

**В.В. Смирнов, канд. техн. наук, Н.В. Смирнова, асп.**  
*Кировоградский национальный технический университет*

## Аппаратная реализация параллельного выполнения потоков в автоматизированной системе управления дозаторами

В статье изложен принцип построения аппаратной реализации параллельного выполнения потоков (threads) на базе микроконтроллеров общего назначения в автоматизированной системе управления дозаторами.

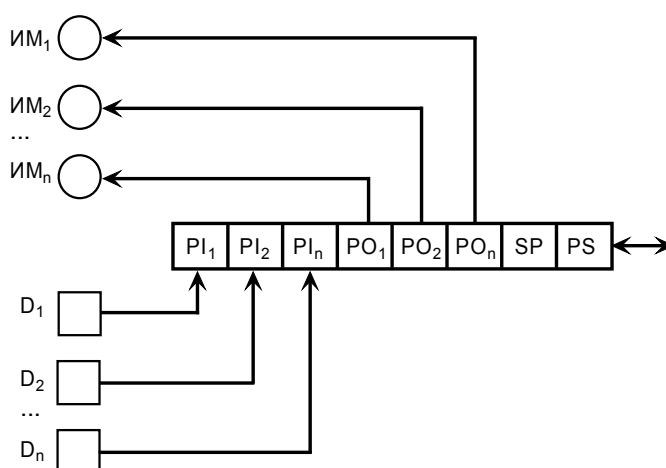
**микропроцессорная техника, передача данных, системы управления, локальные сети**

В настоящее время в системах автоматизированного управления технологическими процессами используются процессоры различной производительности. В частности, в системах управления дозаторами находят применения микроконтроллеры общего применения, имеющие относительно невысокие показатели производительности.

Как правило, архитектура системы управления является стандартной, то есть, центральный процессор выполняет все математические вычисления и выполняет управление периферийными устройствами. Основным недостатком такой архитектуры является последовательное выполнение процессов, каждый из которых реализуется соответствующим программным модулем. Отсюда следует, что выполнение следующего процесса не начнется, пока не завершится выполнение текущего процесса.

Проблема состоит в том, что такие системы, при определенных условиях, могут стать неэффективными, поскольку обработка медленнотекущего процесса занимает все процессорное время и система перестает выполнять функции управления объектом.

На рисунке 1 представлен пример последовательного выполнения процессов в стандартной реализации системы управления дозаторами.



$D_1..D_n$  – датчики,  $ИМ_1..ИМ_n$  – исполнительные механизмы,  $PI_1..PI_n$  – процессы обслуживания датчиков,  $PO_1..PO_n$  – процессы обслуживания исполнительных механизмов,  $SP$  – сервисный процесс,  $PS$  – коммуникативный процесс

Рисунок 1 – Последовательное выполнение процессов в системе управления

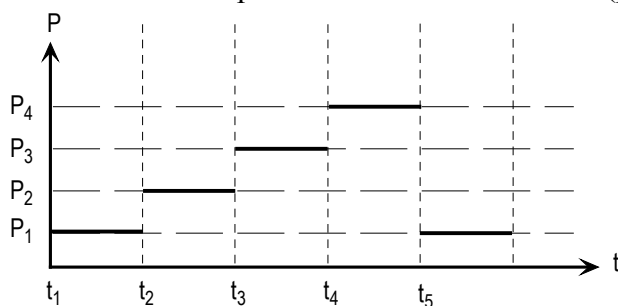
То есть, в каждый конкретный момент времени выполняется процесс 1 из  $N$ , в то время, как остальные процессы ожидают своей активизации в соответствии с заданным алгоритмом.

Частично проблема надежности системы решается путем реализации механизма прерываний, однако, многие процессы требуют непрерывной обработки, что значительно снижает общую эффективность и надежность системы управления.

