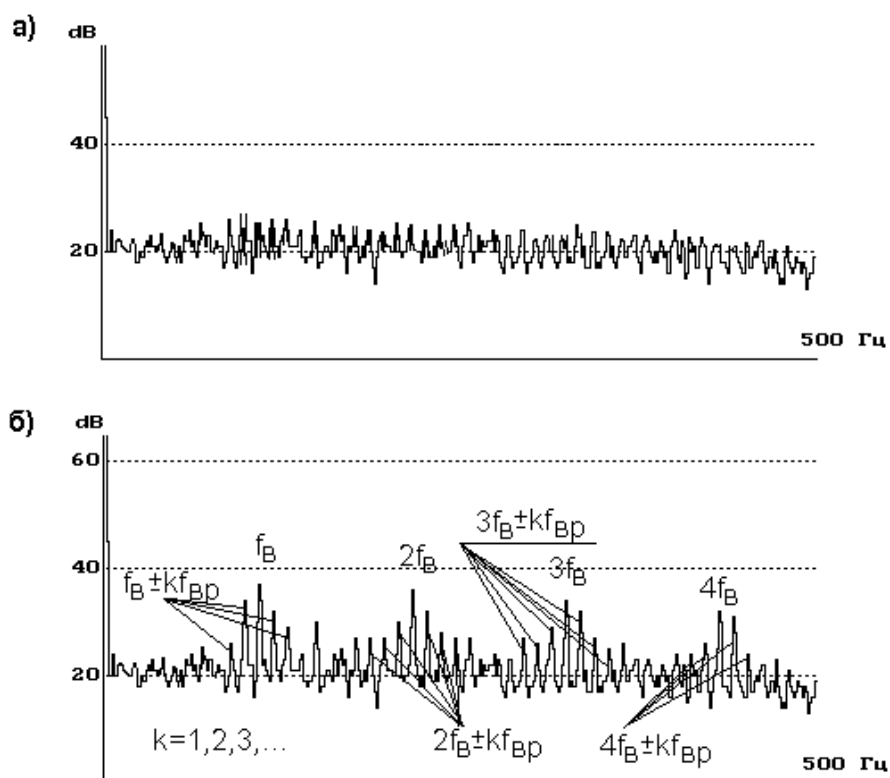


Рис.2.3.3. Спектр огибающей вибрации подшипника без дефектов (а) и подшипника с раковинной на внутреннем кольце (б).

f_{Bp} - частота вращения внутреннего кольца подшипника; f_B - частота перекатывания тел качения по внутреннему кольцу подшипника.

При обнаружении дефектов в узлах механических передач, в частности, зубчатых, ременных, с упругими муфтами и других, основное внимание уделяется таким признакам дефектов, как появление динамических нагрузок на элементы передачи, в частности, подшипники. Эти нагрузки приводят, во-первых, к росту

определенных компонент вибрации, а, во-вторых, к флуктуациям сил трения в подшипниках, и, как следствие, к колебаниям мощности высокочастотной вибрации. Таким образом, по совокупности результатов измерения спектров вибрации и ее огибающей можно успешно обнаруживать и идентифицировать зарождающиеся дефекты механических передач.

Основные диагностические признаки дефектов потокосоздающих узлов роторного типа, в частности, рабочих колес центробежных насосов и компрессоров, а также рабочих колес турбин, похожи на признаки дефектов механических передач и их подшипников. Они связаны с изменением свойств турбулентного потока в зоне направляющих аппаратов и рабочих колес, пульсации давления в котором возбуждают высокочастотную вибрацию корпуса, с одной стороны, и создают динамические нагрузки на подшипники, с другой. Поэтому обнаруживаются дефекты потокосоздающих узлов при совместном анализе спектров вибрации машины и огибающей высокочастотной вибрации корпуса и подшипниковых узлов.

Силы трения в подшипниках и пульсации давления в потоках жидкости или газа существенным образом изменяют свои свойства при появлении многих, но не всех дефектов в узлах роторных машин. Так ряд дефектов в электрических машинах изменяет параметры электромагнитного поля в зазоре и на ротор машины начинают действовать дополнительные электромагнитные силы и пульсирующие моменты. Поскольку в бездефектных машинах пульсирующих моментов нет, именно они и возбуждаемая ими вибрация наиболее эффективно используются для обнаружения зарождающихся дефектов.

