

ОБЗОР ФАКТОРОВ, ВЛИЯЮЩИХ НА ЧАСТОТУ ПОЛЛИНГА ДАННЫХ С УСТРОЙСТВ В РАСПРЕДЕЛЕННЫХ ВСТРАИВАЕМЫХ СИСТЕМАХ С ШИННЫМ ИНТЕРФЕЙСОМ СВЯЗИ

Россия, г. Пенза, Пензенский государственный технологический университет

This article discusses polling as a way to poll slaves in embedded systems. It is noted that one of the most important points in these systems is how often the master can poll the entire set of slaves in order to receive actual data from them. It also discusses what parameters affect this polling rate, how significantly, which of them can be neglected and under what conditions.

Введение. Современные встраиваемые системы находятся в более тесном контакте с физической средой, чем их предшественники. Кроме того, на них часто возлагается задача непрерывного анализа физических процессов, происходящих в физической среде вокруг, отслеживания состояния внешней среды и сбора информации о реальном мире. Зачастую также подразумевается, что встраиваемая система будет оказывать на внешнюю среду свое влияние, т.е. каким-либо образом управлять окружающей средой. И объединять информацию от различных датчиков, распределенных во внешней среде, для лучшего понимания происходящего и выполнения более точных «продуманных» действий, необходимых на текущий момент. Для решения подобных задач необходимо, чтобы вычислительные элементы встраиваемой системы осуществляли взаимодействие с датчиками, которые обеспечили бы мониторинг физических показателей среды, необходимо внедрение исполнительных элементов, которые вносили бы во внешнюю среду изменения.

При работе с встраиваемыми системами довольно часто применяется способ обмена данными по шине данных. Одним из примеров такого подключения, может служить объединение устройств с использованием протокола modbus “поверх” RS-485.

Как и в данном протоколе, так и во многих других подобных, обмен данными инициализируется управляющим устройством - “ведущим” или “мастером”. Тогда как все остальные устройства в данной системе выступают в роли “ведомых”. И одним из важнейших пунктов в данных системах является то, как часто ведущее устройство может опрашивать весь набор ведомых устройств, чтобы получать с них актуальные данные.

То, какие параметры влияют на эту частоту опроса, насколько значительно, какими из них можно пренебречь и при каких условиях будет рассмотрено в данной статье.

Поллинг. В информатике дается определение понятию операции поллинга (polling, polled operation) или же опроса, как определение состояния внешнего устройства с помощью клиентской программы как синхронного действия.

Поллинг – это процесс, в котором компьютер или управляющее устройство ожидает, пока внешнее устройство проверит его готовность или состояние.

Циклом поллинга называется время, в течение которого каждый элемент проверяется один раз. Оптимальный цикл поллинга будет зависеть от нескольких факторов, таких как желаемая скорость ответа и накладные расходы на опрос (например, время процессора и полоса пропускания).

Альтернативой поллингу является использование прерываний (рисунок 1), которые представляют собой сигналы, генерируемые устройствами или процессами, чтобы сигнализировать о необходимости обработать действие, связанное с этим устройством.

Анализу временных характеристик систем, использующих поллинг, прерывания и операционную систему реального времени (RTOS) посвящена статья [1].

