

УДК 621.391:004.94

## РЕАЛІЗАЦІЯ ШВИДКОГО ПЕРЕТВОРЕННЯ ФУР'Є НА ПЛІС

Васильєв О.Ю.

Науковий керівник – проф. Хаханова І.В.

Харківський національний університет радіоелектроніки  
(61166, Харків, пр. Науки, 14, каф. АПВТ, тел. (057) 702-13-26)

e-mail: irina.hahanova@nure.ua

This work focuses on researching and implementing a real-time Fast Fourier Transform (FFT) application using a pipelined approach on modern Field Programmable Gate Arrays (FPGA). The author provides theoretical foundation and proposes a methodology for selecting solutions with optimal resource efficiency. Following this methodology, the FFT application is designed, verified, and implemented on Xilinx FPGA. The choice of Xilinx FPGA adds practical validation to the proposed methodology. These inquiries offer avenues for further understanding the research and its implications.

### Вступ.

В сучасному світі цифрові пристрої виконують ключову роль у формуванні сьогоденного життя. Ці пристрої є основою інфраструктури сучасного суспільства та постійно взаємодіють із навколишнім світом, здійснюючи спостереження за допомогою різноманітних датчиків. В процесі моніторингу навколишнього світу ці датчики генерують цифрові сигнали, що є об'єктами уваги для методів аналізу цифрових сигналів. Один із широко використовуваних способів аналізу цифрових сигналів є перетворення Фур'є, а апаратна реалізація цього алгоритму дозволяє проводити аналіз сигналів у реальному часі. *Метою дослідження* є пошук ефективних потокових алгоритмів швидкого перетворення Фур'є та їх відтворення у ПЛІС.

### Зміст дослідження.

Перетворення Фур'є базується на переході від області часу до області частоти. Це перетворення використовується для розкладання сигналу на його синусоїдальні та косинусоїдальні складові частоти. Математично, для функції  $f(t)$  залежної від часу, її перетворення Фур'є  $F(\omega)$  визначається наступним інтегралом:

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \cdot e^{-i\omega t} dt,$$

де  $i$  - уявна одиниця,  $\omega$  - кутова частота,  $t$  - час, а  $F(\omega)$  визначає амплітуду і фазу сигналу для кожної частоти  $\omega$ .

Цифрові системи оперують дискретними сигналами, тому в цифрових системах використовується дискретне перетворення Фур'є (ДПФ), що

визначається формулою:

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-i\frac{2\pi}{N}kn}, \quad 0 \leq k \leq N-1$$

де  $X(k)$  - результат ДПФ для частоти  $k$ ,  $x(n)$  - вхідна послідовність дискретних відліків, а  $N$  - кількість відліків.

ДПФ це ітеративний алгоритм в якому кожне обчислення проходить окремо, а тому він не є ефективним. Для вирішення цієї проблеми Кулі-Тьюклі був запропонований алгоритм швидкого перетворення Фур'є (ШПФ) в якому послідовність довжиною  $N = 2^m$  розділяється на дві послідовності  $\frac{N}{2}$ , потім на чотири  $\frac{N}{4}$  і так далі, що зменшує кількість розрахунків у порівнянні з ДПФ. При цьому використовуються властивості симетрії та періодичності комплексної послідовності:

властивість симетрії

$$W_N^{\frac{k+N}{2}} = -W_N^k,$$

властивість періодичності

$$W_N^{k+N} = W_N^k,$$

де  $W_N^k$  представляє собою корінь  $N$ -го порядку з одиниці, піднесений в ступінь  $k$ ,  $N$  - розмір послідовності даних, а  $k$  - ціло-численний індекс частоти в діапазоні від 0 до  $N-1$ .

Застосувавши ці властивості для послідовності довжиною  $N = 2^m$  точок, отримаємо що ШПФ визначається формулою:

$$X(k) = \sum_{m=0}^{\frac{N}{2}-1} x(2m) \cdot W_{\left(\frac{N}{2}\right)}^{mk} + \sum_{m=0}^{\frac{N}{2}-1} x(2m+1) \cdot W_{\left(\frac{N}{2}\right)}^{mk},$$

де

$$W_{\left(\frac{N}{2}\right)}^{mk} = e^{-i\frac{2\pi}{N}mk}.$$

При цьому застосування ШПФ для перетворення довжиною  $N$  зменшує кількість розрахунків з  $O(N^2)$  для ДПФ до  $O(N \log_2 N)$ .