Спектральный анализ вибрации электрических машин на низких и средних частотах, измеряемой в разных направлениях, дает возможность идентифицировать практически все дефекты электромагнитной системы электрических машин, кроме старения изоляции, так как до момента пробоя изоляции электромагнитное поле и вибрация машины не изменяются.

Еще одно свойство вибрации используется для обнаружения дефектов роторов, особенно высокооборотных машин, которое связано с изменением величины и структуры центробежных сил. В зависимости от вида дефекта растет низкочастотная вибрация и определенным образом изменяется ее спектральный состав, что и используется для его идентификации.

Таким образом, по сигналу вибрации могут быть обнаружены практически все виды зарождающихся дефектов во вращающемся оборудовании без привлечения для диагностики других видов физических процессов. А это означает,

что для перехода на обслуживание и ремонт этого оборудования по фактическому состоянию могут использоваться переносные системы вибрационной диагностики, не требующие встраивания измерительных датчиков в оборудование и его узлы.

2.4. Программное обеспечение

Реализация предложенной модели структурно-функционального построения комплекса на модульном принципе сделана, используя COM технологию (Component Object Model) в реализации компании Microsoft (Win32 API). Как уже было отмечено выше, реализация всех возможных модулей АЦП, ЦАП, алгоритмов обработки данных, а также различных графических компонентов на основе модульного принципа (в нашем случае, используя COM технологию), позволяет разрабатывать дополнительные библиотеки компонентов комплекса на любом языке программирования, поддерживающим эту технологию, что, в свою очередь, дает возможность комплексу как наращивать свой потенциал, так и специализироваться на конкретную задачу. Подключение модулей к комплексу, а также взаимодействие модулей друг с другом обеспечивается поддержкой модулем расширения комплекса (или просто модулем) определенных т.н. COM интерфейсов, т.е. набора различных функций работы, экспортируемых модулем.

Согласно схеме модульного построения, основной и неотъемлемой частью комплекса, обеспечивающей своевременную загрузку кода COM компонентов в оперативную память, создание COM Объектов, организацию взаимодействия потоков, настройку режимов работы COM Объектов является ядро комплекса.

Каждому созданному COM Объекту присваивается внутренний уникальный идентификатор (IUID – Internal Unique ID), по которому объект опознается в системе. Ответственность за правильную раздачу IUID ложится на ядро. Для визуального различения пользователем различных созданных COM Объектов в системе также присутствует псевдо-уникальное сочетание букв и цифр, также называемое «кратким названием объекта».

2.4.1. Ядро

Для обеспечения доступа к установленным в системе COM компонентам организовано иерархическое древовидное меню (далее главное меню), в котором компоненты с общим принципом действия объединены в единые подпункты (Main Menu на рис.3). Организация главного меню, в том числе изменение порядка и структуры расположения, добавление и удаление компонентов, ложится на

пользователя. В связи с чем, ему предоставляется комплексом интерфейс конструктора комплекса, переключение в который осуществляется через меню Tools->Constructor. Физическое сохранение содержания меню осуществляется в специализированном файле menu.dat, который может быть расположен как в рабочей директории, так и в другой указанной директории. Этот файл состоит из последовательных записей, содержащих идентификационный номер опции, номер опции от которой происходит наследование (раскрывающийся список) название, краткое название, имя COM компонента в системе, а также другая служебная информация. При запуске программы осуществляется считывание содержимого файла меню в память. В случае, если комплексу не удалось найти файл меню, он предлагает создать этот файл в текущем рабочем каталоге. Создание древовидной структуры в окне “Main Menu” осуществляется путем динамического массива, таким образом временная сложность создания меню является $O(n)$, что может быть решающим фактором в случае большого числа установленных модулей в системе. Такой выбор построения меню обеспечивает любую вложенность меню, что облегчает группировку модулей по принципу действия и/или по каким-нибудь другим параметрам.

Для работы с предоставляемыми COM компонентом функциями необходимо создать экземпляр этого COM компонента – COM Объект (список кратких названий созданных объектов представлены в меню I/O List и Driver List на рис.3), используя главное меню. Для этого надо либо два раза щелкнуть мышкой на выбранном модуле, либо нажать “Enter”, либо перетащить выбранный модуль в область I/O List или Driver List. При этом вновь созданному объекту будет назначено короткое имя с добавлением номера, для обеспечения унификации этого имени в системе, для визуального отличия при настройке потоков данных. Это имя в последствии может быть изменено пользователем. Комплекс разрешит переименование только в случае, если в системе нет объектов с таким же именем.

