

УДК 681.3

А.О.Мельник, Мохаммад аль Хабабсах, В.А. Мельник

Національний університет „Львівська політехніка”

ПРОЕКТУВАННЯ БАГАТОКАНАЛЬНОГО ПРОЦЕСОРА ШПФ З ВИКОРИСТАННЯМ ПАМ'ЯТІ З ВПОРЯДКОВАНИМ ДОСТУПОМ

. В статті проводиться проектування багатоканального процесора ШПФ, досліджуються його швидкісні характеристики та затрати обладнання на його реалізацію залежно від розміру перетворення для 32-розрядної сітки для 2, 4, 8, 16 та 32-канальних ланок процесора при різній кількості даних та різній кількості каналів поступлення даних. При цьому процесор реалізується на основі пам'яті з впорядкованим доступом за новою методикою. Наводяться результати синтезу, моделювання та тестування процесора.

Вступ

Для отримання швидких алгоритмів обчислення дискретного перетворення Фур'є існує багато методів, до яких відносяться алгоритми за основою два та чотири (метод Кулі-Тьюкі), за розщепленою основою, алгоритми Винограда, з дійсними фазовими множниками (за методом Рейдера-Бреннера), на основі поліноміальних перетворень, за векторною основою, гніздові двовимірні алгоритми, на основі перетворення Радона [1]. Хоча кожний з цих методів має ті або інші переваги, проте найширше використовують методи Кулі-Тьюкі та з дійсними фазовими множниками завдяки простоті їх базових операцій. Алгоритми ШПФ з дійсними фазовими множниками у порівнянні з алгоритмами за методом Кулі-Тьюкі характеризуються меншою кількістю множень та простішою базовою операцією, проте вимагають додаткових етапів операцій додавання. При побудові процесорів ШПФ в більшості випадків використовується метод Кулі-Тьюкі [2]. Нижче проводиться проектування багатоканального процесора ШПФ за методом Кулі-Тьюкі, досліджуються його швидкісні характеристики та затрати обладнання на його реалізацію залежно від розміру перетворення для 32-розрядної сітки для 2, 4, 8, 16 та 32-канальних ланок процесора при різній кількості даних та різній кількості каналів поступлення даних. При цьому процесор реалізується на основі пам'яті з впорядкованим доступом за новою методикою. Наводяться результати синтезу, моделювання та тестування процесора. Наведена нижче методика може бути використана для синтезу спеціалізованих процесорів, які виконують інші алгоритми.

1.Багатоканальний процесор ШПФ

Якщо $x(n)$ є послідовністю комплексних чисел, а $X(k)$ - її дискретним перетворенням Фур'є, $n, k=0, 1, \dots, N-1$, то

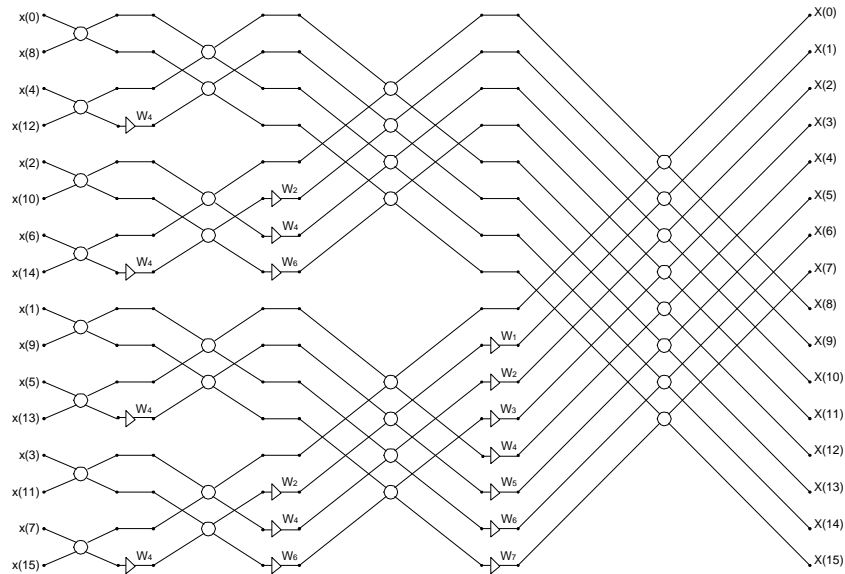
$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}, \quad W_N^r = \exp(-j2\pi r / N). \quad (1)$$

