

СПОСОБИ РЕАЛІЗАЦІЇ АЛГОРИТМІВ ЦОС

1. Реальний час ЦОС
2. Апаратна реалізація ЦОС
3. Програмна реалізація ЦОС
4. Апаратно-програмна реалізація ЦОС

Серед алгоритмів ЦОС з погляду організації обчислень можна виділити як прості, так і надзвичайно складні. Проте, незалежно від складності алгоритму обчислення здійснюються за допомогою базових операцій складання, віднімання і множення. Зведення в ступінь — це багатократне множення, а ділення — багатократне віднімання. Оскільки обчислювальні операції проводяться з даними, що затримуються щодо один одного на один і більш періодів дискретизації T за допомогою елементів затримки, що є регістрами (елементи пам'яті), що об'єднуються в лінії затримки, необхідно мати можливість здійснювати пересилки і зміщення даних. Крім того, для управління обчислювальним процесом необхідно передбачити і логічні операції.

Реальний час ЦОС

Чи досить всього перерахованого для побудови алгоритму? Виявляється, достатньо. У математичній логіці доводиться, що алгоритм будь-якої складності може бути побудований і обчислений з використанням кінцевого числа тільки простих математичних і логічних операцій, а також операцій зрушення і пересилки, причому обчислення одного відліку вихідної послідовності здійснюється за обмежене число кроків. Сама процедура може або продовжуватися виконуватися, повторюючись і ніколи не зупиняючись, або з якої-небудь причини урватися на якому-небудь кроці, припинити поточні обчислення і звернутися до іншого алгоритму. Таке звернення називають *перериванням*, а сигнал, що викликає переривання, називається *запитом на переривання*.

Реальний час ЦОС

Принципова можливість обчислення того або іншого алгоритму зовсім не означає, що обмежена кількість кроків завжди задовольнить практичним потребам. Найважливішим чинником, що визначає придатність створеної процедури, є час обчислення одного відліку.

Дійсно, нехай один відлік $y(n)$ формується за 0,1 с. Тоді, якщо це відлік мовного сигналу, то процедура, будучи математично вірною, практично абсолютно даремна; якщо ж це — відліки сигналу, що поступає від датчика температури повітря, що змінюється дуже поволі, використовувана процедура з лишком задовольнить найвимогливішого метеоролога. Таким чином, визначальною властивістю процедури стає її практична обчислюваність, тобто її здатність обчислювати відлік $y(n)$ за розумний час, або, як прийнято говорити, за *реальний час*, при цьому мається на увазі обов'язкове досягнення заданої точності.

Реальний час ЦОС

Визначення реального часу залежить від конкретного завдання і пов'язано з об'ємом обчислень алгоритму, точністю обчислень і частотою дискретизації (періоду дискретизації).

Нехай T — період дискретизації (рис.1), τ_a — час виконання алгоритму.

Визначення. Говорять, що цифрова система працює у реальному часі, якщо час виконання алгоритму τ_a не перевищує періоду дискретизації T .