Для настройки режимов работы объекта, а также настройки потоков данных комплексом используется COM интерфейс IOption, который должен быть реализован всеми компонентами АПК. Данный интерфейс представляет собой набор 3-х функций:

```
STDMETHOD( SetOption(int, const COption&) ) PURE;  
STDMETHOD( SetRefreshFunc( func_type func, void *object ) ) PURE;  
STDMETHOD( SetValid(bool isvalid) ) PURE;
```

, которые позволяют комплексу получить количество опций, а также получить данные по каждой опции и передать объекту новые настройки каждой опции. В случае если объект хочет иметь возможность интерактивного обновления данных в области Properties программы (см. рис.3), объект должен поддерживать интерфейс IOptionReverse, состоящий из следующих функций:

Первая из этих функций устанавливает функцию для вызова из объекта в случае необходимости обновления области Properties. Вторая устанавливает активна ли область Properties для данного объекта (в случае если область Properties в данный момент настроена для другого объекта, то нет необходимости там что либо обновлять).

В случае если COM компонент поддерживает графический интерфейс (IGraphics), то при создании объекта для него создается отдельное окно (MDI Child) и настраиваются соответствующие указатели для обеспечения прямой и обратной связи между ядром и созданным COM объектом. Реализация всех графических обработчиков ложится на COM компонент.

Существует возможность сохранения и восстановления текущего состояния настройки всех созданных объектов. Сохраняется также текущие состояния настройки каждого модуля в отдельности (т.н. настройки по умолчанию).

При возникновении исключительной ситуации во время работы, в т.ч. нарушение целостности данных, попытки доступа по неверным адресам памяти и проч., комплекс аварийно завершает свое выполнение с выводом диагностирующего сообщения на экран.

2.4.2 Модули расширения

Как уже было отмечено, комплекс построен на модульном принципе, поэтому основным для функциональности комплекса является разработка набора компонентов комплекса, также называемых модулями расширения или просто COM компонентами. Модули расширения комплекса могут быть “пассивными” и “активными”. “Пассивные” модули начинают свою работу только при вызове какой-либо процедуры одного из поддерживаемых интерфейсов, т.е. для них в системе не создается отдельного потока (thread) задач. К “пассивному” модулю можно отнести модуль визуализации, который при получении вектора данных визуализирует их после чего передает управление процедуре его вызвавшей. С другой стороны для “активного” модуля при вызове функции Start() соответствующего интерфейса создается отдельный поток, в котором в цикле

производятся определенные действия. При вызове функции Stop() соответствующего интерфейса этот поток выполнения завершается. К “активному” модулю относится например модуль разделения каналов данных, который в цикле запрашивает вектор данных, после чего в соответствии со своим алгоритмом разбивает этот вектор на определенное число каналов и передает вновь созданные векторы данных всем модулям, на которые настроен этот модуль (используя интерфейс IExchanger). Для унификации в операционной системе каждому модулю должен быть присвоен уникальный номер CLSID (уникальный CLSID может быть сгенерен с помощью утилиты GUIDGEN.EXE поставляемой вместе с Microsoft Visual Studio 6.0) и дано символьное имя, по которому также можно опознать модуль в операционной системе. Для упрощения данных операций был разработан шаблон модуля, который имеет возможность саморегистрироваться в системе, а разработчику лишь необходимо в заголовочном файле main.h изменить значения констант CLSID_Module, MODULE_DESCR, MODULE_NAME, MODULE_VERSION. Например, необходимо отредактировать следующим образом:

```
// {0332176E-13A0-41a0-8E2B-690FEF53BC5B}
DEFINE_GUID( CLSID_Module,
0x332176e, 0x13a0, 0x41a0, 0x8e, 0x2b, 0x69, 0xf, 0xef, 0x53, 0xbc, 0x5b);

#define MODULE_DESCR      "APIK preprocessing component"
#define MODULE_NAME      "Apik.Driver.Div10"
#define MODULE_VERSION    "1"
```

Для реализации различных по функциональности и принципу действия компонентов используется следующий набор COM интерфейсов:

