

Висновки. Одержані після проведених досліджень результати параметричного синтезу число-імпульсної структури експоненційного перетворювача дозволили отримати за допомогою такої структури високоточну розгортку експоненційної функції (не тільки з від'ємним, але й з додатним показником степеня) в широкому діапазоні зміни аргументу. Розроблено також ефективні автоматизовані методи параметричного синтезу експоненційних перетворювачів різної розрядності і способи розширення діапазону роботи експоненційних перетворювачів. Високі характеристики розроблених структур дозволяють використовувати їх як базові в автоматизованій методиці синтезу цифрових інтегруючих структур.

1. Горпенюк А.Я. *Реверсивні число-імпульсні функціональні перетворювачі: Автореф. дис. ... канд. техн. наук. Львів, 1998.* 2. Шеннон К. *Работы по теории информации и кибернетике: Пер. с англ. М., 1963.* 3. Лагун А.Е. *Питання структурного синтезу число-імпульсних функціональних перетворювачів кодів. // Вісник Державного університету "Львівська політехніка" 1991. № 257. С. 43-46.* 4. Дудикевич В.Б. *Число-імпульсні функціональні перетворювачі: Автореф. дис. ... д-ра техн. наук. Львів, 1991.*

УДК 681.32

ПРОГРАМНО-АПАРАТНА РЕАЛІЗАЦІЯ ПРОЦЕСОРІВ ЦИФРОВИХ СИСТЕМ З МІНІМІЗАЦІЄЮ ЗАТРИМКИ ОБЧИСЛЕНЬ

© Стрепко І.Т., Тимченко О.В., 2000

Українська академія друкарства, кафедра АКТ

На основі різницевих методів обробки сигналів запропоновано алгоритми та структурні схеми процесорів цифрових систем реального часу з мінімізацією затримки обчислень.

On the basis a differencely of methods processing of signals the algorithms and block diagrams of processors of digital systems of real time with minimization of a delay of calculations are developed.

У цифрових системах реального часу, тобто системах, в яких час затримки обчислень не перевищує періоду дискретизації сигналу, необхідно мінімізувати час, протягом якого формується вихідний сигнал процесора. У разі послідовного виконання всіх необхідних операцій (додавання і множення) вихідний сигнал процесора отримують лише наприкінці періоду дискретизації. Уважний розгляд структури різницевих рівнянь, якими описуються цифрові системи, показує, що велика кількість операцій може виконуватися не в даному періоді дискретності, а завчасно, скоротивши тим самим затримку на формування вихідного сигналу.

Додатково скоротити згадану затримку можна за рахунок спрощення і відповідно пришвидшення виконання операцій перемноження та підсумовування. Одними з таких методів є запропоновані в [1] різницеві методи обробки, однак алгоритми мінімізації затримки обчислень висвітлені там недостатньо. Запропоновані у [2] методи не мають відповідного апаратного втілення, крім того, в них не оцінені реальні витрати часу та обсяги пам'яті процесорів, реалізованих у розподілених системах керування [3].

Використовуючи методи, запропоновані в [1,2], запишемо рекурсію, тобто алгоритм функціонування лінійної системи у вигляді

$$y_n = \sum_{m=0}^{M-1} x_{n-m} h_m - \sum_{m=1}^{M-1} y_{n-m} g_m,$$

де $\{x_k\}, k > 0$ – вхідний сигнал, $\{h_m\}, m = \overline{0, M-1}$, $\{g_m\}, m = \overline{1, M-1}$ – нерекурсивна та рекурсивна частина імпульсної характеристики (ІХ) реалізовуваної системи. У випадку однорозрядного різницевого подання сигналів отримаємо [1]:

$$y_n = \sum_{i=1}^n \sum_{k=1}^i \left(\sum_{m=0}^{M-1} e_{r-m}^{(x)} e_m^{(h)} - \sum_{m=1}^{M-1} \nabla^2 y_{k-m} e_m^{(g)} \right). \quad (1)$$

Тут $\{e_r^{(\cdot)}\}$ – однорозрядні кроки квантування відповідних сигналів, $\{\nabla^2 y_r\}$ – послідовність других різниць вихідного сигналу $\{y_n\}$, причому $\nabla^2 y_r \equiv 0, \forall r \leq 0$.

