Описание (черновик)

Определения:

1) Измеритель – устройство для измерения поверхности образца (трубы) изнутри, отправки измеренных данных на ВУ и взаимодействия с ВУ через некоторый интерфейс управления.

2) ВУ (Верхний уровень) – ПО, осуществляющее взаимодействие с измерителем, а также обработку, сохранение и вывод результатов измерения.

3) Хранилище – БД для хранения результатов измерения с целью их последующего просмотра и обработки (например, для построения некоторой статистики).

4) Внешний клиент – некоторая система для получения данных с хранилищ ВУ и их последующего просмотра и обработки.

5) ПУ или поток управления – протокол для взаимодействия между измерителем и ВУ. Данный протокол является двусторонним и надежным (осуществляется с помощью взаимодействия по TCP).

6) ПД или поток данных – протокол для передачи данных от измерителя к ВУ. Данный протокол является односторонним (т.е. мы не передаем какие-либо данные от ВУ на измеритель) и потоковым (осуществляется с помощью взаимодействия по UDP).

7) ПХр или поток взаимодействия ВУ с хранилищем – протокол для взаимодействия ВУ с хранилищем (для передачи данных из ВУ и сохранения их в хранилище).

8) ПВнеш или поток взаимодействия внешнего клиента с хранилищем/ВУ – протокол для взаимодействия внешнего клиента с хранилищем/ВУ (для запроса данных внешним клиентом из хранилища/ВУ).

Стадии работы:

1) Ввод данных/выбор режима работы

2) Получение данных + контроль потерь (контроль неупорядоченности пакетов)

3) Калибровка

4) Обработка данных

5) Вывод результатов обработки

Варианты работы

1) Измерение данных по упрощенной схеме (на пересечениях двух перпендикулярных плоскостей и трубы):

* Ввод данных/выбор режима работы
* Калибровка
* Получение данных + контроль потерь
* Вывод полученных данных (результатов обработки)

2) Измерение данных по полной схеме:

* Ввод данных/выбор режима работы
* Калибровка
* Получение данных + контроль потерь
* Обработка данных
* Вывод результатов обработки

3) Взаимодействие с внешним клиентом:

* Запрос данных у ВУ

Ошибочные ситуации во время работы

1) Отклонение данных контролем потерь (слишком много потерь и/или слишком велика неупорядоченность пакетов):

* Ввод данных/выбор режима работы
* Калибровка
* Получение данных + контроль потерь
* Останов работы из-за отклонения данных контролем потерь
* Сообщение о данном событии

2) Уход температуры измеряемого образца из диапазона температур, при которых проходила калибровка сканера (диапазон температур определяется как температура калибровки плюс-минус некоторая дельта):

* Ввод данных/выбор режима работы
* Калибровка
* Получение данных + контроль потерь
* Получение температуры из данных, сравнение с диапазоном допустимых температур
* Останов работы из-за ухода температуры измеряемого образца из диапазона температур, при которых проходила калибровка сканера
* Сообщение о данном событии

3) Удар датчика (сканера) о поверхность измеряемого образца

* Ввод данных/выбор режима работы
* Калибровка
* Получение данных + контроль потерь
* Получение сигнала об ударе
* Останов работы из-за удара датчика (сканера) о поверхность измеряемого образца
* Сообщение о данном событии

Потоки данных/управления

Обозначения:

* ПУ или поток управления – протокол для взаимодействия между измерителем и ВУ
* ПД или поток данных – протокол для передачи данных от измерителя к ВУ
* ПХр или поток взаимодействия ВУ с хранилищем – протокол для взаимодействия ВУ с хранилищем
* ПВнеш или поток взаимодействия внешнего клиента с хранилищем/ВУ – протокол для взаимодействия внешнего клиента с хранилищем/ВУ

1) Измерение данных по упрощенной схеме:

* Ввод данных/выбор режима работы: происходит внутри ВУ
* Калибровка: ВУ--(ПУ)-->Измеритель--(ПУ)-->ВУ
* Начало измерений: ВУ--(ПУ)-->Измеритель
* Получение данных + контроль потерь: Измеритель--(ПД)-->ВУ
* Окончание измерений: Измеритель--(ПУ)-->ВУ
* Вывод полученных данных (результатов обработки): ВУ--(ПХр)-->Хранилище

2) Измерение данных по полной схеме:

* Ввод данных/выбор режима работы: происходит внутри ВУ
* Калибровка: ВУ--(ПУ)-->Измеритель--(ПУ)-->ВУ
* Начало измерений: ВУ--(ПУ)-->Измеритель
* Получение данных + контроль потерь: Измеритель--(ПД)-->ВУ
* Обработка данных: происходит внутри ВУ
* Окончание измерений: Измеритель--(ПУ)-->ВУ
* Вывод результатов обработки: ВУ--(ПХр)-->Хранилище