Формула розкладу алгоритму швидкого перетворення Фур'є за основою два з часовим прорідженням за методом Кулі-Тьюкі є наступною:

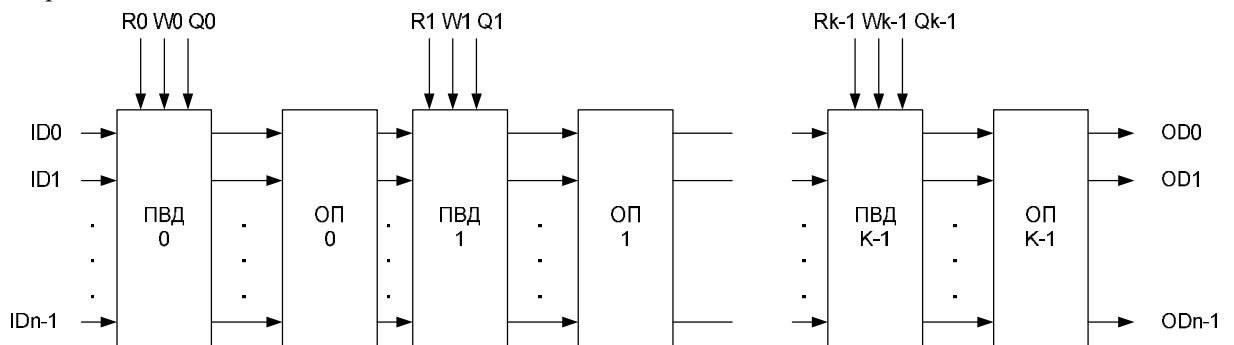
$$X(k) = X_0(k) + W_N^k X_2(k) \\ X(k + N/2) = X_0(k) - W_N^k X_2(k), \quad (2)$$

$k=0, 1, \dots, N/2-1$.

Потоковий граф алгоритму ШПФ за основою два з часовим прорідженням за методом Кулі-Тьюкі показано на рис.1 [3].

Рис.1. Поточковий граф алгоритму ШПФ за основою два з часовим прорідженням для $N=16$.

На основі цього поточкового графу шляхом реалізації ланок, які виконують операції одного ярусу графа алгоритму, отримуємо структуру багатоканального поточкового процесора ШПФ, показану на рис.2 [4].

Рис.2. Структура багатоканального поточкового процесора ШПФ для $N=2^k$.

2. Характеристики багатоканального процесора ШПФ

Кожним каналом здійснюється передача комплексного числа, що в дійсності говорить про наявність $2n$ каналів для передачі дійсних чисел. На рисунку ПВД - пам'ять з впорядкованим доступом, об'єм якої рівний подвійному розміру перетворення, тобто $N=2^m$ комплексних чисел. ОП – операційний пристрій. В кожному ОП виконується $N/2$ базових операцій алгоритму ШПФ. Залежно від кількості каналів поступлення даних вибирають кількість одночасно виконуваних базових операцій алгоритму ШПФ, які реалізуються в одному ОП та, відповідно, кількість пристроїв для виконання базових операцій алгоритму ШПФ. Їх може бути 2, 4 і т.д. аж до $N/2$. Так само і ПВД може мати від 1 до N каналів поступлення даних.

Відповідно, оскільки опрацьовують комплексні числа, то кожна ПВД має об'єм $N=2^m$ комплексних чисел, або $2N$ дійсних чисел. Для забезпечення опрацювання даних в реальному часі необхідно забезпечити одночасне приймання та зчитування даних, тому необхідно включити паралельно 2 ПВД. Тому ємність ПВД однієї ланки становить $4N$ комірок, кожна з яких зберігає одне дане, а створювана затримка в ПВД становить N тактів. Для розміру перетворення N кількість ланок рівна $k = \log_n N$, де n – кількість входів ланки процесора. Загальний об'єм ПВД для всього процесора становить $4N \log_n N$. Відповідно затримка проходження даних крізь ПВД

$$T_{\text{ПВД}} = 4N \log_n N \quad (3)$$

тактів.