Найпростішим видом однорозрядного різницевого кодування є лінійна дельта-модуляція (ЛДМ), застосування якої дозволяє максимально скоротити операції перемноження і підсумовування [1]. Вихідний сигнал кодера ЛДМ формується за правилом

$$e_k^{(x)} = \begin{cases} \varepsilon^{(x)}, \alpha_k^{(x)} = x_k - \hat{x}_k \geq 0; \\ -\varepsilon^{(x)}, \alpha_k^{(x)} < 0, \end{cases} \quad (2)$$

де $\{\hat{x}_k\}$ – апроксимуючий вхідний сигнал.

Аналогічно до (2) формуються послідовності $\{e_m^{(h)}\}$ та $\{e_m^{(g)}\}$ через визначення знака похибки апроксимації $\alpha_m^{(\cdot)}$ для сигналу (\cdot) .

Застосуємо нормування кроків (2) у вигляді

$$B_{1,k}^{(\cdot)} = (e_k^{(\cdot)} + \varepsilon^{(\cdot)}) / 2\varepsilon^{(\cdot)}, \quad (3)$$

Тоді замість операцій з кроками $e_k^{(\cdot)} \in \{-\varepsilon^{(\cdot)}, \varepsilon^{(\cdot)}\}$ переходимо до операцій з відповідними бітами $B_{1,k}^{(\cdot)} \in \{0,1\}$. Отже, вагова послідовність подана форматом ЛДМ, тобто однорозрядним кодом $\{B_{1,m}^{(h)}\}, m = \overline{0, M-1}$ та $\{B_{1,m}^{(g)}\}, m = \overline{1, M-1}$. У цьому випадку перемноження стає однорозрядним $\{e_m^{(h)} e_k^{(x)}\}$, істотно спрощується і його доцільно виконати за допомогою елемента Виключне АБО за формулою

$$e_m^{(h)} \cdot e_k^{(x)} = 2 \left(B_{1,m}^{(h)} \oplus B_{1,k}^{(x)} \right) - 1, \quad (4)$$

а перемноження $\{e_m^{(g)} \nabla^2 y_k\}$ вироджується в операцію підсумовування за модулем два знакового розряду другої різниці $B_k^{(\nabla^2)0}$, поданої прямим кодом, та біта вагової послідовності $B_{l,m}^{(g)}$ у вигляді

$$\left[2 \left(B_k^{(\nabla^2)0} \oplus B_{l,m}^{(g)} \right) - 1 \right] \cdot |\nabla^2 y_k|.$$

Обчислення за формулою (1) найраціональніше здійснювати у три етапи, в кожному з яких виконується операція підсумовування

$$\nabla^2 y_k = \sum_{m=0}^{M-1} e_{k-m}^{(x)} e_m^{(h)} + \sum_{m=1}^{M-1} \nabla^2 y_{k-m} e_m^{(g)} = S_k + R_k; \quad (5)$$

$$\nabla y_i = \sum_{k=1}^i \nabla^2 y_k = \nabla y_{i-1} + \nabla^2 y_i; \quad (6)$$

$$y_r = \sum_{i=1}^n \nabla y_i = y_{n-1} + \nabla y_n, \quad (7)$$

де перша різниця вихідного сигналу $\nabla y_r \equiv 0, r \leq 0$, а $e_{M-1}^{(g)} \neq 0$.

Слід зазначити, що в (1) у n -му періоді дискретизації є лише одне значення, яке відповідає цьому періодові і не може бути сформоване раніше, – це вихідний сигнал кодера (вхідний – процесора) – $e_n^{(x)}$.

Саме тому всі перемноження і підсумовування в (1)

$$S_k^{(n)} = \sum_{m=1}^{M-1} e_{k-m}^{(x)} e_m^{(h)}, \quad (8)$$

а також $R_k, S_k^{(n)} + R_n$ можуть бути виконані завчасно [4].