- IUnknown - базовый интерфейс COM компонента, который необходимо реализовывать всем модулям независимо от задач;

- IOption - интерфейс, обеспечивающий базовую работу с настройкой объекта на необходимые режимы из ядра комплекса через интерфейс пользователя:

```
STDMETHOD( NumOptions(int*) )      PURE;
STDMETHOD( GetOption(int, COption*) ) PURE;
STDMETHOD( SetOption(int, const COption&) ) PURE;
```

- IOptionReverse - интерфейс дополняющий IOption специфическими функциями обратной связи ядро-модуль:

```
STDMETHOD( SetRefreshFunc( func_type func, void *object ) ) PURE;
```

```
STDMETHOD( SetValid(bool isValid) ) PURE;
```

- **IGetData** - интерфейс сбора данных:

```
STDMETHOD( Start() ) PURE;
```

```
STDMETHOD( Stop() ) PURE;
```

```
STDMETHOD( GetData(CData*, int, int*) ) PURE;
```

- **IPutData** - интерфейс посылки данных:

```
STDMETHOD( Start() ) PURE;
```

```
STDMETHOD( Stop() ) PURE;
```

```
STDMETHOD( PutData(CData*, int) ) PURE;
```

В приведенных двух интерфейсах в качестве параметра для функций **GetData** и **PutData** используется указатель на структуру **CData**, представляющую собой:

```
struct CData  
{  
    float myX;  
    float myY;  
};
```

Таким образом, вектор передаваемых данных представляет собой сочетание координат *x* и *y*. Такой вектор является наиболее универсальной единицей данных, поскольку содержит в себе все необходимые для обработки и/или визуализации параметры. В случае, если в векторе содержатся данные о частотной характеристике сигнала, то координата *X* будет содержать частоту, а координата *Y* – амплитуду при этой частоте.

- **IExchanger** - специализированный интерфейс обеспечивающий возможность ядру комплекса настроить указатели на загруженные в память COM объекты необходимым для правильного взаимодействия объектов друг с другом:

```
STDMETHOD( GetSlotCount(int *count) ) PURE;
```

```
STDMETHOD( GetSlot(int index, CSlot *slot) ) PURE;
```

```
STDMETHOD( Start() ) PURE;
```

```
STDMETHOD( Stop() ) PURE;
```

Вследствие того, что некоторые модули комплекса являются “пассивными”, т.е. не могут самостоятельно запрашивать данные откуда либо, а только получают их, то с этой точки зрения верным является утверждение, что

управление потоками должно осуществляться “активными” модулями. Однако это утверждение противоречит разрабатываемой структуре последовательного направления потоков данных. Поэтому появилась необходимость разработать дополнительный интерфейс, с помощью которого возможно было бы осуществить настройку потока данных в самом “пассивном” модуле.

В качестве примера использования данного интерфейса можно привести модуль разделения каналов. Можно провести физическую аналогию между модулем разделения сигналов и устройством под названием мультиплексор. У последнего есть один вход и некоторое количество выходов, к которым подключаются оконечные устройства. Аналогом входа у модуля разделения сигналов является указатель на модуль сбора данных (IGetData*), инкапсулированный как член класса. В качестве множества выходов, к которым подключаются устройства, играют так называемые слоты. Как видно из описания интерфейса возможно получение количества слотов и получение информации о слотах. Правильная установка указателей на объекты, которые будут получать данные из данного модуля при запросе из “пассивного” модуля осуществляется комплексом специальным способом, через установку опций, предоставляемых модулем разделения каналов.

- IGraphics - специализированный интерфейс обеспечивающий работу графических COM компонентов:

```
STDMETHOD( SetWndPtr(CMDIChildWnd *ptr) ) PURE;
STDMETHOD( OnSize(UINT nType, int cx, int cy) ) PURE;
STDMETHOD( OnChar(UINT nChar, UINT nRepCnt, UINT nFlags) ) PURE;
STDMETHOD( OnCreate(LPCREATESTRUCT lpCreateStruct) ) PURE;
STDMETHOD( OnClose() ) PURE;
STDMETHOD( OnDestroy() ) PURE;
STDMETHOD( OnKeyDown(UINT nChar, UINT nRepCnt, UINT nFlags) ) PURE;
STDMETHOD( OnKeyUp(UINT nChar, UINT nRepCnt, UINT nFlags) ) PURE;
STDMETHOD( OnHScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar) ) PURE;
STDMETHOD( OnLButtonDblClk(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnLButtonDown(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnLButtonUp(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnMouseMove(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnMouseWheel(UINT nFlags, short zDelta, CPoint pt, BOOL *ret) ) PURE;
STDMETHOD( OnPaint() ) PURE;
```



```

STDMETHOD( OnMove(int x, int y) ) PURE;
STDMETHOD( OnRButtonDblClk(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnRButtonDown(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnRButtonUp(UINT nFlags, CPoint point) ) PURE;
STDMETHOD( OnCommand(WORD wNotifyCode, WORD wID, HWND hwndCtl) ) PURE;

