

4. Баскаков С.И. Радиотехнические цепи и сигналы: учеб. Для вузов по спец. «Радиотехника». — М: Высш. шк. 1983. — 536.
5. Голубинский А.Н, Гущина А.А. Математическая модель вокализованных сегментов речевого сигнала, основанная на модели речевого тракта
6. Резонансная теория пения — Википедия. [http://ru.wikipedia.org/wiki/ %D0 %A0 %D0 %B5 %D0 %B7 %D0 %BE %D0 %BD %D0 %B0 %D0 %BD %D1 %81 %D0 %BD %D0 %B0 %D1 %8F_ %D1 %82 %D0 %B5 %D0 %BE %D1 %80 %D0 %B8 %D1 %8F_ %D0 %BF %D0 %B5 %D0 %BD %D0 %B8 %D1 %8F](http://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B7%D0%BE%D0%BD%D0%B0%D0%BD%D1%81%D0%BD%D0%B0%D1%8F_%D1%82%D0%B5%D0%BE%D1%80%D0%B8%D1%8F_%D0%BF%D0%B5%D0%BD%D0%B8%D1%8F)

Проблемы организации распределенной обработки алгоритмов функций логики централизации и методы их решения

Шатковский Олег Юрьевич, кандидат технических наук, технический директор;
Родяков Алексей Юрьевич, руководитель отдела технологического программного обеспечения;
Гоман Александр Александрович, руководитель отдела СПО и САПР
ЗАО «Фортатек АТ» (г. Москва)

Введение

В настоящее время сфера применения многопроцессорных систем (МС) непрерывно расширяется, охватывая все новые области в различных отраслях науки, бизнеса и производства. В данном случае сфера железнодорожного транспорта не является исключением.

На железнодорожном транспорте внедрение многопроцессорных систем во многом обусловлено необходимостью создания единой высоко производительной программно — аппаратной платформы для построения информационно-управляющих систем с возможностью централизации большого количества объектов без ограничений накладываемых географическим распределением этих объектов. Как и во множестве других концептуальных вопросов имеются сторонники и противники создания многопроцессорных систем с распределенной обработкой алгоритмов технологического программного обеспечения, однако как показывает опыт такие системы успешно зарекомендовали себя на железных дорогах всего мира.

В данной статье речь идет о проблемах организации распределенной обработки алгоритмов функций логики централизации и методах их решения.

Синхронизация состояний объектов

В многопроцессорных системах одной из первоочередных задач является синхронизация состояний объектов логики. Управляющие компьютеры централизации (УКЦ) должны максимально быстро сообщать участникам распределенного вычислительного процесса об изменении состояния объекта логики.

Решение этой проблемы достигается путем немедленной передачи этих состояний при изменении состояния объекта логики.

Потеря сообщений синхронизации

В процессе работы многопроцессорной системы есть вероятность потери сообщений (обрыв канала, перепутывание разъемов и т.п.).

Решение этой проблемы достигается путем периодической передачи сообщений синхронизации для каждого объекта логики, участвующего в распределенном вычислительном процессе. Если в заданный период времени сообщение синхронизации не приходит на УКЦ, то объект переходит в безопасное состояние.

Посылка команд оператора

Оператор с автоматизированного рабочего места (АРМ) посылает команды, которые должны быть восприняты одновременно всеми УКЦ.

Решение этой проблемы достигается путем широковещательной посылки команды на все УКЦ. При этом аргумент команды содержит номер объекта управления, для которого эта команда отправлена. В распределенной системе команда воспринимается только тем узлом, в котором работает указанный объект управления. Другие УКЦ принятую команду отвергают.

Общестанционные объекты логики

Логика централизации имеет общестанционные объекты, которые должны продолжать свою работу даже в случае выхода из строя одного или нескольких УКЦ.

Решение этой проблемы достигается путем создания таких объектов в каждом УКЦ. Синхронизация таких объектов осуществляется не с помощью сообщений синхронизации, а по входным воздействиям (условиям перехода), т.е. эти объекты имеют одинаковый список входных привязок объектов, за которыми они наблюдают. При выполнении какого-либо условия, все общестанционные объекты одновременно меняют свое состояние. Следовательно, входные воздействия для таких объектов не должны зависеть от каналов подсистемы ввода/вывода конкретного УКЦ, а только от состояний логических объектов, которые, передаются на все узлы.

Посылка ответственных команд оператора

Оператор с АРМа посылает ответственные команды, которые ожидают подтверждения. В период ожидания ни один УКЦ не должен воспринимать никакие команды,

кроме двух: подтверждение и отмена ответственной команды.

Решение этой проблемы достигается путем создания специальных логических объектов («зона управления») на каждом УКЦ, которые следят за выполнением команд оператора. В случае прихода ответственной команды на УКЦ определяется, что команда пришла объекту, который есть на узле, затем запускается на объекте «зона управления» таймер ожидания подтверждения и блокируется прием команд. Другие вычислительные узлы также принимают эту команду, распознают, что она ответственная, и переводят объект «зона управления» в состояние блокировки приема команд с запуском таймера ожидания. После подтверждения или отмены ответственной команды объект «зона управления» соответственно выполняет или отвергает команду на том узле, где ожидалось подтверждение и разблокирует прием команд. На других УКЦ прием команд также разблокируется. Таким образом, разблокировка приема команд может выполняться или по истечении таймера или по приему команды подтверждения или отмены ответственной команды.