Значення

$$S_k = S_k^{(n)} + e_k^{(x)} e_0^{(h)}, \quad (9)$$

а

$$\nabla^2 y_k = S_k + R_k. \quad (10)$$

Отже, якщо всі дії згідно з (8) виконати в $(n-1)$ -му періоді дискретизації, то в n -му для отримання вихідного відліку y_n необхідно виконати лише операції (9), (10), (6) та (7), до того ж (9) та (10) можна сумістити в часі.

Власне такий алгоритм функціонування дозволяє мінімізувати затримку вихідного сигналу процесора практично без збільшення обсягу процесора, як при розпаралелюванні обробки [5], а також без застосування швидкодіючої елементної бази.

Розглянемо реалізацію запропонованих алгоритмів. Нерекурсивна частина такого процесора (на рис.1 показана його структурна схема) містить аналого-цифровий перетворювач – дельта-кодер ЛДМ (A/D), елемент Виключне АБО, реверсивний лічильник СТ, мультиплексор МХ1 на два положення, блок оперативної RAM1 та постійної ROM1 пам'яті та блок керування СО. Блок керування СО забезпечує

формування адрес ОЗП та ПЗП та формування керуючих сигналів і містить генератор G тактових імпульсів, двійкові лічильники $CT1$ та $CT2$, позиційний суматор $SM3$, елемент АБО-НЕ, формувач імпульсів FI .

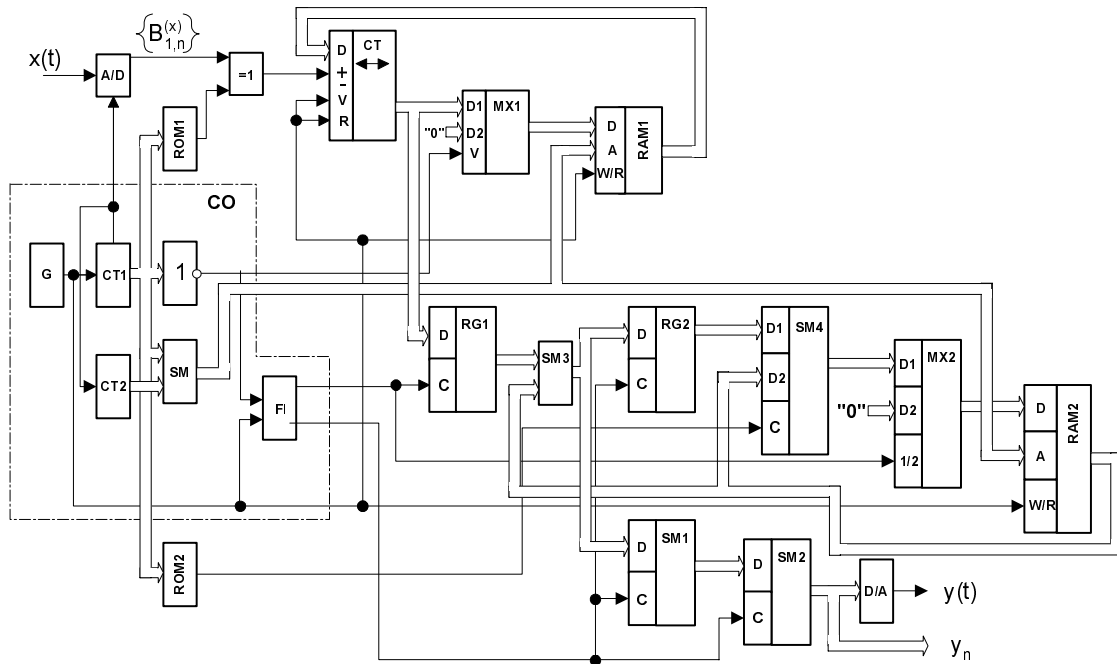


Рис 1. Структурна схема процесора.