```

При создании объекта модуля, поддерживающего интерфейс IGraphics, создается дочернее (MDI Child) окно и все сообщения посылаемые системой этому окну перенаправляются модулю. Таким образом, модуль может полностью контролировать весь процесс визуализации, изменения размеров окна и других различных событий. Для облегчения работы в модуле может создаваться дополнительное окно, наследуемое от CWnd, обработка событий которого ведется стандартными средствами среды программирования.

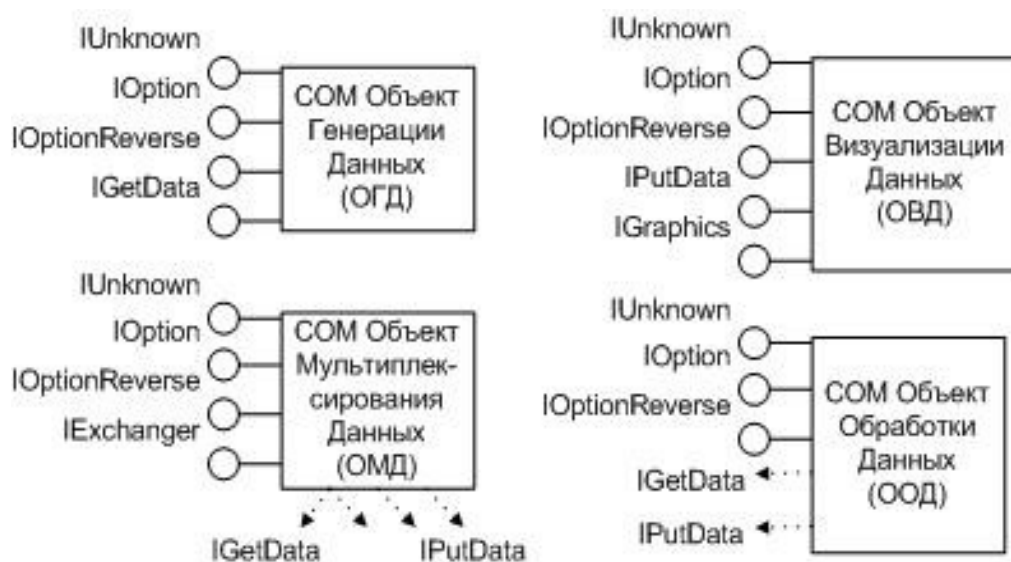


Рис.2.41. Модули расширения АПК

Модули расширения ПАИК должны быть разработаны с учетом поддержки соответствующих интерфейсов, учитывая ограничения и соглашения по разработке СОМ компонентов для ПАИК. На рис.2.4.1. представлены основные модули расширения, используемые в комплексе с указанием поддерживаемых интерфейсов и инкапсулируемых указателей на СОМ объекты. Возможны и другие сочетания поддержки интерфейсов и инкапсуляции указателей на другие СОМ объекты для реализации специализированных задач.

2.4.3. Создание нового модуля расширения

2.5 Лингвистическое обеспечение

2.5.1. Выбор лингвистического обеспечения

Сегодня существует множество широко применяемых средств создания программного обеспечения. Различные средства используются для различных классов задач.

Поскольку в качестве операционной системы была выбрана платформа на базе ядра Windows NT, среди средств лингвистического обеспечения наиболее полно подходящих к решению поставленной задачи можно выделить Microsoft Visual C++, Microsoft Visual J++, Borland Delphi, Borland C++ Builder. В связи с ограничениями виртуальной машины Java, использование Visual J++ является нецелесообразным.