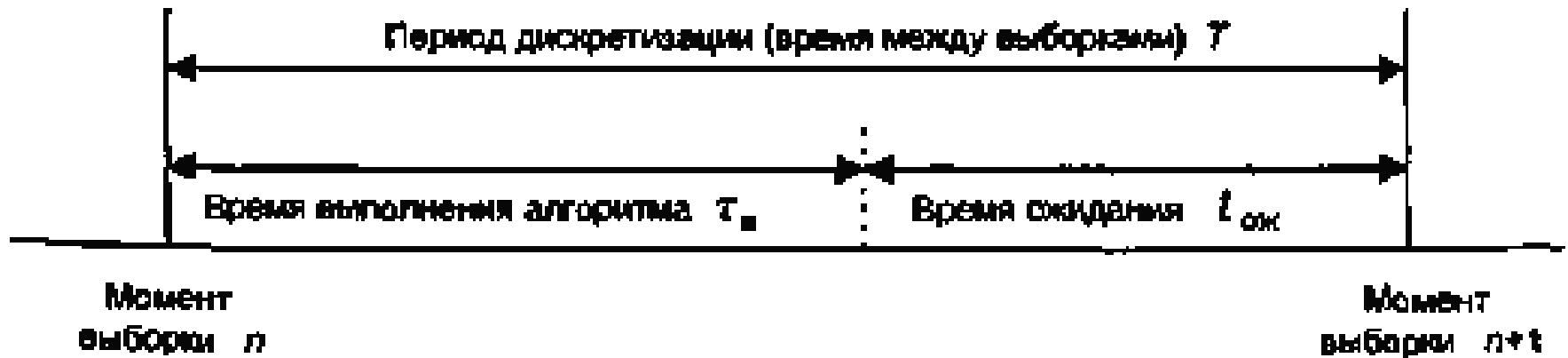


Рис.1

Реальний час ЦОС

Це означає, що залишається ще деякий запас часу, звичайно званий *часом очікування* $t_{ож}$. Знайти час виконання алгоритму можна, якщо знати час виконання елементарної (одноцикловий) команди, який називають *командним циклом*, і кількість командних циклів N_a , необхідних для виконання алгоритму (це можна визначити в процесі *відладки*). Тоді

$$\tau_a = \tau_k N_a, \quad t_{ож} = T - \tau_a$$

Тактова частота показує, як швидко процесор виконує просту одиницю роботи, наприклад пересилку в регістрі з розряду в розряд. Тактова частота повинна істотно перевищувати частоту дискретизації. Відношення тактової частоти до частоти дискретизації відноситься до найбільш важливих характеристик, що визначають, яким чином буде реалізована система. Це відношення частково визначає кількість апаратних засобів, необхідне для реалізації алгоритму заданої складності у реальному часі.

Реальний час ЦОС

Якщо відношення вказаних частот падає, то кількість і складність апаратних засобів, потрібних для реалізації алгоритму, збільшується.

Наприклад, вокодери, будучи надзвичайно складними пристроями, при частоті дискретизації 8 кГц можуть бути реалізовані тільки на декількох процесорах молодшого покоління TMS320C10 (тактова частота 6 МГц, командний цикл 200 нс, відношення частот 750) і всього на одному (!) сучасному процесорі TMS320C67xx (тактова частота 167 МГц, командний цикл 1 нс, відношення частот 20 875) можна реалізувати декілька вокодерів.

Час виконання алгоритму залежить не тільки від процедури, що представляє цей алгоритм, але і від способу реалізації алгоритму. Можливі три способи реалізації алгоритмів ЦОС (рис.2):

- апаратний;
- програмний;
- апаратно-програмний.

