

## **ПРИМЕНЕНИЕ КОМАНД SCPI ДЛЯ АВТОМАТИЗАЦИИ ПРОВЕДЕНИЯ ЛАБОРАТОРНЫХ СПЕЦИАЛЬНЫХ ИССЛЕДОВАНИЙ С ИСПОЛЬЗОВАНИЕМ АНАЛИЗАТОРОВ СПЕКТРА**

**Аннотация.** В статье рассматривается вопрос дистанционного управления измерительными приборами в целях автоматизации проведения лабораторных специальных исследований с использованием анализаторов спектра. Анализ был проведён для измерительных приборов компании Rohde & Schwarz, однако он с небольшими измерениями может быть распространён на любые анализаторы спектра. Показаны преимущества автоматизации задачи измерения и структура автоматизированного испытательного оборудования, проанализированы структура и синтаксис команд SCPI, исследована структура и основные функции архитектуры VISA, приведена процедура настройки подключения измерительного оборудования к ПК с использованием команд SCPI и архитектуры VISA. Также проведён анализ таких операций, как синхронизация измерений и проверка прибора на наличие ошибок, рассмотрены вопросы оптимизации скорости и выбора наиболее быстрого интерфейса связи среди поддерживаемых (Telnet, GPIB, RS232, LAN, USB-TMC) в зависимости от конкретной ситуации. На основе практического опыта автора был составлен перечень наиболее важных и часто используемых команд SCPI при автоматизации проведения лабораторных специальных исследований с использованием анализаторов спектра.

**Ключевые слова:** специальное исследование, анализатор спектра, SCPI, VISA, Rohde & Schwarz.

## **USING SCPI COMMANDS TO AUTOMATE A RUN LABORATORY SPECIAL RESEARCH WITH THE USE OF SPECTRUM ANALYZERS**

**Abstract.** The article deals with the issue of remote control of measuring instruments in order to automate special laboratory studies using spectrum analyzers. The analysis was carried out for measuring instruments from Rohde & Schwarz, however, with small measurements, it can be extended to any spectrum analyzers. The advantages of automation of measurement tasks and the structure of automated test equipment are shown. The structure and syntax of SCPI commands are analyzed. The structure and main functions of the VISA architecture have been studied. The procedure for setting up the connection of measuring equipment to a PC using SCPI commands and VISA architecture is given. An analysis of such operations as synchronization of measurements and checking the device for errors was carried out. The issues of optimizing the speed and choosing the fastest communication interface among those supported (Telnet, GPIB, RS232, LAN, USB-TMC) depending on the specific situation are considered. Based on the practical experience of the author, a list of the most important and frequently used SCPI commands in the automation of laboratory special studies using spectrum analyzers was compiled.

**Keywords:** special research, spectrum analyzer, SCPI, VISA, Rohde & Schwarz.

### **Введение**

Проведение лабораторных специальных исследований представляет из себя довольно сложную и кропотливую задачу. Для их качественного выполнения необходимо провести большое количество измерений. Однако ручное выполнение данного процесса даже на современном анализаторе спектра займёт достаточно много времени, да и, к тому же, всегда существует вероятность пропустить какую-либо частоту, на которой можно наблюдать опасный сигнал, или, напротив, интерпретировать «неопасную» частоту как «опасную». Таким образом, явно встаёт вопрос о необходимости автоматизации данного процесса. Наиболее простым и универсальным путём для решения этой задачи является использование команд SCPI (Standard Commands for Programmable Instruments, стандартные команды для программируемых инстру-

ментов, общепринято произносится как «скиппи»), которые поддерживаются почти всеми выпущенными за последние 30 лет анализаторами спектра. В данной статье будет рассмотрено применение команд SCPI для анализаторов компании Rohde & Schwarz (R&S, с нем. – «Роде и Шварц»), однако ввиду стандартизованности команд SCPI материал, изложенный в статье, может быть с небольшими измерениями распространён на любой анализатор спектра. Следует отметить, что, несмотря на популярность анализаторов спектра R&S среди органов по аттестации (в частности, они входят в состав большого числа продающихся на рынке ПАК), на русском языке практически отсутствуют материалы по работе с командами SCPI для анализаторов спектра данной компании. Таким образом, данная статья будет полезна не только сотрудникам органов по аттестации, но и всем тем, кто хочет выполнять дистанционное управление или автоматизировать выполнение каких-либо прикладных задач на анализаторах спектра компании R&S. Нужно заметить, что данная статья не будет касаться вопроса автоматизации измерительных задач с использованием языка программирования или скриптового языка, так как данный вопрос будет подробно исследован в одной из последующих статей автора.

### **Основы дистанционного управления прибором**

Рост производительности привел к резкому улучшению качества жизни за последние несколько десятилетий. Возможность производить больше с меньшими ресурсами за более короткий промежуток времени делает труд более ценным, а продукт труда более дешевым. Одним из факторов, способствующих росту производительности, является автоматизация.