Восстановление номера текущего оператора

В каждом УКЦ многопроцессорной системы запоминается номер оператора, который работает в данный момент с системой в той зоне управления, за которую он отвечает. При выходе из строя одного или нескольких УКЦ, а затем их восстановлении необходимо также автоматически восстанавливать номер текущего оператора.

Решение этой проблемы достигается путем передачи номера оператора от работающих УКЦ всем узлам, которые в данный момент включились в работу. Новый узел при старте посылает запрос на работающие узлы о номере оператора. Каждый работающий узел присылает ему

ответ. Номер оператора устанавливается по последнему присланному ответу.

Одновременное выполнение условий логики

В распределенной системе существует вероятность того, что могут одновременно выполняться несколько логических условий, которые влияют на работу определенного объекта логики. В этом случае объект логики находится в состоянии неопределенности, какому условию следовать.

Решение этой проблемы достигается путем введения приоритетов для логических условий, которые задаются проектировщиком в явном виде. В случае одновременного выполнения нескольких логических условий, выбирается то условие, которое имеет более высокий приоритет.

Для логических условий на множественных входах объекта кроме логических функций И/ИЛИ введена специальная логическая функция ОДИН, которая принимает истинное значение, если выполняется логическое условие только на одном аргументе. Это позволяет отсеять ситуации, при которых несколько логических условий на множественном входе имеют истинное значение, например, одновременное задание двух и более маршрутов на одном участке пути.

Одновременный запуск нескольких алгоритмов

Существует вероятность того, что два или несколько объектов (исполнители) могут одновременно начать выполнение своего алгоритма на основании состояния другого объекта (инициатор).

Для исключения одновременного выполнения алгоритмов двух или нескольких объектов вводится понятие «ключа логического объекта» (аналог маркера). Этот ключ посылается только тому объекту-исполнителю, который первым послал запрос (см. рис. 1). Значение ключа равно номеру объекта-исполнителя. Объект-исполнитель, получив ключ, проверяет его значение и состояние

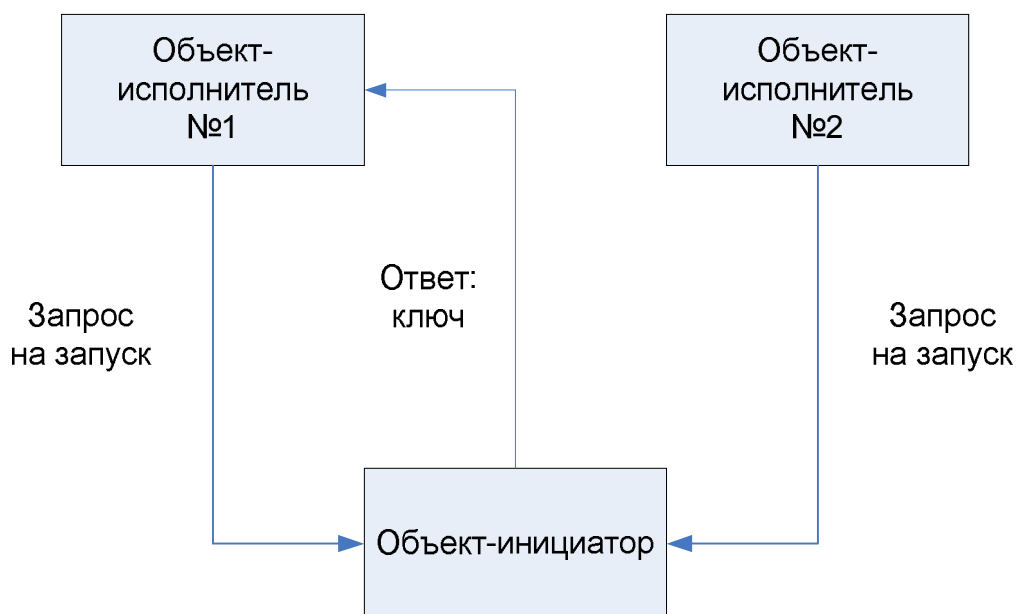


Рис. 1. Схема исключения одновременного запуска двух и более объектов

объекта-инициатора, и затем после успешной проверки запускает свой алгоритм работы. Другой объект-исполнитель не получает ключ и после определенного периода ожидания аннулирует свой запрос.

Заключение

В заключении следует отметить, что описанные, в данной статье, методы решения проблем организации распределенной обработки алгоритмов функций логики

централизации, с успехом применены при создании микропроцессорной системы электрической централизации МПЦ-МЗ-Ф разработки компании «Форатек АТ». Безусловно, представленные положения могут быть использованы при создании автоматизированных многопроцессорных систем применяемых не только на железнодорожном транспорте, но и в ряде других областей промышленности.