Реальный час ЦОС

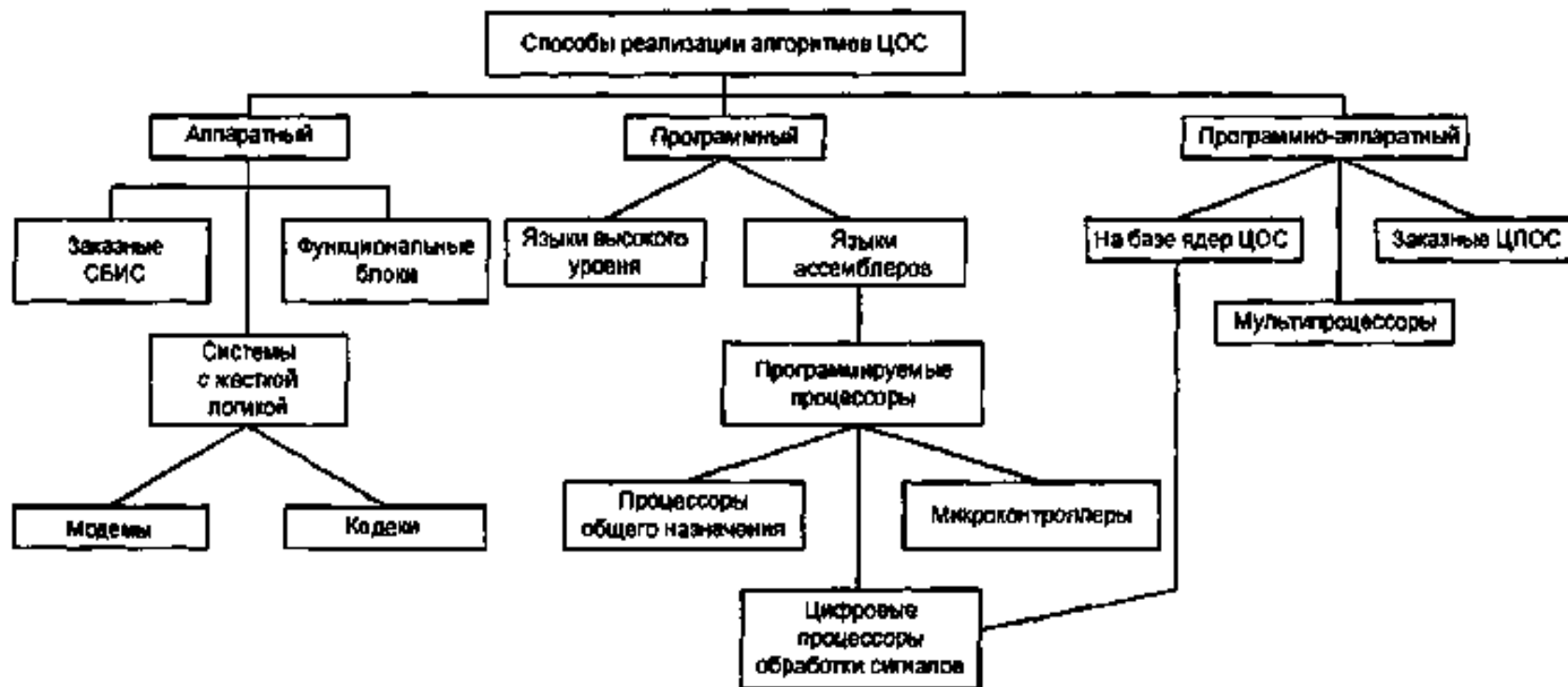


Рис.2

Апаратна реалізація ЦОС

Апаратна реалізація має на увазі використання різноманітних функціональних блоків: регістрів, суматорів, шифраторів і дешифраторів, лічильників, ліній затримок, пристроїв пам'яті, помножувачів, логічних елементів, що реалізуються на основі інтегральних і великих інтегральних схем, програмованих логічних матриць і т.п. Сукупність функціональних блоків і зв'язків між ними визначає алгоритм, що реалізовується.

Розглянемо приклад апаратної реалізації БІХ-ланки другого порядку, що описується різницевим рівнянням:

$$y(n) = b_0x(n) + b_1x(n-1) + b_2x(n-2) + a_1y(n-1) + a_2y(n-2)$$

Апаратна реалізація ЦОС

Обчислення по цьому рівнянню можна організувати декількома алгоритмами, один з яких, за так званою *прямою формою*, у вигляді структурної схеми зображений на рис.3.