Автоматизация в испытательно-измерительной отрасли стала настолько широко использоваться, что в настоящее время практически каждым современным измерительным прибором можно управлять дистанционно. Многие современные измерительные приборы даже не оснащены экраном, так как их основное назначение – дистанционное управление в автоматизированной испытательной установке.

Основные преимущества автоматизации задачи измерения:

1. Автоматизация экономит время. Если инженер потратит время на разработку приложения для удаленного управления, повторение одной и той же задачи измерения займет значительно меньше времени.

2. Инструментами можно управлять на расстоянии. Например, исследование тестируемого устройства (DUT) в температурной камере или безэховой камере RF.

3. Измерения повторяемы. Задачи измерения всегда выполняются с определенной процедурой. После правильной отладки они всегда дают воспроизводимые результаты. Это способствует более высокой достоверности теста.

4. Автоматизированные системы легче расширять. При правильной структуре инженер можете выполнить одно и то же измерение на большем количестве тестируемых устройств. Например, четырехканальный осциллограф может помочь протестировать четыре ИУ. Если ему нужно подключить больше тестируемых устройств, он можете использовать автоматический коммутационный блок.

5. Протоколирование результатов испытаний. Многие среды программирования предлагают готовые наборы инструментов для составления отчетов и обработки результатов тестирования.

6. Автоматизация избавляет от рутинной работы и позволяет вместо этого сосредоточиться на творческих частях проекта.

Оборудование, используемое для выполнения автоматизированных задач, называется автоматизированным испытательным оборудованием (АИО). АИО состоит из:

- контроллер, обычно ПК, ноутбук или промышленный компьютер;
- один или несколько измерительных приборов с дистанционным управлением;
- физический интерфейс связи между контроллером и измерительным прибором (-ами);
- программное обеспечение для удаленного управления, работающее на управляющем компьютере;
- тестируемое устройство (DUT) [1].

Структурная схема АИО приведена на рис. 1.

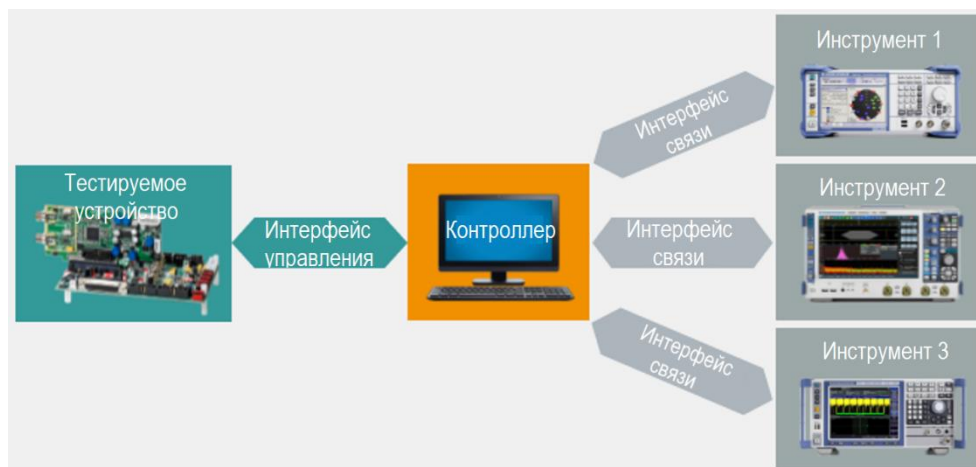


Рис. 1. Структурная схема АИО

### Команды SCPI

В 1975 году IEEE стандартизировал шину, разработанную Hewlett-Packard, которая первоначально называлась HP-IB (Hewlett-Packard Interface Bus, интерфейсная шина Hewlett-Packard), позже переименованная в GPIB (General Purpose Interface Bus, интерфейсная шина общего назначения). Стандарт назывался IEEE 488 (IEEE 488.1) и определял механические аспекты шины. Более поздний стандарт IEEE 488.2 определил свойства протокола. Однако в нём не хватало набора правил по составлению команд для управления приборами, согласованный между их производителями. Иногда команды различались даже между разными моделями одного и того же производителя.

Первой попыткой стандартизировать команды для управления приборами был язык СИЛ (Control Interface Intermediate Language, промежуточный язык интерфейса управления, в русскоязычной литературе часто неверно расшифровывается как Command Instrument Interface Language), разработанный ВВС США, однако она не увенчалась успехом.

В 1990 году Консорциум SCPI выпустил первый стандарт SCPI (Standard Commands for Programmable Instruments, стандартные команды для программируемых инструментов) в качестве дополнительного уровня стандарта IEEE-488.2.

Команды SCPI представляют собой строки ASCII, которые передаются прибору через физический уровень связи. Они могут выполнять:

- операции установки, например, команда \*RST (сброс прибора);
- операции запроса, например \*IDN? запрос (запрос идентификационной строки прибора).