Рекурсивна частина відповідно містить другий блок оперативної пам'яті $RAM2$, блок пам'яті вагових коефіцієнтів $\{B_{1,m}^{(g)}\}$ – $ROM2$, буферні регістри $RG1$ та $RG2$, позиційний суматор $SM3$, керований суматор $SM4$, другий мультиплексор $MX2$. Крім того, процесор містить перший та другий накопичувачі $SM1$, $SM2$ для накопичення послідовностей першої та другої різниці вихідного сигналу і цифроаналоговий перетворювач D/A для перетворення послідовностей відліків $\{y_n\}$ в аналогову форму $y(t)$. На рис. 2 показані часові залежності основних сигналів процесора.

Розглянемо спочатку функціонування нерекурсивної частини процесора.

Тактові імпульси з генератора G частотою T^{-1}/M , яка в M разів більша за частоту дискретизації, подаються на лічильник $CT1$, стан m якого відповідає номеру вагового коефіцієнта $\{B_{1,m}^{(h)}\}$, $m = \overline{0, M-1}$. Послідовність цих коефіцієнтів зчитується з виходу $ROM1$ в кожному періоді дискретизації T в одній й тій самій послідовності, починаючи з коефіцієнта $B_{1,0}^{(h)}$ і закінчуючи $B_{1,M-1}^{(h)}$. За імпульсами з генератора G також зчитують значення коду вхідного сигналу в ЛДМ-форматі $B_{1,k}^{(x)}$. За період T це значення, подане на вхід елемента Виключне АБО, перемножується з усіма M коефіцієнтами вагової послідовності, тобто на виході елемента Виключне АБО за цей період формується послідовність $\{B_{1,k}^{(x)} \oplus B_{1,m}^{(h)}\}$, $m = \overline{0, M-1}$. Ця послідовність подається на керуючий вхід реверсивного лічильника CT .

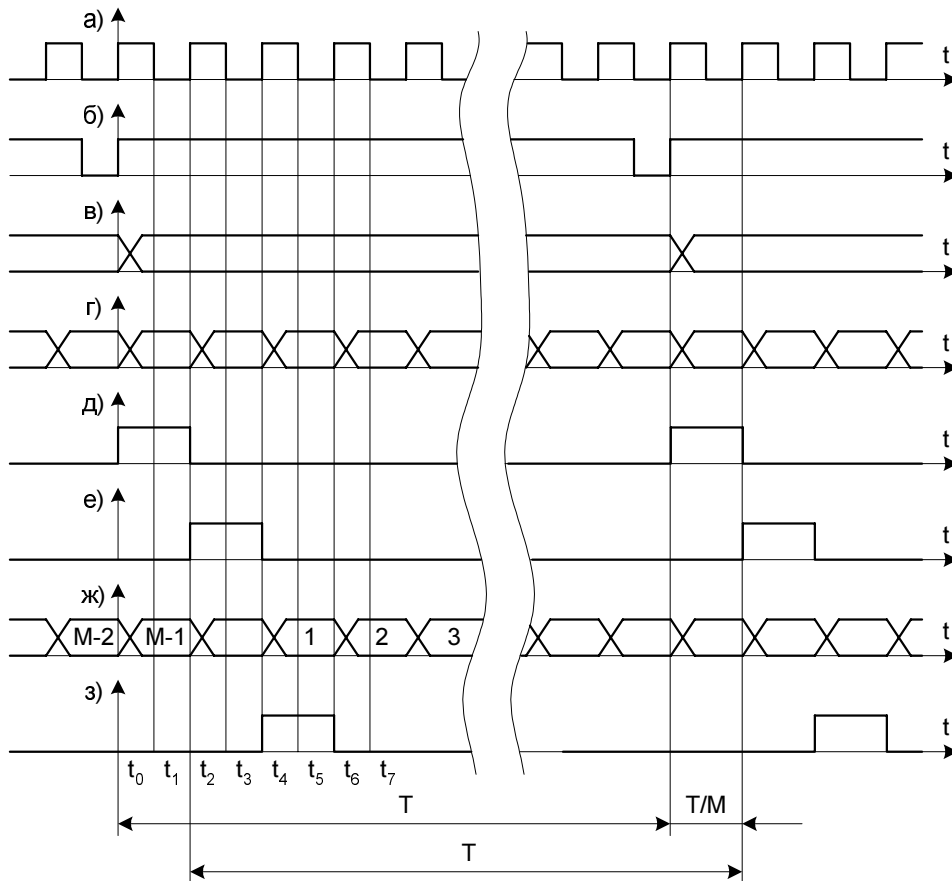


Рис 2. Часові залежності сигналів процесора.