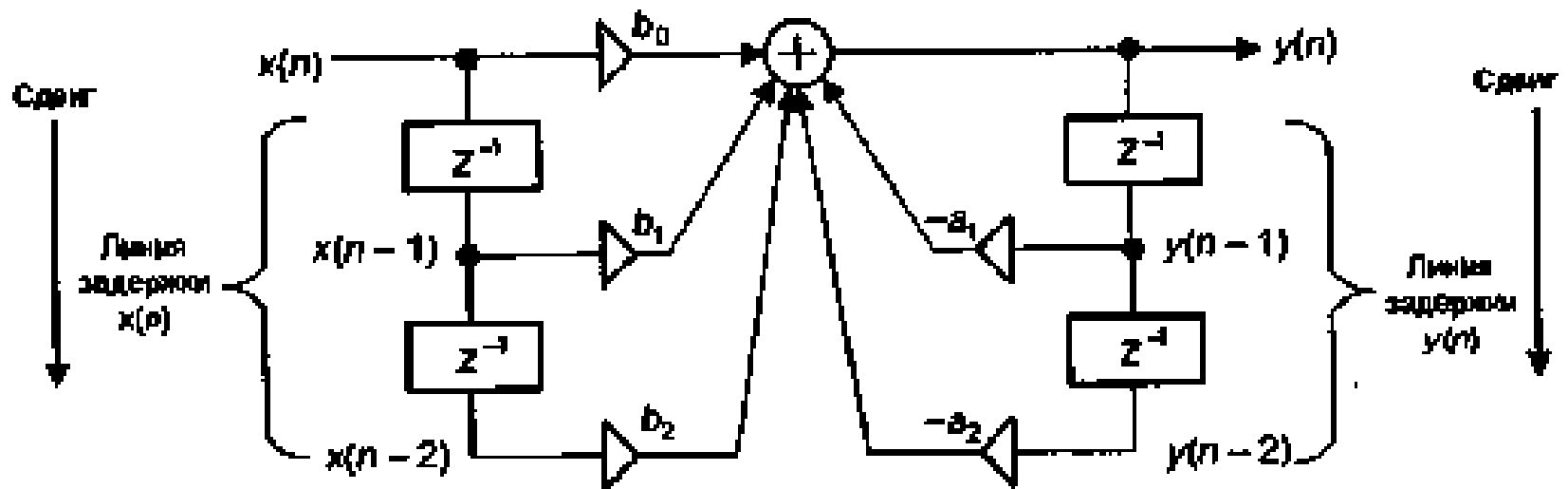


Рис.3

Апаратна реалізація ЦОС

З рівняння і структурної схеми виходить, що цифровий пристрій, що реалізовує даний алгоритм, повинен мати:

- пам'ять коефіцієнтів і пам'ять початкових і проміжних даних;
- систему введення і виведення початкових даних $x(n)$ і результату $y(n)$ відповідно;
- помножувачі;
- суматор;
- генератор тактової частоти (ГТЧ) з періодом проходження імпульсів $\tau_{ТИ} \ll T$.

Такий цифровий пристрій зображений на рис.4.

