

ОБ АВТОМАТИЧЕСКОМ ВОССТАНОВЛЕНИИ СОЕДИНЕНИЙ В СИСТЕМАХ УПРАВЛЕНИЯ КРУПНЫМИ ФИЗИЧЕСКИМИ УСТАНОВКАМИ

Д. Ю. Болховитянов^{а, б, 1}, П. Б. Чеблаков^{а, б}

^а Институт ядерной физики им. Г. И. Будкера СО РАН, Новосибирск, Россия

^б Новосибирский государственный университет, Новосибирск, Россия

Все современные системы управления крупными физическими установками являются распределенными. Связь между узлами осуществляется посредством сети Ethernet либо через какие-то специфические линии связи — CAN, RS232/485, GPIB и т. п. Во время штатной работы все узлы активны и связь между ними установлена и функционирует. Однако в процессе эксплуатации часть узлов может становиться недоступной вследствие разных причин: отказа электроники, отказа питания, обрыва линий связи, человеческих ошибок и т. д. Таким образом, встает задача восстановления связи между узлами (после устранения проблем). В статье обсуждается реализация автоматического восстановления связей в различных ситуациях на примере системы управления инжекционным комплексом ВЭПП-5 в ИЯФ СО РАН.

All modern control systems for large experimental facilities are distributed. Nodes are connected via Ethernet or via some specific links — CAN, RS232/485, GPIB, etc. During normal operation, all nodes are online, connections between them are established and work. However, during operation some nodes can become unavailable due to various reasons: electronics failure, power failure, communication lines breaks, human errors, etc. Thus, a task of restoring connections between nodes arises (after initial problems are solved). Implementation of automatic reconnects in various situations is discussed, based on experience of VEPP-5 Injection Complex control system at BINP.

PACS: 07.05.Hd; 07.05.Wg; 29.50.+v

ВВЕДЕНИЕ

На небольших установках, где система управления (СУ) включает всего несколько узлов, решение этой задачи обычно возлагается на пользователя. Например, перед началом работы он должен проверить статус (несколько экранных «лампочек», могущих быть зелеными в норме либо красными при проблемах) и в случае неполадок нажать кнопку («Включить», «Восстановить» или «Соединиться»), по которой система попыталась бы установить все нужные связи.

Однако на больших установках, где количество узлов и компонентов достигает десятков и сотен, а число связей между ними — сотен и тысяч, возлагать эту работу

¹E-mail: D.Yu.Bolkhovityanov@inp.nsk.su

на пользователя нерационально¹. Система управления должна обеспечивать *автоматическое* восстановление связей при первой возможности.

Далее обсуждается реализация автоматического восстановления связей в различных ситуациях на примере СУ СХv4 [1], используемой на инжекционном комплексе ВЭПП-5 [2] в ИЯФ СО РАН.

ОБЩЕЕ ПРАВИЛО: ПОЛНАЯ ИНИЦИАЛИЗАЦИЯ

После восстановления соединения проводится полная инициализация без учета предыдущего состояния устройств и предыдущих известных данных.

Это включает следующие аспекты.

1. Настройка (инициализация) режимов работы. При восстановлении связи с устройством полностью прописывается режим его работы, при надобности выполняется сброс в начальное состояние (например, АЦП с внешним запуском мог находиться в ожидании внешнего сигнала и без сброса никак не реагировать на команды записи параметров).

При восстановлении связи прикладной программы с сервером СУ производится подписка на все интересующие каналы (в СХv4 это обеспечивает клиентская библиотека).

2. Вычитывание текущих значений. Необходимо вычитать текущие значения всех каналов записи, например текущие значения, выставленные в ЦАП.

Для «непростых» устройств определяется текущий режим работы. Например источники, управляемые САН-модулями, могут как находиться в отключенном состоянии, так и выдавать ток. ПО старается прочесть режим из устройства и максимально точно приспособиться к нему, чтобы сбои связи не влияли на функционирование установки.

Прикладные программы запрашивают последние известные значения каналов чтения (в СХv4 это также обеспечивает клиентская библиотека).

СВЯЗЬ ПО ТСР

Для построения компьютерных сетей в настоящее время практически повсеместно используется Ethernet, в том числе для связи с управляющей и измерительной электроникой. В качестве протокола связи, соответственно, в основном применяется ТСР.

С программной точки зрения связь по ТСР осуществляется следующим образом.

- Программа-сервер открывает так называемый слушающий сокет, назначая ему заранее известный номер порта.