Некоторые команды SCPI существуют как в виде команд установки, так и в виде команд запроса. Примером может служить команда времени захвата осциллографа «TIMEbase:ACQTime». Можно установить его с помощью команды SCPI «TIM:ACQT 10ms», а также запросить его текущее значение с помощью «TIM:ACQT?» [2].

Формат, упомянутый в руководствах пользователя Rohde & Schwarz, например: «TRIGger<m>:LEVel<n>[:VALue] <Level>», называется канонической формой. Ниже приведены самые важные правила, которые нужно знать:

1. Части в квадратных скобках не являются обязательными и могут быть опущены.
2. Заглавные буквы обязательны; маленькие буквы можно опустить. Это называется краткой формой. Приведенная выше команда в сокращенной форме выглядит следующим образом: «TRIG1:LEV2 3.5». Можно использовать либо короткую форму, либо полную форму «TRIGGER1:LEVEL2 3.5», но ничего между ними, например, «TRIGGE1:LEVE2 3.5».
3. Команды SCPI не чувствительны к регистру. Можно использовать короткую форму «trig1:lev2 3.5» или длинную форму «trigger1:level2 3.5».

4. Несколько команд можно объединить в одну строку, используя точку с запятой «;». Например, объединенная строка «TRIG1:SOUR CH1» и «TRIG1:LEV2 3.5» будет «TRIG1:SOUR CH1;LEV2». Следует заметить, что во второй команде нет части «TRIG1:». Причина в том, что путь дерева команд не меняется в пределах одной строки. Если нужно сбросить путь дерева команд к корню, можно использовать символ двоеточия в начале второй команды: «TRIG1:SOUR CH1;;CHAN2:STATE ON».

5. Вопросительный знак позволяет создавать формы запросов, чаще всего при добавлении в конец: «TRIG1:LEV2?» Иногда после вопросительного знака ставится дополнительный параметр. Между знаком вопроса и дополнительным параметром должен быть пробел. Например: "TRACE:DATA? ALL"

6. Числовые параметры без указания единиц измерения считаются основанными на единицах измерения. В данном случае числовой параметр 3,5 означает 3,5 Вольта.

## **История VISA**

В то время, когда GPIB был основным физическим интерфейсом связи, приложения удаленного управления взаимодействовали напрямую с аппаратными драйверами GPIB. Позже был представлен последовательный интерфейс RS232 как более дешевая альтернатива GPIB. Здесь также приложения взаимодействовали напрямую с аппаратными драйверами RS232.

Ситуация стала еще более сложной, поскольку интерфейс LAN стал стандартом в компьютерной индустрии. Поставщики контрольно-измерительных приборов осознали, что потенциальный интерфейс LAN может расширить возможности подключения их приборов. Однако использование третьего программного интерфейса для приборов, подключенных к локальной сети, приведет к еще большей несовместимости измерительных приложений.

Стремясь унифицировать программный интерфейс, VXIplug&play Alliance представила архитектуру VISA (Virtual Instrument Software Architecture, архитектура программного обеспечения виртуальных приборов). Текущим владельцем VISA является IVI Foundation. Для интерфейса LAN протокол на основе TCP/IP был назван VXI-11. Позже появилась его улучшенная версия под названием HiSLIP (High Speed LAN Instrument Protocol, высокоскоростной инструментальный LAN протокол). Последним дополнением стала поддержка интерфейса USB. Класс USB-устройств, изначально поддерживаемых VISA, называется USB-TMC (Test and Measurement Class, класс испытаний и измерений) [3].

## **Структура VISA**

Ниже приведен список наиболее важных функций VISA:

Функции доступа и поиска:

– VISA Open Default RM – открывает новый диспетчер ресурсов для всех удаленных подключений VISA;

– VISA Find Resource / Find Next Resource – ищет инструмент по заданным критериям;

– VISA Open/Close – открывает и закрывает удаленное подключение к инструменту;

– VISA Set/Get Attribute – устанавливает или считывает атрибут сессии.

Функции ввода/вывода:

– VISA Write – записывает командную строку в прибор;

– VISA Read – считывает строку ответа с прибора;

– VISA Clear – очищает входные и выходные буферы устройства;

– VISA Read STB – считывает системный байт состояния прибора, называемый SStatus

Byte.

На рис. 2 и 3 показаны различные уровни программного и аппаратного обеспечения, участвующие в удаленном управлении. Инструмент внизу, приложение вверху. Уровень VISA защищает приложение от специфики физического интерфейса связи. Единственным исключением, когда можно обойти VISA, является подключение через Telnet [4].