Апаратна реалізація ЦОС

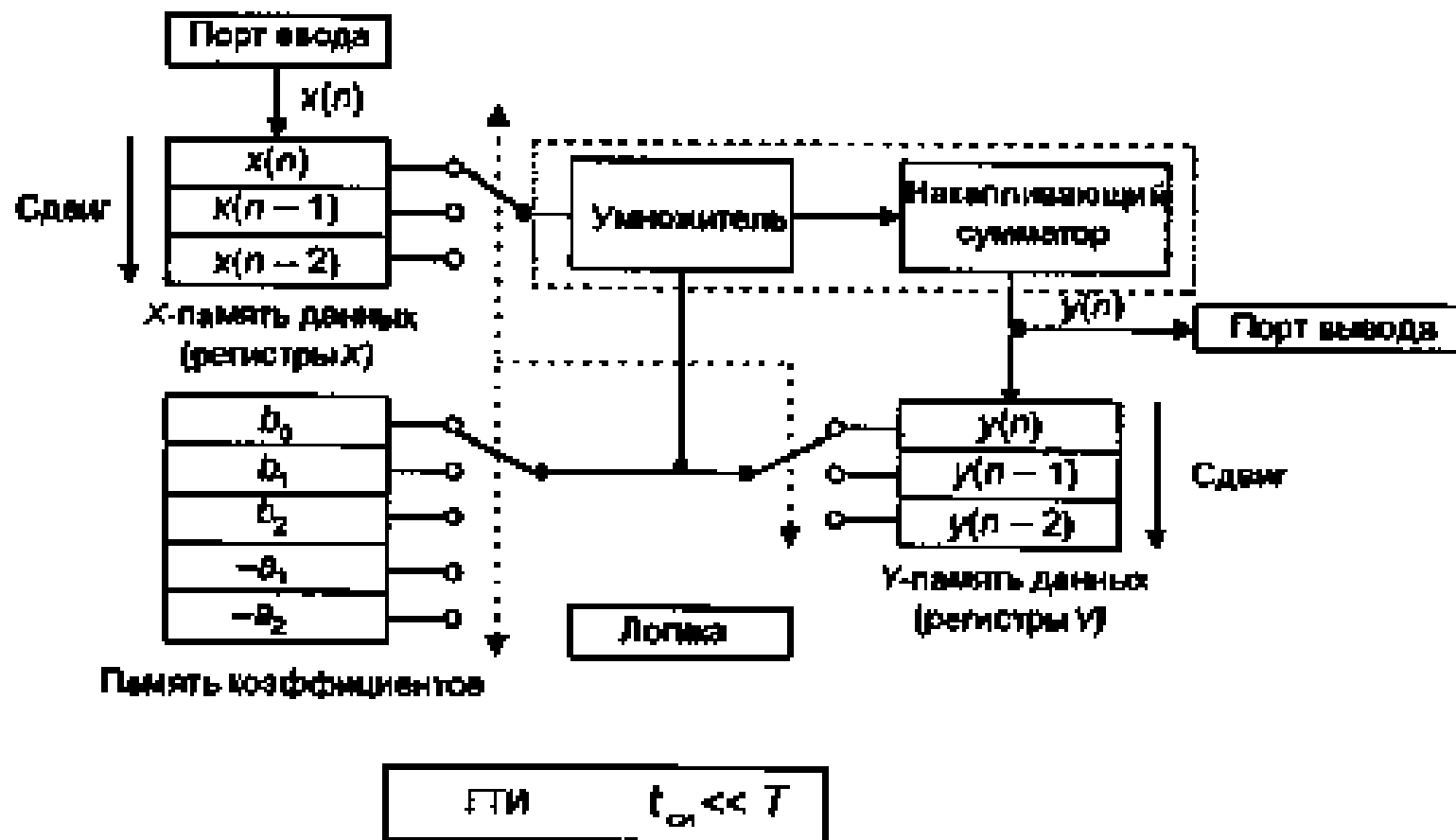


Рис.4

Апаратна реалізація ЦОС

Замість декількох помножувачів використовується один швидкодіючий помножувач, на один вхід якого послідовно згідно логіці обчислень поступають дані $x(n)$ або $y(n)$, на іншій — відповідні коефіцієнти. Отримувані добутки подаються в накопичуючий суматор, результат з якого зчитується один раз за період дискретизації T .

Замість ліній затримок звичайно організують *X-пам'ять* і *Y-пам'ять* даних, а також пам'ять коефіцієнтів. Після закінчення обчислень чергового відліку $y(n)$ здійснюється його пересилка в *Y-пам'ять*, звідки подається на пристрій виведення. У *X-пам'яті* і *Y-пам'яті* відбувається зміщення даних, що означає готовність пристрою до прийому чергового відліку $x(n)$.

Відповідальним за послідовністю виконання всіх операцій є логічний пристрій. Звичайно помножувач, суматор і логічний пристрій об'єднуються в один блок, так званий арифметико-логічний пристрій (АЛП). Зв'язки між всіма блоками здійснюються за допомогою шин. розрядність яких відповідає розрядності комірок *X-пам'яті* і *Y-пам'яті*.

Апаратна реалізація ЦОС

Достоїнство апаратної реалізації полягає в дуже високій швидкодії, що дозволяє обробляти сигнали при частоті дискретизації в десятки мегагерц. Це досягається застосуванням функціональних блоків на базі ТТ-логіки, розпаралелюванням операцій і вузькою спрямованістю (спеціалізацією) створюваних пристроїв (наприклад, для реалізації алгоритму БПФ в системах радіолокації).