¹На начальных этапах создания СХ автоматическое восстановление отсутствовало и при каждом обрыве требовалось вмешательство оператора. Поскольку ВЭПП-5 в то время был в процессе строительства (и не все системы были созданы), большого влияния на работу это не оказывало, но требовало избыточных трудозатрат. В дальнейшем же такой подход стал неприемлем, и в СХ добавилось автоматическое восстановление связей во всех возможных точках.

- Программа-клиент создает «обычный» сокет и запрашивает соединение (connect) с интересующим ее портом на нужном узле.

- После установления соединения программы обмениваются данными, выполняя чтение и запись каждая со своим сокетом.

- В случае обрыва соединения каждая программа получает уведомление об этом.

- Программа-клиент при невозможности установить соединение также уведомляется.

Соответственно, для автоматического восстановления соединений по TCP может использоваться следующий несложный алгоритм.

- При обрыве соединения следует выдержать небольшую паузу (обычно несколько секунд), а затем повторить попытку соединения.

- При успешном соединении вернуться к нормальной работе.

- При неуспешном соединении в любом варианте — таймаут, недоступность сети, недоступность узла и т. п. — следует предпринять те же действия, что при обрыве соединения: выдержать паузу и повторить попытку соединения (и так до тех пор, пока соединение не будет установлено).

- Следствие: сокеты желательно использовать в так называемом неблокирующем режиме (non-blocking, флаг O_NONBLOCK).

Этот подход, кроме CX, с некоторыми вариациями успешно используется в большинстве современных СУ, в том числе в EPICS [3] и TANGO [4].

«Узел не найден» следует считать временной ошибкой, а не фатальной. Многие программы рассматривают ошибки при разрешении имен хостов как перманентные (фатальные). Причина, очевидно, в предположении, что ошибка вызвана человеческой опечаткой — неверно набранным именем. Однако для систем управления такой подход не годится. Ненайденность узла может быть вызвана множеством причин, являющихся временными: отсутствие связи (невключенный свитч, не успевший запуститься VPN, . . .), невключенность или сбой DNS-сервера и т. д. Соответственно, такая ошибка должна рассматриваться как временная, и попытки поиска должны с некоторой периодичностью повторяться аналогично попыткам восстановления связи по TCP.

В современных СУ данный аспект обрабатывается по-разному.

- EPICS по умолчанию использует для разрешения имен каналов широковещательные UDP-запросы, в ответах на которые сразу содержатся IP-адреса узлов. Таким образом, проблемы с разрешением имен отсутствуют.

- TANGO считает любые начальные ошибки при создании объекта DeviceProху (не только «узел не найден», но и, например, незапущенность сервера БД) фатальными. Соответственно, там задача повтора соединений ложится на программиста.

- CX считает любые ошибки соединения, включая «узел не найден», временными и повторяет попытки соединения в фоновом режиме. Также есть вариант поиска каналов через широковещательные UDP-запросы, аналогично EPICS.

Ускоренное обнаружение обрывов связи. При разработке TCP главным приоритетом было обеспечение надежности. TCP-соединение, по которому не идет передача, может переживать многочасовые отключения связи и потом продолжать работу. Соединение может оставаться в бездействующем состоянии бесконечно долго, даже если узел на другом конце отключен. Хотя при «корректном» завершении работы узел присылает уведомление о закрытии соединения, но при отключении питания на

нем или обрыве связи никаких уведомлений не поступит, и такое положение будет продолжаться до первой попытки записи в сокет.

В СУ физических установок такая ситуация неприемлема. Поэтому используются следующие приемы, позаимствованные в обычном системном ПО (например, ssh).

1. Большинство реализаций TCP поддерживают опцию SO_KEEPALIVE, при включении которой раз в 7200 с посылается специальный пустой TCP-пакет. Соответственно, запись приводит к обнаружению обрывов. Но 2 ч — очень большой интервал, и он не настраивается.

2. Технология «application pings» аналогична предыдущему варианту, но «пустые пакеты» посылаются самой программой (или библиотекой). Что именно считать «пустым пакетом», определяется конкретным прикладным протоколом (например для текстового это может быть пустая строка). Здесь интервал выбирается уже программно и обычно лежит в диапазоне от 10 до 600 с.

CX использует оба приема в соединениях как клиент–сервер, так и сервер–контроллеры.

EPICS также использует оба варианта, а TANGO ограничивается вторым.

СВЯЗЬ ПО UDP

К использованию UDP для связи с электроникой могут быть следующие предпосылки.

- TCP — сложный протокол, и примитивные микроконтроллерные его реализации часто страдают низким качеством и ненадежностью. UDP существенно проще в реализации.

- TCP оптимизирован для надежности в ущерб времени отклика (латентности) и скорости обнаружения ошибок. UDP же позволяет организовать общение с электроникой любым образом.