Альтернативой механизму прерываний является использование потоков, функционирующих в рамках одной задачи [2]. В этом случае, каждому потоку выделяется фиксированный квант времени и ресурсы процессора в рамках контекста выполнения процесса. Реализация многопоточности требует наличия:

- операционной системы;
- мощного процессора;
- достаточного объема оперативной памяти для хранения контекстов выполнения параллельных потоков;
- среды выполнения многопоточных задач.

Одновременное выполнение нескольких потоков при наличии в системе одного процессора фактически не является их параллельным выполнением (рис. 2).



$P_1 \dots P_4$  – выполняемые потоки;  $t_1 \dots t_5$  – моменты переключения потоков

Рисунок 2 – Выполнение потоков в среде выполнения

При использовании микроконтроллеров общего назначения ограниченные ресурсы микроконтроллера значительно затрудняют создание высокоэффективной системы управления на основе многопоточности.

Существующие процессоры потоковой обработки (Stream Processing Engines, SPE) специально разработаны для работы с потоковыми данными [3], но они имеют высокую стоимость и не применяются в компактных системах управления.

Решение проблемы создания системы управления на основе параллельного выполнения потоков заключается в использовании многопроцессорной архитектуры, сконфигурированной для параллельной обработки входных данных, параллельного управления исполнительными механизмами и параллельного выполнения сервисных и коммуникативных процессов.

В этой архитектуре каждый процессор выполняет только свой поток (процесс), поэтому имеет в своем распоряжении все ресурсы контроллера: процессорное время, оперативную память, таблицы констант и коэффициентов, то есть весь контекст выполнения процесса. Это значительно повышает общую скорость выполнения программы, поскольку операции с общей памятью существенно увеличивают задержки процесса [4].

Такая архитектура не требует специализированной операционной системы, является достаточно гибкой и перестраивается для выполнения различных задач. От степени сложности конкретной задачи зависит количество потоков выполнения, а следовательно, и количество микроконтроллеров в системе управления.

Проблема синхронизации параллельных потоков, возникающая при параллельной обработке, традиционно решается исполняющей средой путем использования семафоров. Эта проблема решается автоматически в микроконтроллерной системе, поскольку взаимодействие между микроконтроллерами осуществляется посредством высокоскоростной (до 1 Мбит/сек) последовательной шиной I<sup>2</sup>C, которая на аппаратном уровне осуществляет арбитраж доступа микроконтроллеров к общим ресурсам системы, что значительно упрощает архитектуру системы управления и ее практическую реализацию (рис. 3). Наличие шины I<sup>2</sup>C позволяет организовать бездресные асинхронные вычисления [1].

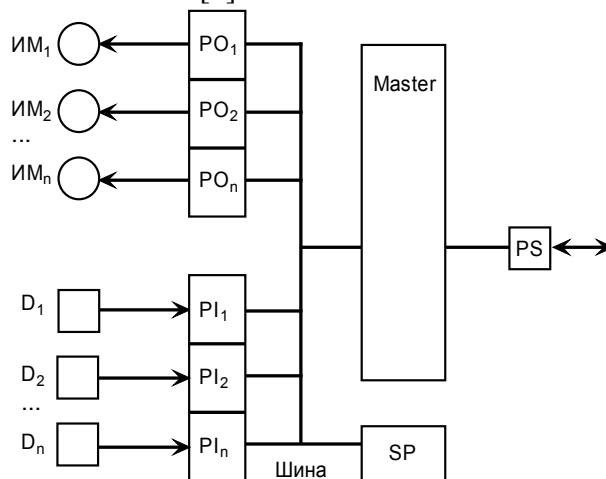
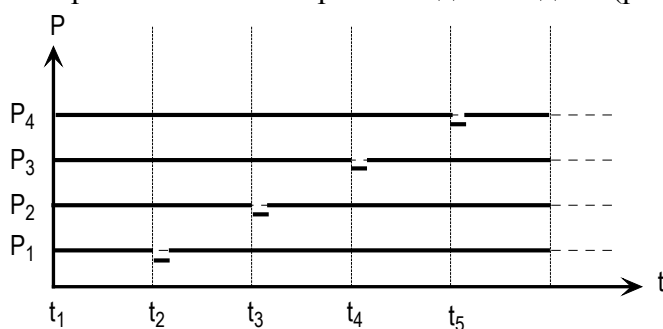


Рисунок 3 – Параллельное выполнение потоков в системе управления

Задачей контроллера Master является мониторинг параллельно работающих контроллеров, установка коэффициентов и параметров вычислений, принятие решений и взаимодействие с контроллером верхнего уровня.