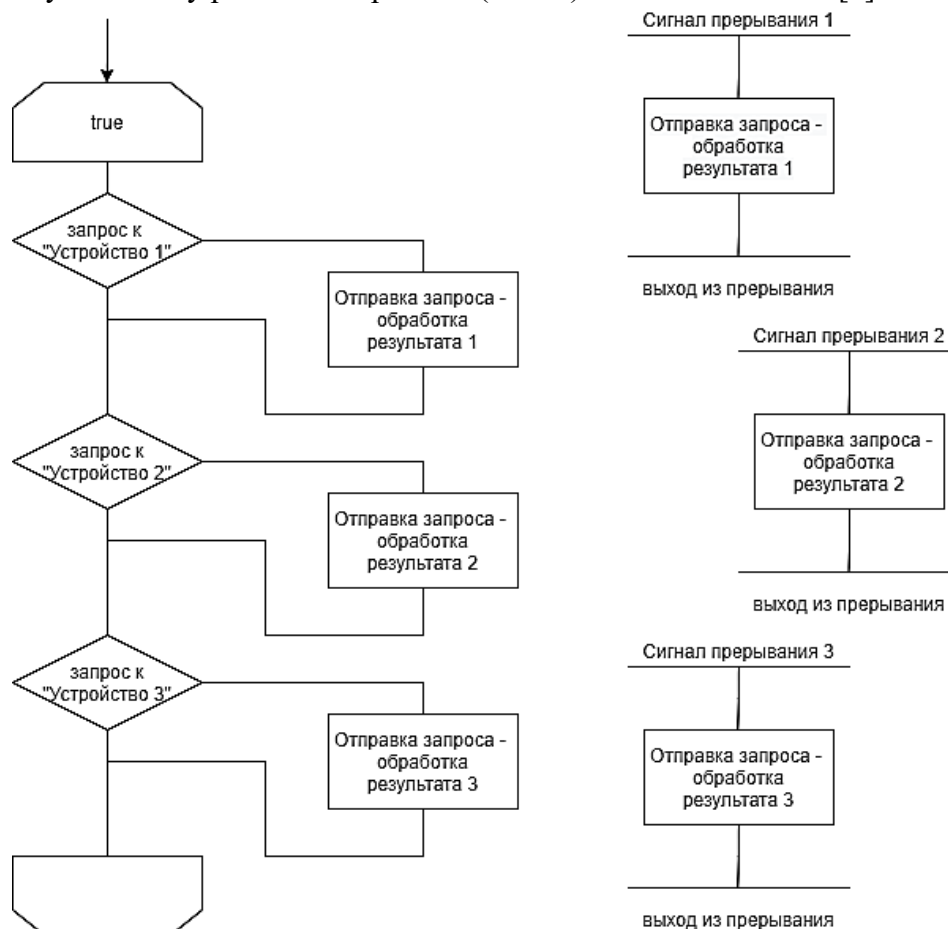


Рисунок 1 – Сравнение алгоритмов для выполнения запросов с использованием поллинга и прерывания

Опрос периферийных устройств достаточно часто встречается в системах цифрового управления объектами и в информационных системах [2, 3]. Алгоритмы, реализующие опрос периферийных устройств, обладают целым рядом особенностей:

- алгоритмы являются циклическими, т.е. они имеют оператор начала, но не имеют оператора окончания, окончание вычислительного процесса производится через внешнее прерывание;
- опрос периферийных устройств производится за счет включения в алгоритм специальных операторов управления транзакциями, количество которых в алгоритме может быть произвольным;
- выбор ветви продолжения вычислительного процесса в местах ветвления алгоритма является случайным и определяется условиями, включенными в операторы принятия решения и законами распределения обрабатываемых данных;
- время выполнения операторов алгоритма является случайным, функции распределения времени выполнения операторов также определяются законами распределения обрабатываемых данных.

Шинный интерфейс. Шинный интерфейс (рисунок 2) допускает подключение к одному устройству нескольких устройств с помощью общих линий связи (шины) с возможностью взаимодействия в каждый отдельный момент только с одним из подсоединенных устройств [4].

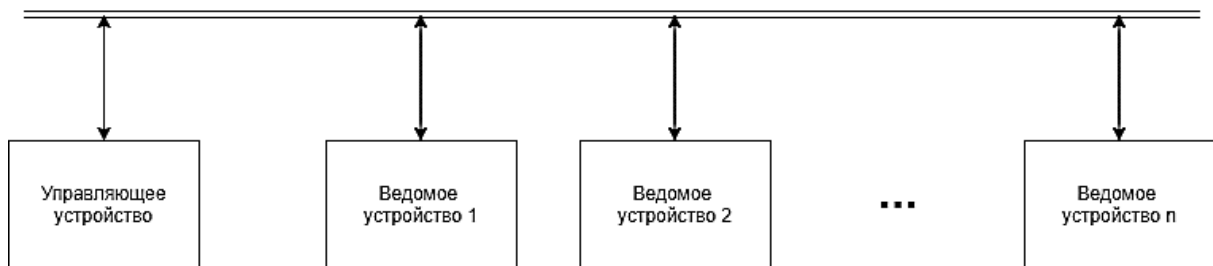


Рисунок 2 – Схема шинного интерфейса

Даже в системах, где ведомые устройства могут самостоятельно инициализировать обмен данными с мастером, от последнего должны осуществляться периодические опросы о работоспособности всех устройств в системе. Поэтому необходимо рассчитать наиболее эффективные периоды для выполнения подобных опросов, и чтобы своевременно узнать об отказе какого-либо устройства, и чтобы не перегружать всю систему лишними запросами.