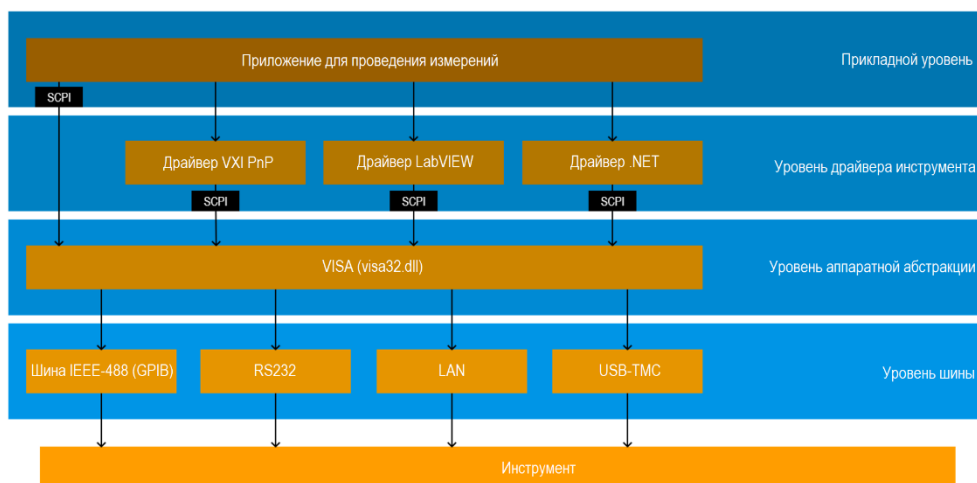


Рис. 2 Взаимосвязь компонентов в архитектуре VISA

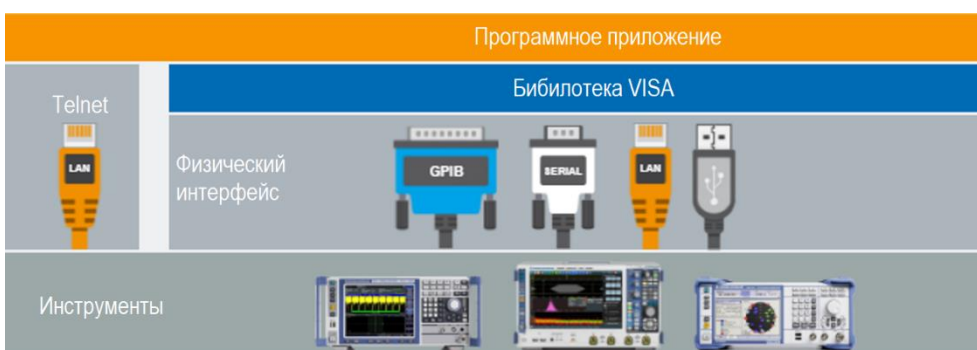


Рис. 3. Многообразие физических интерфейсов в архитектуре VISA

### Настройка подключения измерительного оборудования к ПК

Цель этого раздела – показать, как успешно произвести подключение прибора к ПК. Пример отправляет одну команду на прибор и считывает ее ответ.

Из всех различных операционных систем и физических подключений приведён наиболее распространённый случай: подключение по локальной сети в ОС Windows. Многие из описанных шагов универсальны и могут быть использованы и в других случаях.

Шаг 1. Настройте подключение по локальной сети между компьютером и инструментом.

Цель: Работаящее подключение к локальной сети

1. Подключите компьютер и инструмент к одной и той же сети LAN.
2. Используйте меню прибора, чтобы узнать или настроить его IP-адрес. IP-адрес прибора на базе Windows можно настроить в меню прибора или в Панели управления Windows -> Центре управления сетями и общим доступом.

3. Убедитесь, что инструмент доступен по локальной сети с компьютера. Чтобы убедиться в этом, используйте, например, ping-тест: используйте WinKey + R, введите cmd + ENTER, чтобы открыть командную консоль Windows. Введите ping <IP\_адрес\_инструмента >, например: ping 192.168.2.2

4. Не продолжайте дальше, если соединение TCP-IP не работает должным образом.

Шаг 2: Установите соединение VISA с инструментом

Цель: Успешный \*IDN? запрос с помощью VISA Tester Tool

1. Установите R&S VISA для Windows.
2. Откройте R&S VISA Tester: Windows Start -> R&S VISA -> RsVisaTester.
3. В поле Resource (рус. «Ресурс») введите TCPIP::::INSTR, например TCPIP::192.168.2.2::INSTR
4. Нажмите кнопку Connect (рус. «Подключить») (рис. 4).