Выбирая среду разработки под ОС Windows особенно следует отметить Microsoft Visual C++. Главным преимуществом данной системы, безусловно, являются ее ничем не ограниченные в рамках Windows возможности.

Приложения постоянно усложняются и требуют все большего времени для их создания. Для облегчения программирования Windows-приложений фирмой Microsoft была разработана библиотека MFC (Microsoft Foundation Classes - Базовые Классы Microsoft). Сейчас она представляет собой мощный набор классов C++, которые позволяют программировать приложения на достаточно высоком уровне абстракции, и вместе с тем открывают для опытных программистов легкий доступ к функциям более низкого уровня, что позволяет писать эффективные приложения и полностью использовать все возможности операционной системы.

MFC является альтернативой системам визуального программирования, таким как Delphi или Visual Basic. На сегодняшний день подавляющее большинство программ разрабатывается при помощи Microsoft Visual C++ и MFC. MFC - это стандарт программирования под Windows и "интернациональный язык общения". Такая ситуация объясняется многими причинами. В частности, только MFC позволяет создавать наиболее эффективные и устойчивые приложения, которые будут корректно вести себя не только в системе разработчика, но и в системах реальных пользователей. Также очень важно, что MFC поддерживает все современные технологии, реализованные в Windows, и при дополнении Windows почти сразу же дополняется и MFC.

Для графического отображения результатов решения задачи необходимо использовать одну из существующих сегодня графических библиотек (наиболее известные среди них - DirectX, Mesa, OpenGL).

В качестве такой графической библиотеки была выбрана Open Graphics Library (OpenGL). Основным преимуществом её является надежность. OpenGL – это устоявшийся стандарт, действующий уже 8 лет. Все вносимые в него изменения предварительно анонсируются и реализуются таким образом, чтобы гарантировать нормальную работу уже написанного ПО. Все приложения, использующие OpenGL гарантируют одинаковый визуальный результат вне зависимости от используемого оборудования и ОС. Приложения, использующие OpenGL, могут запускаться на персональных компьютерах, рабочих станциях или суперкомпьютерах. OpenGL хорошо структурирована. Её драйверы включают информацию об основном оборудовании [6].

2.5.2. Соглашение о разработке

В связи с тем, что программная часть разрабатываемого комплекса представляет собой объемное по исходному коду ядро и некоторое множество чуть менее объемных по размеру исходного кода модулей расширения, встает вопрос о единообразии исполнения всех участков текстов. Данное соглашение обеспечивает легкость понимания частей программного обеспечения всеми разработчиками, а также относительную легкость расширения комплекса в дальнейшем. Единообразие и четкость написания кода играет немаловажную роль при поиски ошибок в исходном коде.

Требования к методам проектирования

Модуль (далее программное обеспечение – ПО) должно разрабатываться с учетом требований компонентного подхода, который предполагает построение ПО из отдельных компонентов – физически отдельно сосуществующих частей ПО, которые взаимодействуют между собой через стандартные двоичные интерфейсы.

Разработанные объекты-компоненты, а также набор стандартных средств поддержки модуля АПК должны быть объединены в динамически загружаемые библиотеки.

Требования к структуре программ

Разрабатываемая модель должна проектироваться с учетом блочно-иерархического подхода. Каждая логическая группа функций (компонентов) должны оформляться в виде отдельного *.h файла. Декомпозиция функций (компонентов) проекта должна осуществляться с учетом следующих требований:

- Однозначности – выделение одинаковых блоков различных компонент в один общий с возможность последующего вызова.

- Непротиворечивость – обеспечение согласования компонентов между собой.
- Полнота – компоненты должны быть свободны от избыточных элементов.
- Формализация – строгость использованных методических и математических методов.
- Оптимальность – оптимальная реализации на каждом уровне иерархии.
- Минимизация степени зависимости компонент – оценка уровня сцепления и связности.

Требования к стилю программирования

Общий стиль разработки программных моделей должен обеспечить высокий уровень «технологичности» разрабатываемого исходного кода.

2.5.2.1. Правила наименования объектов программы.