Оскільки затрати обладнання на реалізацію ПВД процесора потокового типу визначаються з виразу

$$W = W_{CM} + lkW_{Pr} + mnW_B = W_{CM} + NW_{Pr} + NW_B,$$

де W_{CM} , W_{Pr} , W_B – відповідно затрати на сортувальну мережу CM , на регістри та на вихідні вентиля, то загальні затрати обладнання на реалізацію ПВД процесора ШПФ будуть рівні

$$W_{ПВД} = \log_n N (W_{CM} + 4NW_{Pr} + 4NW_B). \quad (4)$$

Затрати обладнання на сортувальну мережу ПВД визначаються з виразу [5]

$$W_{CM} = 2^{\log_2 N - 2} (\log^2 N - \log N + 4) \text{ БЕ},$$

де БЕ – затрати обладнання на реалізацію базового елемента CM , який складається з схеми порівняння та двох двохходових мультиплексорів. Оскільки на реалізацію одного двохходового мультиплексора потрібно 4 вентиля, одного тригера – 7 вентилів, то з врахуванням внутрішньої структури ПВД затрати на її реалізацію в кількості вентилів рівні:

$$W_{ПВД} = \log_n N (12m(2^{\log_2 N - 2} (\log^2 N - \log N + 4)) + 32Nm), \quad (5)$$

де n – кількість входів процесора, m розрядність даних.

Розглянемо далі питання реалізації операційного пристрою, в якому виконуються базові операції алгоритму ШПФ, які описуються формулою (2). Структура ОП, який реалізує одну базову операцію алгоритму ШПФ, показана на рис.3. Виконувана базова операція описується виразом

$$A' = A + WB,$$

$$B' = A - WB,$$

де $A = a_1 + ja_2$, $B = b_1 + jb_2$, $W = w_1 + jw_2$.