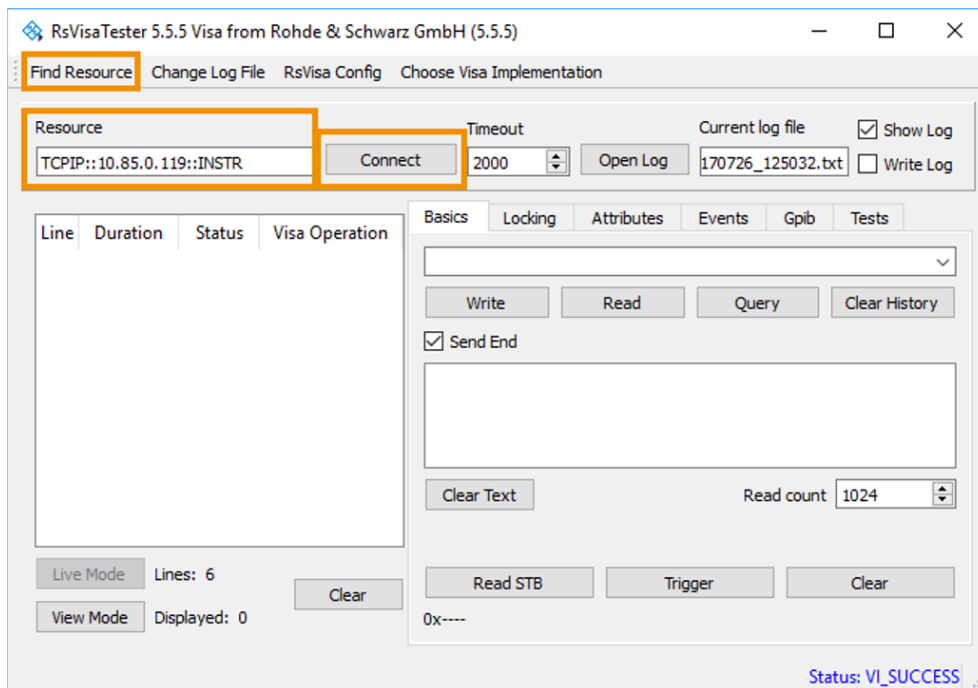


Рис 4. Установка соединения VISA с инструментом

5. На вкладке «Basics» (рус. «Основные») в поле со списком строк выберите команду \*IDN?\n и нажмите кнопку «Query» (рус. «Запрос»).
6. Прибор должен ответить строкой идентификации (рис. 5).

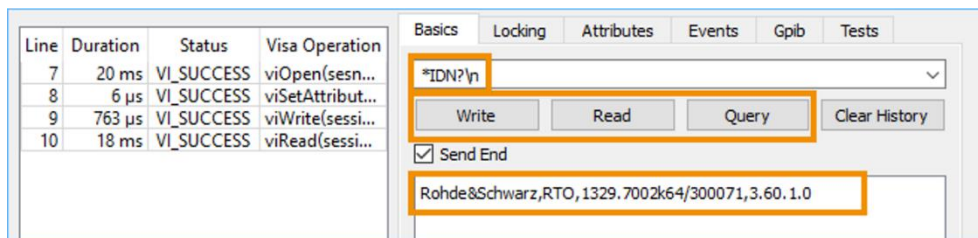


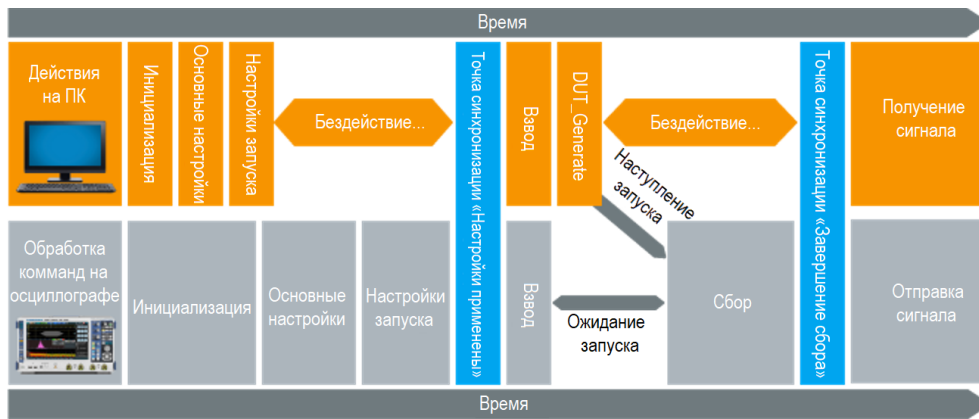
Рис 5. Получение строки идентификации прибора

### Синхронизация измерений

Синхронизация измерений является одним из наиболее важных моментов, о котором следует помнить при написании любого приложения для удаленного управления. Пренебрежение этим аспектом программы приводит к непредсказуемому поведению, неповторимым результатам и большому разочарованию. Явным признаком неправильной синхронизации измерений является то, что программисту приходится вставлять фиксированные паузы в приложение, чтобы оно работало верно.

Современные измерительные приборы представляют собой сложные устройства со своими собственными операционными системами. Измерительное приложение не должно все время заботиться о состоянии прибора. Синхронизация измерений – это метод, с помощью которого программисте может убедиться, что в важных точках программы (SyncPoints, точка синхронизации) прибор находится в ожидаемом состоянии [5].

Пример синхронизации измерений при использовании осциллографа и ИУ (исследуемого устройства), формирующего неперiodический сигнал, приведён на рис. 6.



**Рис. 6. Места размещения точек синхронизации на примере дистанционного управления осциллографом**

На изображении выше видно, что в программе есть неиспользуемые участки, которые должны ждать, пока прибор не навестает упущенное. Следует заметить, что эти периоды простоя являются динамическими по продолжительности. Они должны адаптироваться к различным условиям (например, к скорости ПК, разным временам установки и сбора данных). Как можно достичь этого? Лучше всего позволить прибору сообщать, когда он будет готов.