3) Отклонение данных контролем потерь (слишком много потерь и/или слишком велика неупорядоченность пакетов):

* Ввод данных/выбор режима работы: происходит внутри ВУ
* Калибровка: ВУ--(ПУ)-->Измеритель--(ПУ)-->ВУ
* Начало измерений: ВУ--(ПУ)-->Измеритель
* Получение данных + контроль потерь: Измеритель--(ПД)-->ВУ
* Останов работы из-за отклонения данных контролем потерь: ВУ--(ПУ)-->Измеритель
* Сообщение о данном событии: ВУ--(ПХр)-->Хранилище

4) Уход температуры измеряемого образца из диапазона температур, при которых проходила калибровка сканера:

* Ввод данных/выбор режима работы: происходит внутри ВУ
* Калибровка: ВУ--(ПУ)-->Измеритель--(ПУ)-->ВУ
* Начало измерений: ВУ--(ПУ)-->Измеритель
* Получение данных + контроль потерь: Измеритель--(ПД)-->ВУ
* Получение данных об уходе температуры измеряемого образца: происходит внутри ВУ
* Останов работы из-за ухода температуры измеряемого образца: ВУ--(ПУ)-->Измеритель
* Сообщение о данном событии ВУ: ВУ--(ПХр)-->Хранилище

5) Удар датчика (сканера) о поверхность измеряемого образца:

* Ввод данных/выбор режима работы: происходит внутри ВУ
* Калибровка: ВУ--(ПУ)-->Измеритель--(ПУ)-->ВУ
* Начало измерений: ВУ--(ПУ)-->Измеритель
* Получение данных + контроль потерь: Измеритель--(ПД)-->ВУ
* Получение сигнала об ударе: Измеритель--(ПУ)-->ВУ
* Останов работы из-за удара датчика (сканера) о поверхность измеряемого образца: происходит внутри ВУ
* Сообщением о данном событии ВУ: ВУ--(ПХр)-->Хранилище

6) Взаимодействие с внешним клиентом:

* Запрос данных у ВУ: Внешний клиент--(ПВнеш)-->ВУ--(ПВнеш)-->Внешний клиент

Ожидаемая функциональность (поведение) измерителя

От измерителя мы ожидаем следующего:

* Взаимодействие с ВУ через ПУ или поток управления. Должны поддерживаться все команды, определенные для ВУ (список команд см. ниже).
* Измерение и передача измеренных данных на ВУ через ПД или поток данных (формат см. ниже).
* Возможность калибровки сканера (с возвратом данных калибровки и температуры, при которой калибровка произведена).
* Возможность калибровки положения (с возвратом данных калибровки).
* Периодическое измерение температуры образца и передача этих данных вместе с другими данными на ВУ через ПД или поток данных (формат см. ниже).
* Сигнализация об ударе сканера о стенки измеряемого образца (через ПУ или поток управления).
* Сигнализация об окончании процесса измерения (через ПУ или поток управления).

Функциональные требования к ВУ

Функциональные требования:

* Поддержка обработки измеренных данных по упрощенной схеме (см. ниже).
* Поддержка обработки измеренных данных по полной схеме (см. ниже).
* Взаимодействие с измерителем через ПУ или поток управления (список команд см. ниже).
* Получение данных от измерителя через ПД или поток данных (формат см. ниже).
* Сохранение некоторых данных в хранилище через ПХр или поток взаимодействия ВУ с хранилищем.
* Контроль “качества” пакетов при получении данных от измерителя через ПД или поток данных.
* Управление измерителем согласно следующему протоколу (через ПУ): калибровка измерителя (в нее на данный момент входят калибровка сканера и калибровка положения), запуск процесса измерения, останов процесса измерения.

Нефункциональные требования к ВУ

Нефункциональные требования:

* Поддержка кросплатформенности на уровне сборки исходного кода: один и тот же код должен собираться под разные платформы (минимум под MS Windows и Linux).
* Суммарное время на измерение и обработку данных не должно превышать некоторого заданного значения. Предварительно известно, что должно изготовляться приблизительно 30 труб в час. Следовательно, примерно за 2 минуты мы должны измерить и обработать все данные. Отсюда возникает требование начинать обработку данных как можно раньше, не дожидаясь окончания этапа измерения. Также, по возможности следует распараллелить обработку измеренных данных.
* Возможность конфигурирования на лету. Для этого ВУ должен быть разработан с использованием архитектуры на базе plugin'ов.
* Реализация должна быть осуществлена на языке C++ (стандарт C++11) с помощью библиотеки Qt.
* Наличие тестов.