В інтервалах часу $[t_0, t_1]$, $[t_2, t_3]$ і т.д., тобто при одиничному значенні сигналу генератора G, проводиться зчитування з блока RAM і запис по входах попереднього встановлення реверсивного лічильника СТ величини сигналу, який відповідає неповній згортці, тобто згортці, яка має менше ніж M складових. По задньому фронту тактового імпульса (рис.2а) записаний в лічильнику СТ сигнал підсумовується з результатом перемноження $2 \left(\overline{B_{1,k}^{(x)} \oplus B_{1,m}^{(h)}} \right) - 1$ кроку вхідного сигналу на крок вагової послідовності.

В інтервалі часу $[t_3, t_4]$ і т.д. проводиться запис накопиченої величини сигналу, який відповідає неповній згортці, в ту саму комірку блока RAM1 через мультиплексор MX1. Під час запису мультиплексор знаходиться в першому положенні, тобто в цьому випадку відсутній імпульс (рис. 2д) з виходу елемента АБО-НЕ. Послідовність комірок блока RAM1 визначається сумою стану першого СТ1 та другого СТ2 лічильників за допомогою суматора SM3. Це забезпечує зсув номерів комірок блока оперативної пам'яті RAM відносно номерів комірок блока постійної пам'яті ROM на одиницю в кожному наступному періоді дискретизації, що забезпечує алгоритм рециркуляції (рис.3).

Накопичення значень сигналу у вигляді неповної згортки відбувається протягом всіх інтервалів часу T/M , крім інтервалу $[t_0, t_2]$. В цьому проміжку часу за допомогою елемента АБО-НЕ визначається нульовий стан лічильника СТ1, що призводить до

формування відповідного імпульсу (рис. 2d) на виході елемента АБО-НЕ. Цей імпульс є керуючим для мультиплектора MX1 і його вхід підключається до завдання нульового коду “0”. Тому упродовж часу $[t_1, t_2]$ в комірку блока RAM1 записується нульове значення, а в момент часу t_1 у реверсивному лічильнику СТ формується сигнал, який відповідає сигналу S_n .

Як видно з рис.3, номери комірок RAM1 періодично, з періодом M , повторюються, а число накопичених в них значень, за винятком перехідного процесу для $n < M$, дорівнює $(m + n) \bmod M$, де m – стан лічильника СТ1, n – стан лічильника СТ2, який відповідає номеру відліку вхідного сигналу. Тому комірки RAM1 підготовлені до наступного накопичення відповідних добутоків, а сигнал S_n в момент часу t_2 фіксується в регістрі RG1.

В інтервалі часу $[t_2, t_3]$ виконуються зчитування із блока RAM2 значення R_k і подача його на входи суматора SM3. Тому на виході SM3 формується значення сигналу згідно з (10), яке записується в момент часу t_4 в другий буферний регістр RG2 і перший накопичувач SM1. Це значення (рис. 2e) підсумовується з накопиченим раніше значенням в накопичувачі SM1, а в момент часу t_5 відбувається накопичення вихідного сигналу суматора SM1 (першої різниці вихідного сигналу ∇y_k) з попереднім відліком вихідного сигналу y_{k-1} , формуючи поточний відлік сигналу процесора $y_n = y_{n-1} + \nabla y_n$ у форматі повнорозрядного подання. За допомогою D/A цей сигнал перетворюється в аналогову форму $y(t)$.

Розглянемо формування послідовності значень $\{R_k\}$ в рекурсивній частині процесора.