Дополнительные примечания:

1. Следует использовать прибор для сбора данных (осциллограф, анализатор спектра, измеритель мощности...) в режиме одиночного сбора данных. Только в этом случае можно быть уверенным, что результаты измерений исходят от последнего завершенного сбора данных, а не от предыдущего или от еще не завершенного сбора данных. Самое главное: любой метод синхронизации корректно работает только в режиме одиночного сбора данных.

2. Базовые настройки и настройки триггера не нужно синхронизировать. Единственный важный ключевой момент находится в конце, когда нужно убедиться, что все настройки были применены (точка синхронизации «Настройки применены»).

3. После поступления триггера осциллограф начинает регистрацию сигнала. Программа должна дождаться его завершения (точка синхронизации «Завершение сбора»).

4. Чтение сигнала после этой точки синхронизации гарантирует, что результат исходит от самого последнего сбора данных.

Отметим, что приборы поддерживают 4 различных механизма синхронизации:

- \*OPC? запрос;
- Опрос бита состояния (STB, Status Byte);
- Ожидание запроса на обслуживание (SRQ, Service Request);
- Событие запроса на обслуживание (SRQ, Service Request).

### **Проверка прибора на наличие ошибок**

Проверка ошибок прибора (далее называемая проверкой ошибок) является вторым наиболее важным аспектом приложения дистанционного управления. Целью проверки ошибок является обеспечение безошибочной связи с прибором, чтобы все необходимые настройки были выполнены правильно, а полученные результаты измерений были получены с использованием нужных настроек.

Частью подсистемы состояния прибора является буфер сообщений об ошибках, который называется Error Queue. Если программист, например, отправляет команду SCPI, которую прибор не распознает, он создает новую запись в очереди ошибок. Очередь ошибок имеет конечную длину, обычно 10 записей, и самая старая запись отбрасывается, чтобы освободить место для новой [6].

Если программист использует драйверы приборов, проверка ошибок уже реализована и выполняется в каждой функции и после установки каждого свойства (атрибута). Это очень

удобно, так как программисту сразу сообщается, что за ошибка произошла и после какой команды она произошла. Если ему нужна максимальная скорость, драйверы предлагают возможность отключить эту функцию.

Если программист использует прямые команды SCPI, ему нужно позаботиться о проверке ошибок самостоятельно. Рекомендуется выполнять проверку ошибок после логического блока команд. Следует иметь в виду, что чем больше команд отправляется в одном блоке, тем сложнее указать на конкретную команду, вызвавшую ошибку.

### **Оптимизация скорости**

Под оптимизацией скорости применительно к приложениям дистанционного управления подразумевается выполнение задачи измерения как можно быстрее, но при этом с надежными и воспроизводимыми результатами.

При оптимизации скорости всегда следует учитывать накладные расходы приложения по отношению к продолжительности измерения. Если, например, время захвата данных осциллографом составляет 2 мс, сокращение общих накладных расходов с 5 мс до 3 мс приведет к увеличению скорости примерно на 30%. Однако, если время сбора данных составляет 2 секунды, такое же снижение служебных данных приводит к увеличению скорости только на 0,1% [6].

Советы о том, как повысить скорость выполнения задачи измерения:

1. Никогда не следует жертвовать правильной синхронизацией измерений ради скорости. В конце концов, на отладку проблем и повторные измерения уходит больше времени, чем на то, чтобы сделать это правильно в первый раз.

2. Если в тестовой установке используется более одного измерительного прибора, можно попробовать распараллелить задачи. Например, можно настроить генератор сигналов и осциллограф параллельно и синхронизировать их только перед сбором данных.

3. Не стоит выполнять проверку ошибок прибора в циклах с большим количеством повторений и коротким временем сбора данных (менее 10 мс). Если программист использует драйверы прибора, он может выключить проверку ошибок перед циклом и снова включить её после цикла.

4. Желательно выключить дисплей прибора после отладки приложения. Обычно это улучшает работу инструмента.

5. Не следует читать целые трассы или осциллограммы. Приборы Rohde & Schwarz предлагают множество возможностей для оценки кривых/форм сигналов. Инженеру не нужна каждая точка осциллограммы для оценки частоты, амплитуды или времени нарастания импульса. Прибор обеспечивает готовые измерения таких параметров.

6. Рекомендуется использовать функции усреднения или максимального удержания прибора для трасс и осциллограмм вместо выполнения этих операций в приложении. Это экономит время на передачу данных и затраты на запуск измерений.

7. Желательно применять ограничительные линии и функции маски в тех случаях, когда нужно оценить только простые критерии прохождения/непрохождения. Передача всей трассировки и ее оценка в приложении не требуется.