З іншого боку, апаратна реалізація, орієнтована на рішення вузькоспеціальних задач, має на увазі створення систем з жорсткою логікою, коли будь-яка зміна алгоритму вимагає зміни структури пристрою, тобто введення додаткових функціональних блоків, що, звичайно, є недоліком. Все це разом визначає високу вартість апаратної реалізації, причому проектування, виготовлення і наладка виявляються вельми трудомісткими при великих часових затратах.

Програмна реалізація ЦОС

Програмна реалізація має на увазі представлення алгоритму у вигляді програми, яку послідовно від команди до команди виконує один або одночасно декілька незалежних блоків. Програма повинна бути написана на мові програмування, відповідній конкретному операційному блоку. Так, для персонального комп'ютера це буде будь-якою з мов високо рівня (Pascal C++, Java і ін.), а для мікропроцесорного комплекту або цифрового процесора — відповідна мова асемблера. Звичайно, всі команди, що становлять програму, повинні бути представлені у вигляді, зрозумілому процесору і безпосередньо ним сприйманому. Команди пред'являються процесору як комбінація нулів і одиниць; такі комбінації складають машинну мову.

Наприклад, в процесорах сімейства TMS320Cxxx команді складання відповідає комбінація 0000, команді множення — комбінація 0011.

Програмна реалізація ЦОС

Написання програми на машинній мові справа у край трудомістка. Тому створюються спеціальні засоби, що полегшують підготовку і налагодження програм для процесора, коли кожній машинній команді або групі команд ставиться у відповідність зрозумілий людині символ на основі мнемоніки (правил і прийомів, що полегшують запам'ятовування) так, щоб символ відображав смисловий зміст команди.

Наприклад:

- add — скласти (від англ. *add*);
- sub — відняти (від англ. *subtract*);
- mul — помножити (від англ. *multiply*).

Мова програмування, в якій кожній машинній команді або сукупності машинних команд відповідає скорочений символічний запис, називається мовою *асемблера*.

Програмна реалізація ЦОС

Переклад програми на машинну мову називається *трансляцією* і виконується автоматично за допомогою спеціальної програми — *асемблера* (від англ. *assembler* — складальник).

Розроблені і всього більшого значення набувають спеціальні ефективні програми-транслятори з мов високого рівня на мови асемблерів. Ці програми називаються *компіляторами*.

До достоїнств програмної реалізації відносяться:

- незмінна структура системи при різних алгоритмах і областях застосування;
- гнучкість, що дозволяє достатньо легко змінювати алгоритми роботи системи за рахунок корекції або зміни програми;
- істотне прискорення, полегшення і здешевлення проектування, виготовлення і налагодження системи, оскільки замість приладу розробляється програма.

Програмна реалізація ЦОС

Недоліком програмної реалізації є відносно низька швидкодія унаслідок послідовного виконання операцій програми в одному процесорі: як би не збільшували швидкість виконання команд, вона залишатиметься нижче продуктивності відповідного пристрою, реалізованого апаратно. Звідси витікає завдання забезпечення реального часу, яке має на увазі дві обставини:

- по-перше, час обробки одного відліку або групи відліків сигналу $t_{об}$ повинен бути менше допустимого часу затримки $t_{дон}$; контроль за виконанням цієї умови здійснюється як при написанні програми, так і при її налагодженні;
- по-друге, цикл роботи програми і моменти надходження відліків вхідного сигналу $x(n)$ повинні бути строго узгоджені за часом, тобто початок обробки чергового відліку $x(n)$ повинен співпадати або слідувати за надходженням цього відліку; з іншого боку, результат обробки $y(n)$ повинен бути виведений згідно темпу роботи зовнішнього пристрою, який не зобов'язаний співпадати з темпом надходження відліків $x(n)$.

Апаратно-програмна реалізація ЦОС

Апаратно-програмна реалізація має на увазі, що частина функцій системи ЦОС виконується апаратно (аналого-цифрове і цифро-аналогове перетворення, множення, множення з накопиченням, прийом/передача даних і ін.), а інша частина функцій виконується програмно. Приклад апаратно-програмної реалізації показаний на рис.5.