Широкого застосування для реалізації ШПФ набули програмовані логічні інтегральні схеми (ПЛІС). Значної перевагою ПЛІС є можливість розрахунків в реальному часі з константною очікуваною затримкою результату. Це можливо досягти завдяки конвеєрному підходу, що є високо-ефективною апаратною реалізацією алгоритму ШПФ. Конвеєрні реалізації можна розподілити на два архітектурні типи: зворотного зв'язку (з затримкою одного або багатьох шляхів) та попереднього зв'язку (з комутацією одного або багатьох шляхів).

Ще однією перевагою у виборі ПЛІС у порівнянні з іншими апаратними рішеннями є те, що ПЛІС поєднує в собі високу продуктивність та гнучкість програмованої логіки. Це дозволяє рішенням на ПЛІС аналізувати сигнал достатньо високої частоти в реальному часі без довготривалої розробки та виробництва спеціалізованого апаратного комплексу, а додатковою перевагою є відносна невелика ціна такого рішення. У свою чергу недоліком ПЛІС є обмежена кількість внутрішніх ресурсів та блоків, тому реалізація ефективних, апаратно-орієнтованих алгоритмів є вкрай важливою. Зокрема важливо правильно обрати точність і метод округлення результатів, тому що ширина вхідних, проміжних та вихідних результатів значуще впливає на потребу у внутрішній пам'яті ПЛІС та складності реалізації множення.

Зазвичай виробники ПЛІС надають готові реалізації ШПФ, але також можливе написання своєї реалізації. Використання готових блоків та написання власної реалізації ШПФ на ПЛІС мають свої переваги та недоліки. Готові блоки часто оптимізовані виробником для характеристик конкретних пристроїв. Вони можуть забезпечити високу продуктивність та ефективне використання ресурсів. Крім того, використання готових блоків спрощує розробку, скорочує її час та ціну, зменшує вірогідність помилок.

З іншого боку, написання власної реалізації може бути більш універсальним рішенням. Розробник може створити реалізацію, яка працює на ПЛІС різних виробників без значних змін. Оптимізація для конкретної задачі також може бути легше досягнута власноруч, що може призвести до вищої ефективності та швидкодії.

Окремо потрібно відзначити порядок результуючих відліків обчислення ШПФ. Значна частина ШПФ реального часу дають результати у зворотному порядку бітів, а для перетворення порядку в натуральний порядок потрібне впровадження тимчасового зберігання відліків перед їх видачею на вихід. Готові блоки зазвичай мають вбудовану можливість такої реалізації, але вона вносить додаткову затримку в алгоритм, а також потребує великого обсягу додаткової внутрішньої пам'яті ПЛІС, що є критичним для більшості задач.

В цьому дослідженні для реалізації обрано актуальну ПЛІС Xilinx Artix-7, а в якості обчислювача ШПФ - готовий блок з потоковою структурою та зворотнім порядком вихідних відліків, що працює на частоті дискретизації даних. Цей варіант реалізації дає найменшу затримку результату на виході та потребує найменшу кількість блоків внутрішньої пам'яті ПЛІС. Кількість інших ресурсів для такої реалізації незначуще більша.

### **Висновок.**

Під час дослідження розглянуті підходи до побудови апаратної реалізації ШПФ на сучасних ПЛІС. Було обрано параметри реалізованого потокового перетворення для досягнення найменшої затримки і найменшої необхідної кількості внутрішньої пам'яті. Для економії внутрішньої пам'яті ПЛІС порядок відліків на виході не є натуральним, а є біт-зворотнім, що взято до уваги при подальшому використанні результатів ШПФ.

### **Список використаних джерел:**

1. Cooley J. W., Tukey J. W. *An algorithm for the machine calculation of complex Fourier series*. Mathematics of Computation. 1965. Vol. 19, no. 90. P. 297. URL: <https://doi.org/10.1090/s0025-5718-1965-0178586-1>.
2. *Programs for digital signal processing* / ed. by IEEE Acoustics, Speech, and Signal Processing Society. Digital Signal Processing Committee. New York : IEEE Press, 1979. 600 p.
3. Oppenheim A. V., Schaffer R. W. *Discrete-Time Signal Processing*. Pearson Education, Limited, 2009. 1120 p.
4. *A 1 Million-Point FFT on a Single FPGA* / H. Kandors et al. IEEE Transactions on Circuits and Systems I: Regular Papers. 2019. Vol. 66, № 10. P. 3863–3873. URL: <https://doi.org/10.1109/tcsi.2019.2918403>