За сигналом з другого виходу формувача FI блок MX2 переводиться в положення, при якому його входи підключені до завдання нульового значення “0”. В результаті цього протягом інтервалу часу $[t_3, t_4]$ в комірку RAM2 записується нульове значення.

Упродовж інтервалу часу $(T - T/M)$ – від моменту t_4 поточного періоду дискретизації до t_2 – наступного, значення сигналу $\nabla^2 y_k$ з другого регістра RG2 надходять на вхід керованого суматора SM4, на керуючий вхід якого з RAM2 подаються всі вагові коефіцієнти $\{B_{1,m}^{(g)}\}$, $m = \overline{1, M-1}$, кожний протягом часу T/M (рис. 2ж). Порядок зміни коефіцієнтів постійний і задається першим лічильником СТ1, причому упродовж часу $[t_2, t_4]$ здійснюється зчитування коефіцієнта $B_{1,1}^{(g)}$.

В інтервалах часу $[t_4, t_5]$ і т.д. здійснюється зчитування із другого блока RAM2 і подача на інші входи SM4 величини сигналу, що відповідає значенню R_n для M наступних періодів дискретизації.

За допомогою керуючого суматора SM4 здійснюється накопичення відповідних добутоків $\{e_m^{(g)} \nabla^2 y_{k-m}\}$, причому при $B_{1,m}^{(g)} = 1$ відбувається підсумовування вихідного сигналу RAM2 з $\nabla^2 y_k$, а при $B_{1,m}^{(g)} = 0$ – віднімання. За низьким рівнем сигналу з

генератора G (рис. 2а) здійснюється запис накопиченого значення в ту саму комірку RAM2 через блок мультиплексування MX2. Номери комірок RAM2 збігаються з номерами комірок RAM1 і також змінюються за запропонованим алгоритмом рециркуляції.

№ циклу	№ такту	Вагова послідовність (адреси ROM)	Вхідний сигнал	Накопичене значення та адреси RAM1
1	1	$0 - e_0^{(h)}$	$e_1^{(x)}$	$0 - e_0^{(h)} e_1^{(x)} = S_1$
	2	$1 - e_1^{(h)}$		$1 - e_1^{(h)} e_1^{(x)}$
	3	$2 - e_2^{(h)}$		$2 - e_2^{(h)} e_1^{(x)}$
	4	$3 - e_3^{(h)}$		$3 - e_3^{(h)} e_1^{(x)}$
2	1	$0 - e^{(h)}$	$e_2^{(x)}$	$1 - e_2^{(x)} e^{(h)} + e_1^{(h)} e_1^{(x)} = S_2$
	2	$1 - e_1^{(h)}$		$2 - e_2^{(h)} e_1^{(x)} + e_1^{(h)} e_2^{(x)}$
	3	$2 - e_2^{(h)}$		$3 - e_3^{(h)} e_1^{(x)} + e_2^{(h)} e_2^{(x)}$
	4	$3 - e_3^{(h)}$		$0 - e_3^{(h)} e_2^{(x)}$
3	1	$0 - e^{(h)}$	$e_3^{(x)}$	$2 - e_3^{(x)} e_0^{(h)} + e_2^{(x)} e_1^{(h)} + e_1^{(x)} e_2^{(h)} = S_3$
	2	$1 - e_1^{(h)}$		$3 - e_3^{(x)} e_1^{(h)} + e_2^{(x)} e_2^{(h)} + e_1^{(x)} e_3^{(h)}$
	3	$2 - e_2^{(h)}$		$0 - e_3^{(x)} e_2^{(h)} + e_3^{(h)} e_2^{(x)}$
	4	$3 - e_3^{(h)}$		$1 - e_3^{(x)} e_3^{(h)}$
4	1	$0 - e^{(h)}$	$e_4^{(x)}$	$3 - e_4^{(x)} e_0^{(h)} + e_3^{(x)} e_1^{(h)} + e_2^{(x)} e_2^{(h)} + e_1^{(x)} e_3^{(h)} = S_4$
	2	$1 - e_1^{(h)}$		$0 - e_4^{(x)} e_1^{(h)} + e_3^{(x)} e_2^{(h)} + e_2^{(x)} e_3^{(h)}$
	3	$2 - e_2^{(h)}$		$1 - e_4^{(x)} e_2^{(h)} + e_3^{(x)} e_3^{(h)}$
	4	$3 - e_3^{(h)}$		$2 - e_4^{(x)} e_3^{(h)}$
5	1	$0 - e_0^{(h)}$	$e_5^{(x)}$	$0 - e_5^{(x)} e_0^{(h)} + e_4^{(x)} e_1^{(h)} + e_3^{(x)} e_2^{(h)} + e_2^{(x)} e_3^{(h)} = S_5$
	2	$1 - e_1^{(h)}$		$1 - e_5^{(x)} e_1^{(h)} + e_4^{(x)} e_2^{(h)} + e_3^{(x)} e_3^{(h)}$
	3	$2 - e_2^{(h)}$		$2 - e_5^{(x)} e_3^{(h)}$
	4	$3 - e_3^{(h)}$		$3 - e_5^{(x)} e_3^{(h)}$
6	$e_6^{(x)}$...