Кроме того, в системах, использующих для обмена данными протоколы по логике работы схожие с modbus, т.е. протоколы, основанные на архитектуре ведущий-ведомый, в которых все общение инициализируется мастером, это является крайне важной задачей при разработке всей системы.

Параметры, которые необходимо учитывать при расчете частоты опроса системы:

- скорость канала, т.е. скорость передачи данных в данной системе;
- количество ведомых устройств в системе;
- время обработки команды ведомым устройством;
- тактовая частота устройств;
- количество опрашиваемых подустройств (датчиков), т.е. любое из ведомых устройств в системе в свою очередь может иметь свой собственный ряд ведомых устройств, в том числе и каких либо датчиков.

В дальнейшей работе могут быть выявлены и другие параметры, на которые стоит обратить внимание. Основной параметр, который влияет на то, как часто управляющее устройство сможет опрашивать ведомые устройства – это скорость канала связи, которым они соединены. Ведь именно от того как быстро данные передаются от одного устройства к другому во многом определяет быстродействие всей системы.

Стоит отметить, что обращение может идти как к конкретному устройству, так и ко все устройствам в системе сразу (рисунки 3-4).

При передаче данных в шинных интерфейсах зачастую предусматривается идентификация от мастера, какому ведомому устройству предназначается запрос. Выбор ведомого устройства в различных протоколах передачи данных может реализовываться по-разному, это может быть как программная реализация, так и аппаратная.

В случае с протоколом Modbus и ему подобными, идентификация реализуется на программном уровне за счет указания адреса устройства, к которому идет обращение. Однако дополнительно предусматривается и наличие безадресных команд, которые должны обрабатываться каждым устройством.

Если же рассматривать шину SPI, то тут уже идет аппаратный выбор «получателя». В данном интерфейсе используется линия выбора микросхемы или же выбора ведомого.