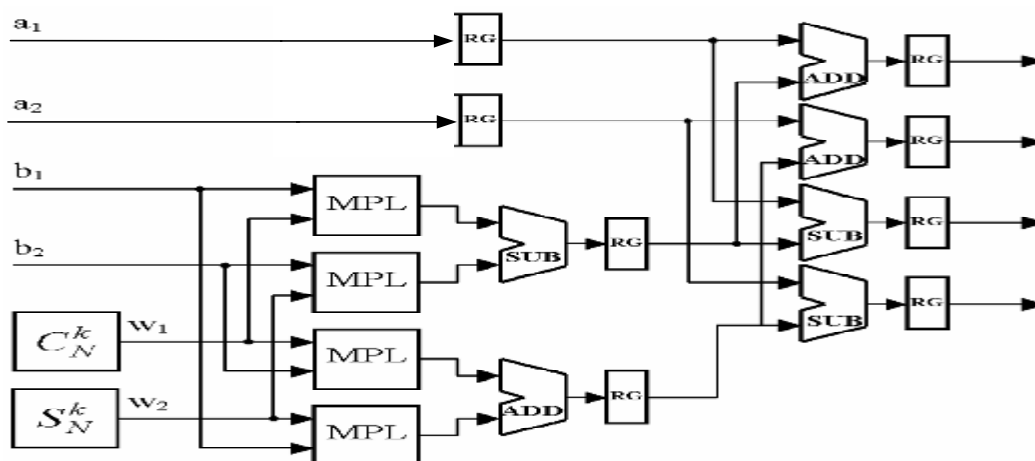


Рис.3. Структура ОП двоканального процесора ШПФ

При умові двоканального потоку даних враховуючи комплексну структуру чисел для реалізації ОП необхідно 4 перемножувачі MPL , 6 додавачів/віднімачів ADD , конверсні регістри та ПЗП фазових множників. Якщо i - номер ОП, $i = 0, 1, \dots, \log_2 N - 1$, то розмір ПЗП в i -му ОП дорівнює 2^{i+1} дійсних чисел. На трактах передачі першого комплексного числа $a = a_1 + ja_2$ може бути розміщено лінійки регістрів, кількість яких визначається глибиною конвеєра перемножувачів. Якщо m – розрядність даних, тоді максимальна глибина конвеєра перемножувача дорівнюватиме $\log_2 m$. Відповідно довжина лінійки регістрів з врахуванням додаткового регістра після додавачів/віднімачів дорівнюватиме $\log_2 m + 1$. Цієї лінійки регістрів можна не ставити (як це й зроблено на рис.3), але тоді в ПВД потрібно здійснювати зміщення даних на відповідну затримці регістрів конвеєра кількість тактів. Загальна кількість регістрів в такому ОП дорівнюватиме $2\log_2 n + 8$. При n -канальному потоці даних кількість описаних вище пристроїв в одному ОП буде рівна $n/2$. Тоді затрати на реалізацію всіх ОП процесора ШПФ дорівнюють:

$$W_{ОП} = \log_n N ((4W_{MPL} + 6W_{ADD} + (2\log_2 m + 8)W_{RG}) + (2N - 2)W_{ROM}) n / 2, \quad (6)$$

де W_{MPL} - затрати на перемножувач, W_{ADD} - затрати на додавач/віднімач, W_{RG} - затрати на регістр, W_{ROM} - затрати на комірку ПЗП.

Якщо врахувати, що затрати обладнання у вентилях на перемножувач рівні $W_{MPL} = 10m^2$, на суматор $10m$, де m – розрядність даних, а на комірку ПЗП – m , то затрати обладнання у вентилях на ОП процесора ШПФ дорівнюють

$$W_{OP} = \log_n N((40m^2 + 60m + 7(2\log_2 m + 8)) + (2N - 2)m)n/2, \quad (7)$$

а загальні затрати обладнання у вентилях на реалізацію процесора ШПФ будуть рівні:

$$W_{ШПФ} = \log_n N(12m(2^{\log_2 N - 2}(\log^2 N - \log N + 4)) + 32Nm) + \log_n N((40m^2 + 60m + 7(2\log_2 m + 8)) + (2N - 2)m)n/2. \quad (8)$$

На рис.4 зображено графік залежності W від N відповідно до виразу 8 для $N=64, 128, 256, 512, 1024, 2048, 4096$ при $m=16$ для n -канального потоку даних при $n=2, 4, 8, 16, 32$. З цього графіка можна оцінити затрати на СП ШПФ при різній кількості даних N та різній кількості каналів n поступлення даних та вибрати найвигідніший варіант процесора.

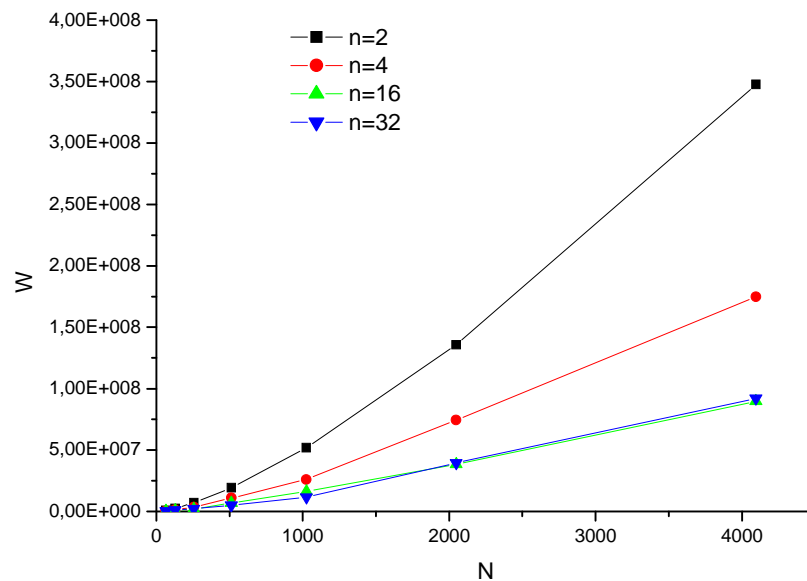


Рис.4. Графік залежності W від N відповідно до виразу 4.8 для n -канального потоку даних при $n=2, 4, 8, 16, 32$ (відповідно криві 1-5)

Затримка проходження даних крізь пристрої ОП залежить від глибини його конвеєризації. Якщо глибина конвеєра ОП дорівнює $\log_2 m + 2$, то загальна затримка проходження даних через процесор складає

$$T_k = 4N \log_n N + \log_n N(\log_2 m + 2)t, \quad (9)$$

де t – час виконання одного такту роботи конвеєра.