8. Приборы Rohde & Schwarz поддерживают параллельную обработку команд. Всегда следует сначала устанавливать параметр, который занимает максимальное время. Если, например, сначала установить ручной аттенюатор анализатора спектра, затем центральную частоту и полосу разрешения, будет затрачено меньше времени, чем при установке их в обратном порядке.

9. Можно объединить несколько команд SCPI (разделенных точкой с запятой) в одну строку для отправки на прибор; Это быстрее, чем отправлять их по отдельности. Кроме того, это позволяет прибору определять наиболее оптимальный порядок выполнения команд. Пример уже упомянутых трех параметров анализатора спектра – механический аттенюатор, центральная частота, настройки ширины полосы разрешения: `INP:ATT 50;:FREQ:CENT 2GHz;:BWIDth 5MHz`.



## Выбор наиболее быстрого интерфейса связи

На выбор физического интерфейса связи влияет множество факторов. В этом разделе сосредоточимся исключительно на одном параметре – скорости.

При рассмотрении скорости любого устройства связи следует разделять общий термин «Скорость» на два разных параметра: задержка и пропускная способность. Чтобы объяснить их значение, воспользуемся аналогией с водопроводной трубой:

Длина трубы определяет ее латентность – время, необходимое для транспортировки небольшой единицы воды из точки А в точку Б. Следовательно, латентность измеряется в секундах.

Диаметр трубы определяет ее пропускную способность – количество воды в секунду, которое может пройти через ее поперечное сечение. Таким образом, пропускная способность измеряется в объемах в секунду.

Из этой аналогии видно, что небольшое количество воды (маленький объем данных) транспортируется быстрее по короткой трубе (малая задержка). Его диаметр (пропускная способность) существенной роли не играет. Напротив, труба большого диаметра (высокая пропускная способность) транспортирует большой объем воды (большие данные) быстрее, даже если она длиннее (большая задержка). На рис. 7 показана задержка и пропускная способность наиболее распространенных коммуникационных интерфейсов.













Интерфейс	GPIB	LAN	USB	RS232
				
Пропускная способность	1.8Mbit/s (488.1) 	12.5 Mbit/s 125 Mbits/s (GB LAN) 	60 Mbit/s (Hi-Speed) 120 Mbit/s (USB 3.0) 	28.8 kB/s 
Задержка (информативно)	300 µs 	250 µs 	1000 µs (USB) 125 µs (Hi-Speed) 	~100 ms 

Рис. 7. Сравнение пропускной способности и задержки различных интерфейсов

Из вышеизложенного можно сделать вывод об общем правиле:

Если приложение удаленного управления выполняет много небольших обменов данными между управляющим ПК и прибором, например, запрашивая множество измерений амплитуды осциллографа, интерфейс GPIB или USB является оптимальным выбором из-за их низкой задержки.

Если необходимо передать большие объемы данных, например целые осциллограммы с десятками тысяч отсчетов, кратчайшего времени передачи можно достигнуть с 1 Гб LAN или интерфейсом USB из-за их высокой пропускной способности.

Следует иметь в виду, что на общую производительность приложения удаленного управления влияет множество других факторов, помимо скорости коммуникационных интерфейсов. На практике очень часто задержки, вызванные другими факторами, перевешивают задержки, вызванные каким-либо коммуникационным интерфейсом. Цель этого раздела – просто дать общее представление о том, как разные интерфейсы ведут себя с точки зрения скорости при разных объемах данных.

Задержка, указанная на рис. 6, может значительно различаться в зависимости от внутренней архитектуры прибора. Цифры, указанные на рисунке, являются чистыми значениями задержки интерфейса. Единственный способ узнать реальную задержку измерительной уста-

новки – выполнить оценку, например, с помощью VISA Trace Tool. Одной из удобных процедур её измерения является запрос \*IDN? из-за быстрого времени обработки команды.

Отметим, что протокол HiSLIP обеспечивает меньшее время ожидания по сравнению с VXI-11. Всегда следует отдавать предпочтение HiSLIP, а не VXI-11, если инструмент его поддерживает. Однако следует заметить, что большинство распространённых среди органов по аттестации анализаторов спектра компании R&S не поддерживают протокол HiSLIP [4, 7].

### Наиболее важных команд SCPI при автоматизации проведения лабораторный специальных исследований

Команды SCPI, которые, исходя из практического опыта автора, являются наиболее важными и часто используемыми при автоматизации лабораторный специальных исследований с использованием анализаторов спектра, приведены в табл. 1. Числовые параметры (<value>), которые используются в данных командах, а также единицы измерений, следует выбирать исходя из задач автоматизации, особенностей приборов, а также требований нормативных руководящих документов.