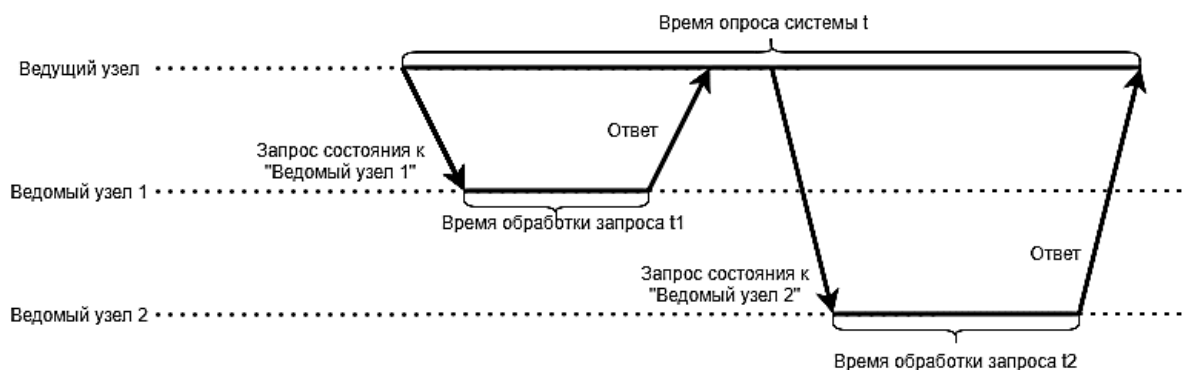


Рисунок 3 – Последовательный опрос устройств

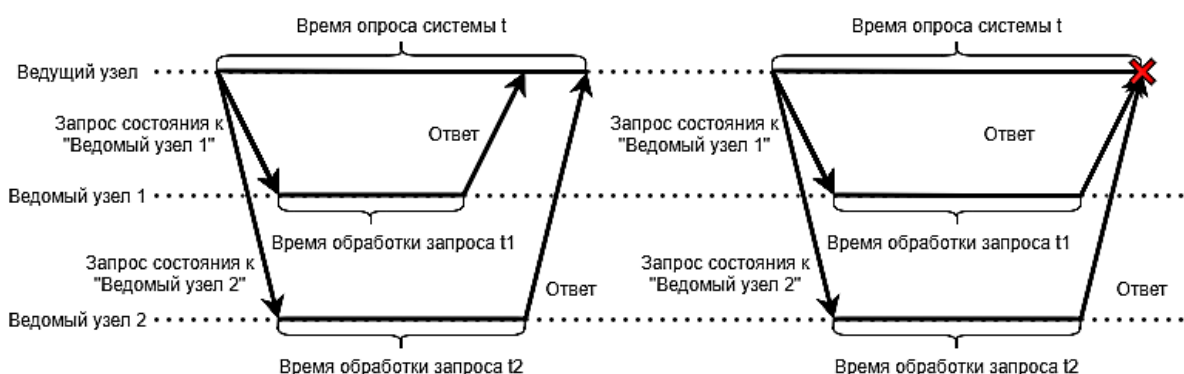


Рисунок 4 – Одновременный опрос устройств

В отличие от передачи ведущий-ведомые, когда запрос данных поступает на все устройства сразу, отвечать по одной линии все устройства одновременно не могут. Т.е. время ответа на запрос будет складываться из времени передачи данных каждым устройством. Однако такой способ опроса данных управляющим устройством используется нечасто. Связано это может быть с тем, что мастер не контролирует порядок поступления данных, а также данный метод может привести к ошибкам в обмене данными всей системы (рисунок 4).

Поэтому чаще всего в таких системах при выполнении опроса используют обращение к конкретному устройству (рисунок 3). Но и данный метод не без минусов, для того чтобы управляющему устройству опросить все ведомые устройства нужно отправить запрос каждому из них, что соответственно увеличивает продолжительность выполнения этого опроса.

Следующим шагом происходит обработка полученного пакета на ведомых устройствах. Так как скорость обработки в разных устройствах может быть разной, то соответственно, отвечать управляющему устройству они будут по мере готовности.

Стоит обратить внимание на такие параметры как тактовая частота, на которой работает устройство, так и набор действий, которые необходимо выполнить для формирования ответа.

После выполнения всех необходимых действий ведомый отправляет ответ мастеру. Как только данный этап проходит каждое ведомое устройство во встраиваемой системе можно считать, что поллинг текущей итерации выполнен.

Выводы. Хотя опрос может быть очень простым как в реализации, так и работе с ним, однако во многих ситуациях (например, в многозадачных операционных системах) зачастую более эффективно использовать прерывания, поскольку это может снизить использование процессора и/или потребление полосы пропускания.

Однако в случае использования в системе шинного способа связи все немного меняется. Так как зачастую в системах с подобной топологией, общение инициализируется ведущим, тогда как все остальные устройства – ведомые, должны ожидать приходящего к ним запроса. Соответственно в таких системах не избежать необходимости разбираться в организации грамотного опроса всех устройств, как для мониторинга работоспособности системы в целом, так и для сбора данных с каждого конкретного узла.

В данной работе не отмечается фактор возможного участия пользователя (оператора), однако его действия могут влиять на логику работы системы, а соответственно должны учитываться при расчете. В дальнейшей работе данный фактор необходимо будет дополнительно рассмотреть.

Опираясь на материалы данной статьи, планируется продолжить исследование в данном направлении. Планируется проанализировать имеющиеся на данный момент исследования на тему определение периодов поллинга во встраиваемых системах. И по полученным материалам разработать свои алгоритмы и модели определения оптимальной периодичности выполнения опросов с учетом различных факторов систем.

1. *Tinotenda Zwavashe, Dr. D Vasumathi, Polling, Interrupts and μ COS-II: A Comparative Timing Response Simulation Model for Wireless Processor-to-Processor Communication, – International Journal of Science and Research (IJSR), Volume 3 Issue 7, 2014, 116 - 122*

2. Аршамян А. А., Ларкин Е. В.. Параметры потока транзакций, генерируемых по принципу поллинга // Известия Тульского государственного университета. Технические науки. – №. 2. – 2016. – С. 40-48.

3. Ларкин Е.В., Ивутин А.Н. Определение временных интервалов в алгоритмах управления // Известия Томского политехнического университета. – № 5. – Т. 324. – 2014. – С. 6-12.

4. Заморин А. П., Мячев А. А., Селиванов Ю. П. Вычислительные машины, системы, комплексы. Справочник. Под ред. Б. Н. Наумова и В. В. Пржиялковского. М., Энергоатомиздат, 1985. – 264 с.