- В случае, если объект является глобальным, он должен начинаться символом верхнего регистра (например Plines), если объект является членом какого то класса, то его имя должно начинаться с ключевого сочетания ‘my’ (например myPlines). В любых других случаях наименование объекта должно начинаться с символа нижнего регистра. В случае, если объект является константным рекомендуется использовать при его наименовании только символы верхнего регистра, но допускается наименование по вышеизложенным правилам.
- Наименование объекта должно соответствовать его содержанию, например: myBPF.
- Если имя объекта является сокращением, то оно задается символами верхнего регистра, если производным от собственного имени, то нижнего с первой заглавной буквой, например: myRe_FFT.
- При наименовании структурно-зависимых объектов следует использовать принцип псевдонимического наименования, когда префикс, характеризующих родителя, отделен от имени объекта знаком, например: 3Dgraf_real
- Все переменные должны объявляться в месте их непосредственного использования. Т.е. не рекомендуется использование объявления переменной без непосредственной инициализации, за исключением случаев, когда это необходимо. Допускается, но не рекомендуется, применение C-стиля объявления переменных – объявление переменных должно находиться до кода выполнения.

– Желательно использовать стиль программирования описанный в пункте 3 настоящих требований, но допускается применение общепринятого стиля программирования. В качестве примера данного стиля оформления кода можно рассматривать исходные тексты ядра Linux.

2.5.2.2. Правила оформления модулей

Каждый модуль должен начинаться заголовком, содержащим:

- Название модуля
- Краткое функциональное описание (назначение).
- Краткое описание входных и выходных параметров с указанием единиц измерений.
- Список используемых (вызываемых) модулей.
- Краткое описание алгоритма (метода) и его ограничений.
- ФИО автора программы и его координаты.
- Идентифицирующую информацию (номер версии, дату последней корректировки).

2.5.2.3. Стиль оформления текстов модулей:

2.5.2.3.1. Каждый блок кода должен иметь отступ в 4 пробела (или один знак табуляции) и расположен как указано ниже:

```
{
    some code;
    {
        some code;
    }
}
```

2.5.2.3.2. Для всех описаний, определений и вызовов функций:

2.5.2.3.2.1 скобка после имени функции следует без пробела

Пример:

```
sin( x );
if( something ) { do };
for( int i=1; i<1; i++ );
```

2.5.2.3.2.2 После открывающейся скобки (перед закрывающейся скобкой) ставится пробел, если количество предшествующий открывающихся скобок (количество последующих закрывающихся скобок) четно

Пример:

```
sin( (x+( y-(1) )) );
```

3.3. Каждый компонент исходного текста программы должен содержать заголовок в виде комментария с кратким описанием функциональности, а также информация об авторе кода, номера версии и даты последнего обновления.

Пример:

```
////////////////////////////////////  
// cmd_list.h : command list class ver.1.1  
// created: 12.11.2002  
// last edit: 25.11.2002,  
//  
// (c) A.A.Ivanov, 2002  
// email: ivanov@iu4.bmstu.ru  
//  
////////////////////////////////////
```

3.4. Критические места исходного текста программы, а также в места, где действие может быть неправильно понято исходя из контекста должны быть снабжены комментариями.

```
////////////////////////////////////  
// комментарии  
////////////////////////////////////
```

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Ю.И.Нестеров, А.И.Власов, Б.Н.Першин. Виртуальный измерительный комплекс // Датчики и системы. №4. 2000 - стр. 12 - 22.
2. Князев В.С., Афанасьев А.В. Компьютерная измерительная лаборатория // Студенческая научная конференции "Информатика и системы управления в XXI веке", Сборника научных трудов, М.: Издательство Эликс+, 2002, стр. 84 - 88
3. Афанасьев А.В. MSTU - Многофункциональный измерительный комплекс // Сборник научных трудов ч.2 молодежной научно-технической конференции "Наукоемкие технологии и интеллектуальные системы 2003", 16-17 апреля 2003г., М.: МГТУ им.Н.Э.Баумана, стр. 116 - 119.
4. А.В.Барков. Возможности нового поколения систем мониторинга и диагностики // Журнал «Металлург», № 11, 1998 г
5. А.В.Барков Н.А.Баркова. Интеллектуальные системы мониторинга и диагностики машин по вибрации // Труды Петербургского энергетического института повышения квалификации Минтопэнерго Российской Федерации и Института вибрации США (Vibration Institute, USA), Выпуск 9, Санкт-Петербург, 1999 г.)
6. Ю. Тихомиров «Программирование трехмерной графики» С-Пб. «БХВ- Санкт-Петербург» 1999, 350 с.