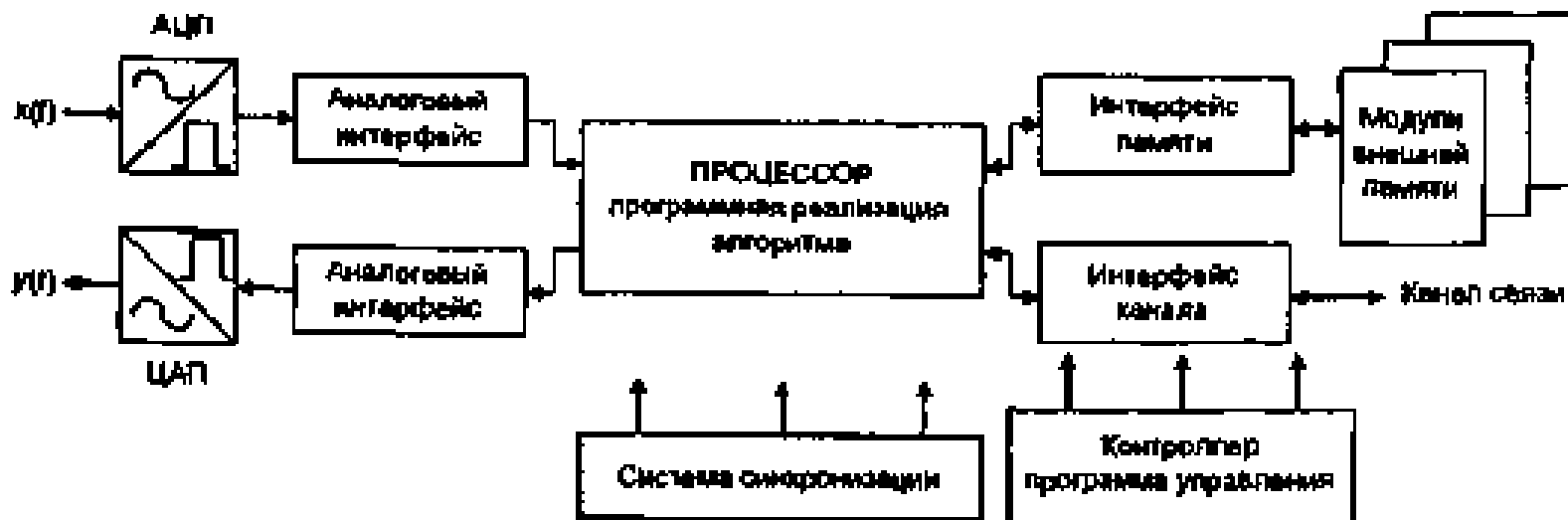


Рис.5

Апаратно-програмна реалізація ЦОС

До процесора, що працює за заданою програмою, підключені:

- аналого-цифровий (АЦП) і цифро-аналоговий (ЦАП) перетворювачі;
- модулі зовнішньої пам'яті, що зберігають програми, різноманітні константи і таблиці функцій (наприклад, \sin і \cos), що дозволяє замінювати тривале їх обчислення швидким зверненням до пам'яті;
- інтерфейс — спеціальні допоміжні схеми, що забезпечують узгодження сигналів на стиках з модулями зовнішньої пам'яті, АЦП, ЦАП, каналом зв'язку (реєстри стиковки також можуть бути віднесені до інтерфейсу);
- система синхронізації, що забезпечує часове узгодження всіх елементів системи.

Апаратно-програмна реалізація ЦОС

Апаратно-програмна реалізація поєднує позитивні властивості апаратної і програмної реалізацій. Розумне поєднання апаратних і програмних засобів дозволяє понизити вимоги до обчислювальних можливостей елементної бази і спростити реалізацію систем ЦОС в цілому, для налагодження якої потрібні спеціальні засоби налагодження. Їх обов'язковість не є недоліком: засоби налагодження створюються під конкретну елементну базу і по суті є *інструментом розробки* численних систем ЦОС на цій елементній базі.