Час виконання алгоритму ШПФ в розробленому процесорі ШПФ визначається затримкою однієї його ланки, оскільки при її вивільненні до її ПВД завантажуються дані наступного масиву даних, та складає

$$T_A = 4N + (\log_2 m + 2)t, \quad (10)$$

де t – час виконання одного такту роботи конвеєра.

3. Визначення вихідної інформації для синтезу процесора ШПФ

Визначимо вихідну інформацію для синтезу процесора ШПФ, а саме: параметри матриць вхідних даних, параметри матриць вихідних даних, сформуємо матриці вхідних даних та правила впорядкування вхідних даних для всіх матриць вхідних даних, а також сформуємо матриці кодів виконуваних операцій, тобто команд, для операційного пристрою процесора.

Проведемо етап маркування ПГА алгоритму ШПФ, на якому виконаємо нумерацію його вершин, дуг та портів (рис.5).

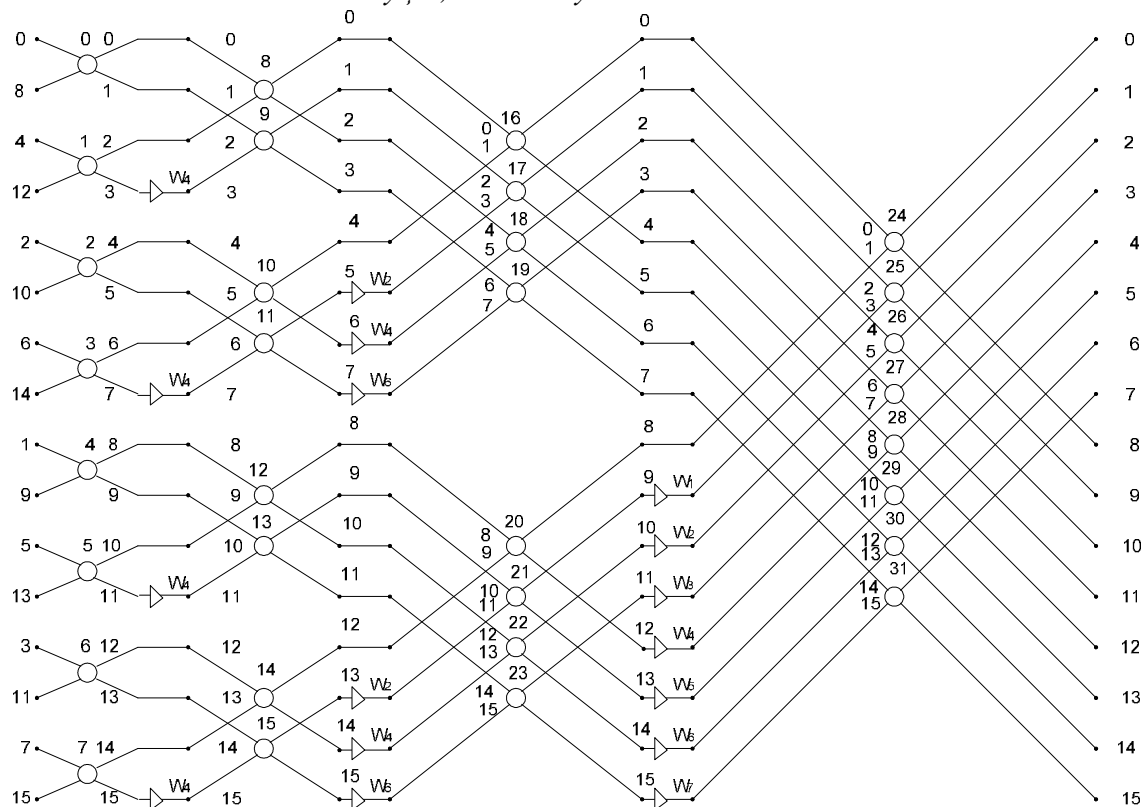


Рис.5. Поточковий граф алгоритму ШПФ за основою два з часовим прорідженням для $N=16$ після виконання маркування.