Таблица 1

Синтаксис команды	Описание команды
*RST	Устанавливает настройки прибора в состояние по умолчанию
*IDN?	Выполняет запрос идентификационной строки прибора
*OPC?	Выполняет *OPC? запросы синхронизации
SYST:DISP:UPD <ON   OFF>	Производит включение/выключение дисплея
INIT:CONT <ON   OFF>	Устанавливает непрерывный/однократный режим развёртки
INIT	Выполняет измерения
INP:ATT <value>dB	Устанавливает величину ослабления сигнала
DISP:TRAC:Y:RLEV <value>dB	Устанавливает опорный уровень
CALC:UNIT:POW <dBm   dBmV   dBμV   dBμA   dBpW   Volt   Ampere   Watt >	Устанавливает единицу измерения по оси уровня
FREQ:CEN <value><Hz   kHz   MHz   GHz>	Устанавливает центральной частоты
FREQ:SPAN <value><Hz   kHz   MHz   GHz>	Устанавливает полосу обзора
FREQ:SPAN:FULL	Устанавливает полную полосу обзора
FREQ:SPAN 0Hz	Устанавливает нулевую полосу обзора
BAND <value><Hz   kHz   MHz   GHz>	Устанавливает полосы пропускания (разрешения)
BAND:VID <value><Hz   kHz   MHz   GHz>	Устанавливает полосы видеофильтра
TRACe<1...6>	Выбирает активную кривую
SWE:COUN <value>	Устанавливает количество усреднений
SWE:POIN <value>	Устанавливает количество измеряемых значений за один цикл развёртки
DISP:TRAC:MODE <WRIT   MAXH   MINH   AVER   VIEW>	Устанавливает режим выбранной кривой
DISP:TRAC OFF	Скрывает выбранную кривую
DET <APE   POS   NEG   RMS   AVER   SAMP   QPE>	Устанавливает детектор для выбранной кривой
FORM <ASCii   REAL   UINT   MATLAB [,8   32]>	Выбирает формат данных, передаваемых из прибор на управляющий компьютер.
TRAC? TRACE<1...4>	Возвращает текущие данные кривой или результат измерений.

## **Заключение**

Применение команд SCPI в целях автоматизации проведения лабораторных специальных исследований является довольно эффективным подходом. Их использование совместно с архитектурой VISA позволяет проводить дистанционное управление прибором с помощью различных видов интерфейсов (Telnet, GPIB, RS232, LAN, USB-TMC). Однако несоблюдение таких рекомендаций, как синхронизация команд, проверка прибора на наличие ошибок, оптимизация скорости, а также выбор наиболее быстрого интерфейса в зависимости от особенностей измерений может создать ложное впечатление о том, что команды SCPI и архитектура VISA неудобны в работе и создают много труднообнаружимых ошибок в программах. В статье предлагаются рекомендации по рациональному применению команд SCPI, позволяющие сотрудникам органа по аттестации создать эффективно работающую систему автоматизации лабораторных специальных исследований.

## **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. Rauscher C., Janssen V., Minihold R. Fundamentals of spectrum analysis. Munich, Rohde & Schwarz, 2021. 205 p.
2. Güttge F. R&S VISA. Munich, Rohde & Schwarz, 2021.
3. Ganzert J. Hints and Tricks for Remote Control of Spectrum and Network Analyzers. Munich, Rohde & Schwarz, 2017.
4. Braunstorfinger T. Connectivity of Rohde & Schwarz Signal Generators. Munich, Rohde & Schwarz, 2018.
5. Tröster C. Top Ten SCPI Programming Tips for Signal Generators. Munich, Rohde & Schwarz, 2019.
6. Schütze F., Gerlach O., Bues D. RSCommander. Versatile Software Tool for Rohde & Schwarz Instruments. Munich, Rohde & Schwarz, 2021.
7. Engelbrecht J., Lienhart K., Kiener G. Fast Remote Instrument Control with HiSLIP. Munich, Rohde & Schwarz, 2017.

## **REFERENCES**

1. Rauscher C., Janssen V., Minihold R. Fundamentals of spectrum analysis. Munich, Rohde & Schwarz, 2021. 205 p.
2. Güttge F. R&S VISA. Munich, Rohde & Schwarz, 2021.
3. Ganzert J. Hints and Tricks for Remote Control of Spectrum and Network Analyzers. Munich, Rohde & Schwarz, 2017.
4. Braunstorfinger T. Connectivity of Rohde & Schwarz Signal Generators. Munich, Rohde & Schwarz, 2018.
5. Tröster C. Top Ten SCPI Programming Tips for Signal Generators. Munich, Rohde & Schwarz, 2019.
6. Schütze F., Gerlach O., Bues D. RSCommander. Versatile Software Tool for Rohde & Schwarz Instruments. Munich, Rohde & Schwarz, 2021.
7. Engelbrecht J., Lienhart K., Kiener G. Fast Remote Instrument Control with HiSLIP. Munich, Rohde & Schwarz, 2017.

## **Информация об авторе**

*Киргизбаев Владислав Павлович* – студент магистратуры кафедры «Информационные системы и защита информации», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: v.p.kirgizbaev@gmail.com

## **Information about the author**

*Kirgizbaev Vladislav Pavlovich* – graduate student of the Department «Information Systems and Information Security», Irkutsk State Transport University, Irkutsk, e-mail: v.p.kirgizbaev@gmail.com