Рис. 3. Пояснення алгоритму рециркуляції нерекурсивної частини процесора для $M = 4$.

Накопичення значень $R_k^{(1)}$ відбувається аналогічно до накопичення $S_k^{(1)}$ і здійснюється до моменту часу t_2 наступного періоду дискретизації, після чого період накопичення і цикл роботи процесора повторюється (рис. 2з).

У разі вибору $M = 2^r, r > 1$ блок СО можна реалізувати на r -розрядних лічильниках і $2r$ -входовому комбінаційному суматорі.

Затримка формування значень другої різниці в даному процесорі мінімальна і дорівнює часу накопичення значень в реверсивному лічильнику СТ $t \leq T/M$. Формування значень першої ∇y_n та другої різниці $\nabla^2 y_n$ відбувається також через час T/M , тобто загальна затримка вихідного сигналу відносно подачі вхідного не перевищує часу

$$\tau \leq 3T/M,$$

що забезпечує високу швидкодію і можливість роботи процесора в замкнутому контурі керування швидкоплинними процесами.

1. Тимченко О.В. Різницеві методи цифрової фільтрації. Львів, 1999. 2. Дурняк Б.В., Стрепко І.Т., Тимченко О.В. Алгоритми швидкодіючих систем реального часу, побудованих на основі різницевих підходів // Вісн. ДУ "Львівська політехніка", 1999. №366. С.56-62. 3. Стрепко І.Т., Тимченко О.В., Дурняк Б.В. Проектування систем керування на однокристальних мікро-ЕОМ. К., 1998. 4. А.с. №1418908 ССРСР. Цифрової фільтр с дельта-модуляцією / Тимченко А.В., Тимченко С.В. (ССРСР) // Бюл. изобр. №31. 1988. 5. Дурняк Б., Стрепко І., Тимченко О. Розпаралелювання обчислень на основі вибору методів різницевого подання сигналів в САК реального часу // Комп'ютерні технології друкарства. Львів, 1998. С.120-123.

УДК 681.518.54

ІДЕНТИФІКАЦІЯ ВНУТРІШНІХ ПАРАМЕТРІВ ОБМОТОК ЕЛЕКТРИЧНИХ МАШИН ЗА ДОПОМОГОЮ ТЕОРІЇ ЧУТЛИВОСТІ

© Кучерук В.Ю., Поджаренко А.В., Селезньова Р.В., 2000

Вінницький державний технічний університет, кафедра "Метрологія та промислова автоматика"

Запропоновано метод ідентифікації внутрішніх параметрів обмоток електричних машин за допомогою теорії чутливості і проведено його дослідження. Метод дозволяє з малою похибкою ідентифікувати активний опір обмотки. Похибка ідентифікації ємності обмотки має більші значення. Це свідчить про доцільність ідентифікації ємності іншими методами ідентифікації, наприклад, методами інваріантного поглиблення або аналітичними методами. Знайдене значення ємності обмотки можна використати для початкового наближення під час ідентифікації іншими методами.