Вершини графа нумеруємо натуральними числами $k = \overline{0,31}$. Ці вершини розподіляються по чотирьох ярусах так, що в i -му ярусі ($i = \overline{0,3}$) розміщені тільки вершини, на які надходять дуги від вершин попередніх ярусів і не надходять від вершин того ж ярусу та наступних ярусів. Нумерацію ярусів починаємо з ярусу, на якому виконуються операції над вхідними даними, і закінчуємо ярусом, в якому знаходяться вихідні дуги ПГА. Вхідні порти графа нумеруються числами від 0 до 15. Кількість n вхідних дуг поточкового графа алгоритму дорівнює 16. Нумерацію дуг розпочинаємо з першого ярусу і кожній вхідній дузі даного ярусу присвоюємо натуральне число j ($j = \overline{0,15}$), аналогічно як і кожній вхідній дузі інших ярусів. Нумерацію вихідних портів функціональних операторів розпочинаємо з першого ярусу і кожному вихідному порту даного ярусу присвоюємо натуральне число i ($i = \overline{0,15}$), аналогічно і кожному вихідному порту наступних ярусів.

Для кожного ярусу ПГА виділяємо по одному вектору вхідних даних, номерів дуг та вихідних портів функціональних операторів, а також кодів операцій.

Вектори вхідних даних ПГА кожного ярусу матимуть наступний вигляд:
 $| D_0, D_1, D_2, D_3, D_4, D_5, D_6, D_7, D_8, D_9, D_{10}, D_{11}, D_{12}, D_{13}, D_{14}, D_{15} |$.

Аналогічно вектори номерів дуг ярусів ПГА мають ту ж нумерацію, оскільки їх значення співпадає з номерами даних, тобто

$| 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 |$

Вектори номерів вихідних портів функціональних операторів ярусів ПГА мають наступну нумерацію:

- для входу: $| 0, 8, 4, 12, 2, 10, 6, 14, 1, 9, 5, 13, 3, 11, 7, 15 |$;
- для першого ярусу: $| 0, 2, 1, 3, 4, 6, 5, 7, 8, 10, 9, 11, 12, 14, 13, 15 |$;
- для другого ярусу: $| 0, 4, 1, 5, 2, 6, 3, 7, 8, 12, 9, 13, 10, 14, 11, 15 |$;
- для третього ярусу: $| 0, 8, 1, 9, 2, 10, 3, 11, 4, 12, 5, 13, 6, 14, 7, 15 |$;
- для четвертого ярусу і виходу: $| 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15 |$.

Наклавши вектори дуг та вихідних портів функціональних операторів отримуємо наступні правила впорядкування вхідних даних ярусів ПГА:

- для першого ярусу:

| | | | | | | | | | | | | | | | | |
|--------|---|---|---|----|---|----|---|----|---|---|----|----|----|----|----|----|
| Входи | 0 | 8 | 4 | 12 | 2 | 10 | 6 | 14 | 1 | 9 | 5 | 13 | 3 | 11 | 7 | 15 |
| Виходи | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

- для другого ярусу:

| | | | | | | | | | | | | | | | | |
|--------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|
| Входи | 0 | 4 | 1 | 5 | 2 | 6 | 3 | 7 | 8 | 12 | 9 | 13 | 6 | 14 | 7 | 15 |
| Виходи | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

- для третього ярусу:

| | | | | | | | | | | | | | | | | |
|--------|---|---|---|---|---|----|---|----|---|----|----|----|----|----|----|----|
| Входи | 0 | 8 | 1 | 9 | 2 | 10 | 3 | 11 | 4 | 12 | 5 | 13 | 6 | 14 | 7 | 15 |
| Виходи | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

- для четвертого ярусу і виходу:

| | | | | | | | | | | | | | | | | |
|--------|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|
| Входи | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
| Виходи | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |

За цими правилами здійснюється впорядкування даних в комутуючих мережах ПВД відповідних ланок потокового процесора ШПФ.