При использовании параллельной архитектуры существенно возрастает скорость обработки входных данных и эффективность работы системы в целом за счет непрерывной параллельной работы потоков в рамках одной задачи (рис. 4).



$P_1 \dots P_4$  – выполняемые потоки;  $t_1 \dots t_5$  – доступ потоков к общим данным

Рисунок 4 – Параллельная работа потоков в среде выполнения

Скорость работы системы в общем случае будет определяться количеством параллельных процессов (потоков) выполняющихся в системе. Предположим, что в управляющей программе доля операций, которые нужно выполнять последовательно, равна  $f$ , где  $0 \leq f \leq 1$ . Крайние случаи в значениях  $f$  соответствуют полностью параллельным ( $f = 0$ ) и полностью последовательным ( $f = 1$ ) управляющим программам.

Оценка ускорения работы системы с параллельной обработкой по сравнению с последовательной может быть получена на основании закона Амдала в соответствии с выражением (1):

$$S \leq \frac{1}{f + (1-f)/p}, \quad (1)$$

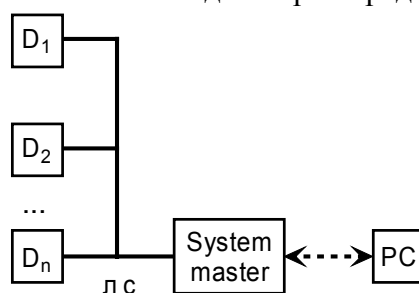
где  $S$  – ускорение работы системы;

$p$  – количество процессоров;

$f$  – количество параллельно выполняемых процессов.

Для построения системы управления для группы объектов целесообразно использовать аналогичную архитектуру. Различия в реализации касаются только подсистемы коммуникации объектов управления низкого уровня и Мастера системы.

Поскольку Мастер системы может находиться на большом удалении от низкоуровневых объектов (дозаторов), то вместо интерфейса  $I^2C$  может быть использован CAN – интерфейс (Ethernet) или другая локальная сеть. Общая архитектура системы управления комплексом дозаторов представлена на рисунке 5.



$D_1..D_n$  – дозаторы, PC – компьютер пользователя

Рисунок 5 – Архитектура системы управления комплексом дозаторов

SystemMaster является основным процессором верхнего уровня. Он решает все задачи связанные с управлением всей системой в целом. Компьютер пользователя является опциональным компонентом и предназначен для визуализации состояния текущего процесса, задания начальных установок и проведения тестирования.

Таким образом, аппаратная реализация выполнения параллельных потоков на базе недорогих (\$3 .. \$5) микроконтроллеров позволяет существенно повысить качественные характеристики системы управления комплексом дозаторов.

## Список литературы

1. В.Коржов. Асинхронные вычисления, или компьютер без процессора.- “Открытые системы”, № 6-1997.- 194с.
2. Таненбаум Э. Современные операционные системы. – Спб.: Питер, 2004. – 1040 с.
3. A. Arasu, B. Babcock, STREAM: The Stanford Stream Data Manager. In ACM SIGMOD Conference, June 2003.
4. M.Stonebraker, U.Getintemel, The 8 Requirements of Real-Time Stream Processing, SIGMOD Record, Vol. 34, No. 4, Dec. 2005

У статті викладений принцип побудови апаратної реалізації рівнобіжного виконання потоків (threads) на базі мікроконтролерів загального призначення в автоматизованій системі керування дозаторами.

In clause the principle construction of parallel performance hardware threads realization for the batchers automated control system on the basis general purpose microcontrollers with parallel processing is stated.