В UDP нет понятия «соединение», но на прикладном уровне есть заменяющее его понятие сессии, за состоянием которой нужно следить, обнаруживая потери пакетов, отключения/перезагрузки устройств и прочие проблемы.

- Для диагностики потери пакетов каждый из них должен иметь последовательно увеличивающийся номер. В том числе при передаче устройством (пикапы, CCD-камеры) больших объемов информации пачкой пакетов (вместо запрос–ответ), это позволяет обнаруживать «выпавшие» группы. Здесь в прикладном протоколе необходима возможность перезапросить из устройства чтение произвольного блока данных («с такого-то по такой-то адрес»).

- Отключения обычно обнаруживаются по таймауту, когда от устройства некоторое время нет пакетов.

- Если в устройстве предусмотрена не влияющая на его работу ячейка, в которую можно записать произвольное число, то запись в нее при инициализации «магического числа» с последующим периодическим контрольным чтением позволяет легко обнаруживать перезагрузки.

На ВЭПП-5 UDP используется для связи с пикапами линака и CCD-камерами. Внедрение перечисленных мер существенно повысило надежность работы этих устройств.

RS232/RS485 И АНАЛОГИЧНЫЕ ПОСЛЕДОВАТЕЛЬНЫЕ ЛИНИИ СВЯЗИ

Эти недорогие последовательные интерфейсы используются в случаях небольших объемов передаваемых данных. Например, на ВЭПП-5 контроллеры шаговых двигателей (ШД) подвижек подключены через RS485, причем одной линией (RS485 это допускает, в отличие от RS232).

Однако помимо малой скорости у этих шин есть существенный недостаток: отсутствует аппаратный арбитраж, так что общение возможно лишь в режиме «ведущий–ведомый» (запрос–ответ), никакие же асинхронные сообщения по инициативе устройств невозможны.

Поэтому для надежного мониторинга состояния таких устройств нужно постоянно вести опрос их состояния, причем всех (по очереди), а во время исполнения команд (например при движении ШД) повышать частоту опроса.

CAN-bus

Шина CAN, первоначально разработанная для автотранспорта, оказалась чрезвычайно удобна для передачи небольших объемов данных (таких, как единичные значения ЦАП и АЦП). Достоинство CAN — в наличии аппаратного арбитража, позволяющего асинхронный обмен сообщениями.

В ИЯФ эксплуатируется широкая номенклатура CAN-устройств собственной разработки, объединенных собственным протоколом, получившим условное название CAN-BINP [5]. Его ключевой особенностью является наличие специального пакета (0xFF), высылаемого устройствами при важных событиях — включении, сбросе, переподключении.

Отслеживание уведомительных пакетов существенно упрощает ПО, оставляя лишь задачу отслеживания зависших/отключенных устройств, аналогично тому, как это делается с UDP.

Несоответствие типа устройства — временная ошибка, а не фатальная. Отдельного внимания требует проверка соответствия типа реального устройства указанному в конфигурации. В первых версиях ПО такое несоответствие считалось фатальной ошибкой. Современный вариант рассматривает это как временную ошибку и переходит в режим периодического опроса: если три проверки подряд покажут соответствие типа, то проблема считается решенной. Поскольку основная причина подобных несоответствий — ошибки установки перемычек в устройствах (и редкие ошибки сети), то после устранения ошибок система автоматически переходит в рабочее состояние.

ЗАКЛЮЧЕНИЕ

Следует отметить, что процесс начального включения установки похож на восстановление после неполадок: не все узлы включаются сразу и одновременно, а обычно одни раньше других, и не все соединения можно установить сразу.

Таким образом, автоматическое восстановление соединений также решает и задачу автоматизации начального включения машины.

СПИСОК ЛИТЕРАТУРЫ

1. *Bolkhovityanov D. et al.* CXv4, a Modular Control System // Proc. of ICALEPCS-2015, Melbourne, Australia, 2015. P. WEPGF093; doi:10.18429/JACoW-ICALEPCS2015-WEPGF093.
2. *Maltseva Yu. I. et al.* VEPP-5 Injection Complex Performance Improvement for Two Collider Operation // Proc. of RuPAC-18, Protvino, Russia, Oct. 1–5, 2018; doi: 10.18429/JACoW-RUPAC-2018-TUZMH02.
3. EPICS — Experimental Physics and Industrial Control System. <https://epics.anl.gov/>.
4. TANGO Controls. <https://www.tango-controls.org/>.
5. *Козак В. П.* Особенности использования интерфейса CANbus в ИЯФ. <http://www.inp.nsk.su/~kozak/appnotes/anc10.pdf>. 2005.