Оскільки граф алгоритму обчислення ШПФ має 32 вершини, причому різних лише 8, то для закодування виконуваних у вершинах графа операцій достатньо 3 розряди. Позначимо операції наступними кодами: з множителем $W_0 - 000$, з множителем $W_1 - 001$, з множителем $W_2 - 010$, з множителем $W_3 - 011$, з множителем $W_4 - 100$, з множителем $W_5 - 101$, з множителем $W_6 - 110$, з множителем $W_7 - 111$. Тоді вектори кодів операцій ГА виразу матимуть наступний вигляд:

- для першого ярусу: $|000, 000, 000, 000, 000, 000, 000, 000|$;
- для другого ярусу: $|000, 100, 000, 100, 000, 100, 000, 100|$;
- для третього ярусу: $|000, 010, 100, 110, 000, 010, 100, 110|$;
- для четвертого ярусу: $|000, 001, 010, 011, 100, 101, 110, 111|$.

Отже, здійснено перегляд усіх вершин, портів та дуг кожного ярусу та записано в відповідні вектори.

4. VHDL-модель процесора ШПФ

Вхідні, вихідні і проміжні дані у процесорі ШПФ подаються в доповняльному коді. Старший біт використовується як знаковий, а решта - як дані. Старший значущий біт знаходиться відразу за знаковим.

У ядрі процесора розрядність даних параметризується. Процесор розроблено для виконання ШПФ з розмірами масивів даних, поданими в табл.1.

Таблиця 1

| | | | | | | | | |
|--------|-----|-----|-----|-----|-----|------|------|------|
| Розмір | 32 | 64 | 128 | 256 | 512 | 1024 | 2048 | 4096 |
| Код | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

Щоб задати розмір перетворення використовується 3-бітовий код. Він подається процесору перед початком обробки вхідних даних на вхід процесора і фіксується у регістрі розміру перетворення заднім фронтом сигналу скидання.

Розмір перетворення може бути збільшений шляхом збільшення кількості елементів процесора усередині VHDL моделі.

У початковий стан процесор встановлюється сигналом скидання reset. Коли на вхід подано низький рівень, тригери, лічильники та регістри, на які подається сигнал, встановлюються у нульовий стан.

Процесор ШПФ вводить вхідні дані і виводить результати тільки тоді, коли сигнали i-ack (підтвердження вводу) і o-ack (підтвердження виводу) знаходяться у високому стані. Вхідні дані вводяться слово за словом із кожним тактом сигналу синхронізації clock, коли i-ack високого рівня, і виводяться із кожним тактом синхросигналу, коли o-ack у високому стані.

Вхідні дані надходять у процесорі ШПФ виводяться з нього як масиви даних. Розмір масиву визначається розміром перетворення. Дані в масиві розміщені в прямому порядку (нульовий елемент, перший, другий, третій, і так далі).

Масиви з тими ж самими розмірами можуть перебувати в процесор постійно, без пауз, тобто перші дані наступного масиву прибувають у процесор при такті синхросигналу що слідує після прийому останнього елемента попереднього масиву. У цьому випадку результати виводяться без паузи також.

Часова діаграма симуляції роботи процесора ШПФ в пакеті Xilinx Foundation 2.1:

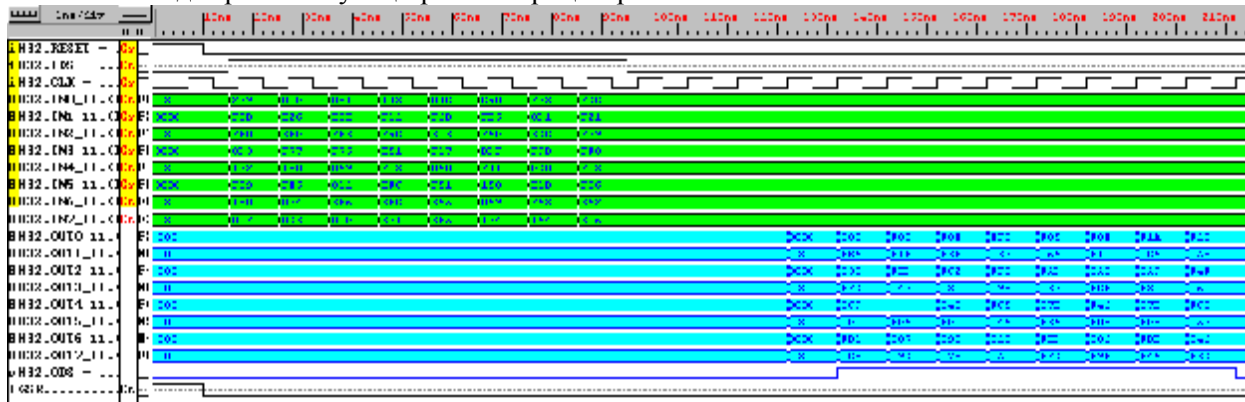


Рис.6. Часова діаграма симуляції роботи процесора ШПФ

Реалізація розробленого процесора на ПЛІС Xilinx XC4085XLABG560 з використанням пакету проектування Xilinx Foundation 2.1 дало наступні результати за затратами обладнання та продуктивності:

- кількість логічних комірок - 877 CLB;
- максимальна тактова частота – 84,9 МГц.

Для порівняння в [6] розглядається побудова паралельного 8-канального процесора ШПФ. В цьому процесорі реалізація 8-канального процесора ШПФ вимагає 1664 CLB при максимальній тактовій частоті 50 МГц. Таким чином, в розробленому процесорі отримано скорочення затрат обладнання майже в 2 рази при підвищенні швидкодії в 1,5 рази.

Висновки

1. Проведено проектування багатоканального процесора ШПФ за методом Кулі-Тьюкі, розроблено його складові, операційний пристрій, ПВД, ланка, отримано аналітичні вирази для оцінки їх швидкісних характеристик та затрат обладнання на їх реалізацію, а також процесора в цілому. Отримано графіки залежності затрат обладнання на процесор ШПФ від розміру перетворення для 32-розрядної розрядної сітки для 2, 4, 8, 16 та 32-канальних ланок процесора, з яких можна оцінити затрати на процесор при різній кількості даних та різній кількості каналів поступлення даних та вибрати найвигідніший варіант.

2. Виконано розроблення конфігуровної програмної моделі процесора ШПФ та здійснена її реалізація в реконфігурованому прискорювачі, для чого визначена вихідна інформація для синтезу процесора ШПФ, а саме: параметри матриць вхідних даних, параметри матриць вихідних даних, сформовані матриці вхідних даних та правила впорядкування вхідних даних для всіх матриць вхідних даних, а також сформовані матриці кодів виконуваних операцій, тобто команд, для операційного пристрою процесора шляхом проведення маркування ПГА ШПФ та використано розроблені методи, розроблена VHDL-модель процесора ШПФ, проведено синтез його вузлів, здійснено моделювання та тестування процесора.

1. Задирака В.К., Мельникова С.С.. Цифровая обработка сигналов. – Киев: Наукова думка, 1993. – 294 с.
2. Мельник А.О. Програмовані процесори обробки сигналів. - Львів. „Львівська політехніка, 2000. - 57 с.
3. Oppenheim, A. V. and R. W. Schaffer. Discrete-Time Signal Processing. Prentice-Hall, Englewood Cliffs, NJ, 1989.
4. А. О. Мельник, М. Аль Хабабсах. Структурна організація ланок потокових процесорів на основі пам'яті з впорядкованим доступом / А. О. Мельник, М. Аль Хабабсах // Наукові нотатки. – Луцьк: Луцький національний технічний університет, 2010. - С. 220-224.
5. А. Melnyk. Organization and application of the programmable ordered access memory / A. Melnyk, J. Al Rawashdeh, M. Al Hababsah // Proceedings of Xth International Conference “The Experience of Designing and Application of CAD Systems in Microelectronics”. - Polyana-Svalyava, Lviv Polytechnic National University, 2009. P 240-241.
6. Wosnitza, M., M. Cavadini, M. Thaler, and G. Tröster. \A High Precision 1024-point FFT Processor for 2D Convolution." In IEEE International Solid-State Circuits Conference, volume 41, pp. 118{119, 424, 1998.