Обработка данных, измеренных по упрощенной схеме

Упрощенная схема измерений:

1. Находим ось трубы (точнее центр кругового сечения на торце трубы, после чего строим ось);
2. Проводим две перпендикулярные плоскости через ось трубы;
3. Последовательно строим сечения плоскостью, перпендикулярной этой оси трубы;
4. В каждом сечении проводим измерения в точках пересечения перпендикулярных плоскостей с внутренней поверхностью трубы и плоскостью сечения. Как итог, в каждом сечении измеряемого образца измеряется всего 4 точки.

Обработка данных, измеренных по упрощенной схеме, заключается в следующем:

* Фильтрация.
* Сравнение обработанных данных с ожидаемыми данными.

Видно, что фактически никакой обработки данных нет

Обработка данных, измеренных по полной схеме

Полная схема измерений:

1. Находим ось трубы (точнее центр кругового сечения на торце трубы, после чего строим ось);
2. Последовательно строим сечения плоскостью, перпендикулярной оси трубы;
3. В каждом сечении находим линию пересечения внутренней поверхности трубы с плоскостью сечения:
4. На каждой линии пересечения измеряем N точек, получаемых поворотом датчика вокруг оси на некоторый постоянный угол (значение угла – 2\*Pi/N).

Обработка данных, измеренных по полной схеме, заключается в следующем:

* Фильтрация.
* Вычисление положения для каждой измеренной точки (по данным 6 “энкодеров”).
* Вычисление координат для каждой измеренной точки, по вычисленному положению, данным сканера и данным калибровки.
* Вычисление внутренней поверхности (профиля) по координатам измеренных точек.

Список команд для взаимодействия с измерителем через ПУ

Есть следующие команды для взаимодействия ВУ с измерителем через ПУ:

1. Начало работы измерителя – “СТАРТ”. Входные данные: данные о способе перемещения измерителя при измерении. Возвращаемое значение: Ok.
2. Окончание работы измерителя – “СТОП”. Входные данные: отсутствуют. Возвращаемое значение: Ok.
3. Калибровка. Входные данные: данные о способе перемещения измерителя при калибровке. Возвращаемое значение: данные калибровки и температура, при которой происходила калибровка.

Список событий измерителя, передаваемых на ВУ

Есть следующие события измерителя, передаваемые на ВУ (события передаются через ПУ):

1. Удар датчика (сканера) о поверхность измеряемого образца. Дополнительные данные: отсутствуют.
2. Окончание измерения. Дополнительные данные: отсутствуют.

Формат данных в потоке данных от измерителя через ПД

Пакет с данными измерений:

* Заголовок: номер пакета, номер измерения.
* Длина данных в пакете
* Температура образца
* N точек (с координатами X, Y), измеренных сканером.
* Данные 6 энкодеров.

Архитектура системы

ВУ можно разделить на следующие части (“сервисы”):

1. Транспорт или блок для взаимодействия с измерителем и получения от него данных. Транспорт инкапсулирует знание о том, что такое к ПУ и ПД и предоставляет доступ к ним. Клиенты транспорта знают следующее. ПУ – это протокол взаимодействия между ВУ и измерителем. ВУ взаимодействует с измерителем через ПУ по принципу “запрос-ответ”, измеритель взаимодействует с ВУ передачей данных (событий) в одном направлении. ПД – это протокол передачи данных от измерителя к ВУ в одном направлении. Транспорт о формате данных знает по минимуму: только информацию, которая позволяет ему организовать передачу данных потребителю и осуществить контроль “качества” пакетов. За сериализацию и десериализацию оставшейся части данных занимается вычислитель.
2. Вычислитель. Отвечает за обработку данных и за взаимодействие с ВУ. Содержит часть GUI, специфичную для некоторой задачи (выбор режима работы, запуск измерений, обработка результатов измерений, вывод результатов), код для взаимодействия с измерителем на высоком уровне (главным образом, это относится к сериализации и десериализации данных к/от измерителя), а также код для обработки результатов измерений. Вычислитель может содержать динамически подгружаемые модули (например, для фильтров для стадии фильтрации при обработке данных).
3. Блок для взаимодействия с хранилищем.
4. Связующий каркас. Это приложение, связывающее другие части (“сервисы”) в одно целое (в одно приложение). Умеет динамически подгружать разные варианты частей (“сервисов”), позволяет взаимодействовать разным частям (“сервисам”) друг с другом и содержит еще ряд вспомогательной функциональности (возможно, что некоторую шину сообщений и т.п.).