https://repository.uaeh.edu.mx/revistas/index.php/xikua/issue/archive
XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan
Publicación semestral, Vol. 13, No. 26 (2025) 25-29
Xikua
ISSN: 2007-4948

Cómputo reconfigurable en la aceleración de herramientas de Mapeo de ADN.

Reconfigurable Computing in Accelerating DNA Mapping Tools.

Daniel Pacheco-Bautista [a], Francisco Aguilar-Acevedo [b], Yuliana García-Amaya [c], Efraín Dueñas-Reyes [d]

Abstract:

The extraction of the genetic code from the cell nucleus is known as DNA sequencing and is a current topic of global relevance given its importance. One of the steps in sequencing consists of aligning millions of short sequences to complete genomes of several hundred million nucleotides using complex computer programs called DNA Aligners or Mappers. However, their run times are still very long compared to the speed at which NGS machines produce sequences, making them the bottleneck of the overall DNA analysis process. This paper systematically reviews the use of reconfigurable computing using FPGAs to accelerate short-read DNA aligning tools, identifying the main algorithms that have been efficiently implemented within each of the stages of such tools, as well as the acceleration factors achieved. It is hoped that this article will serve as a basis for future related research.

Keywords:
Reconfigurable computing, DNA mapping tools, FPGAs, Bioinformatics.

Resumen:

La extracción del código genético a partir del núcleo celular es conocido como secuenciación de ADN y es un tema de investigación actual relevante a nivel mundial dada la importancia que representa. Uno de los pasos de la secuenciación consiste en alinear millones de secuencias cortas a genomas completos de varios cientos de millones de nucleótidos mediante complejos programas informáticos denominados Alineadores o Mapeadores de ADN. No obstante, sus tiempos de ejecución aún son muy largos comparados con la velocidad en que las máquinas NGS producen las secuencias, lo que los convierte en el cuello de botella del proceso general de análisis de ADN. En este artículo se revisa sistemáticamente el uso del cómputo reconfigurable utilizando FPGAs para acelerar herramientas de alineación de lecturas cortas de ADN, identificando los principales algoritmos que se han podido implementar de manera eficiente dentro de cada una de las etapas de tales herramientas, así como los factores de aceleración logrados. Se espera que este artículo sirva de base a futuras investigaciones relacionadas.

***Palabras Clave**:*
Cómputo reconfigurable, Herramientas de mapeo de ADN, FPGAs, Bioinformática.

Introducción

El ADN es una macromolécula muy larga en forma de doble elipse contenida principalmente en los cromosomas dentro del núcleo celular, la cual codifica mediante cuatro bases nitrogenadas, Adenina (A), Citosina(C), Guanina (G) y Timina (T), la forma de desarrollo, evolución y reproducción de un ser vivo. La extracción de tal código a partir del núcleo celular es conocido como secuenciación de ADN y es un tema de investigación actual relevante a nivel mundial. La secuenciación de ADN tiene múltiples aplicaciones, por ejemplo, resulta fundamental en el desarrollo y pruebas de nuevos fármacos y vacunas, la identificación de virus, bacterias y enfermedades como el cáncer en el ser humano, la medicina genómica, la identificación de identidad, el desarrollo de nuevos productos agrícolas, entre otras [1-2].

[a] Autor de Correspondencia, Universidad del Istmo, https://orcid.org/0000-0001-5840-9798, Email: dpachecob@bianni.unistmo.edu.mx

[b] Universidad Anáhuac Puebla, https://orcid.org/0000-0002-5248-3230, Email: facevedo@anahuac.mx

[c] Universidad del Istmo, https://orcid.org/0009-0004-1787-7439, Email: yuliana@sandunga.unistmo.edu.mx

[d] Universidad del Istmo, https://orcid.org/ 0009-0006-5639-766X, Email: eduenas@sandunga.unistmo.edu.mx

Fecha de recepción: 29/03/2025, Fecha de aceptación: 30/04/2025, Fecha de publicación: 05/07/2025

DOI: https://doi.org/10.29057/xikua.v13i26.14844

El proceso de secuenciación de ADN es realizado por sofisticados equipos denominados secuenciadores de siguiente generación (NGS), capaces de procesar material genómico a velocidades impresionantes debido al uso masivo de paralelismo. Sin embargo, el problema fundamental radica en que estos equipos solo pueden secuenciar los genomas por fragmentos, produciendo en su salida millones de códigos pequeños denominados lecturas, que posteriormente deben ensamblarse para reconstruir el código genético completo, como se ve en la Figura 1. El ensamble puede realizarse únicamente comparando las lecturas entre sí, en cuyo caso recibe el nombre de ensamble de Novo. Sin embargo, es más recurrente tomar como referencia el código de un genoma de la misma especie o similar secuenciado previamente, en cuyo caso recibe el nombre de alineación o mapeo.

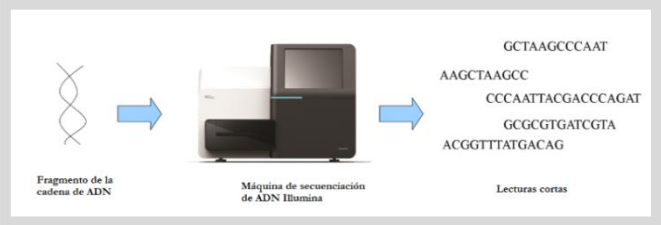

Figura 1. Secuenciación de ADN utilizando máquinas NGS

Dada la importancia del tema, en la última década se han desarrollado múltiples programas de alineación, los cuales utilizan diferentes heurísticas y algoritmos multietapa y son ejecutados en computadoras equipadas con varios procesadores con múltiples núcleos de procesamiento. Ejemplos representativos de estas herramientas son Bowtie [3], PASS [4], RazerS [5], mrFAST [6], GASSST [7], BWA [8], MOSAIK [9], Soap3-dp [10], deSALT [11] y conLSH [12].

En esencia, la meta principal de la alineación es localizar segmentos (de entre 50 y 30 nucleótidos) dentro de la cadena de referencia (de millones de nucleótidos) que son similares a la lectura, mientras permiten a lo mucho *e* ediciones (eliminaciones, inserciones o mutaciones de nucleótidos). Para lograr lo anterior, la mayoría de los programas de alineación recientes utiliza una estrategia de 3 pasos denominada siembra-filtra-extiende. La estrategia está basada en la observación de que, para un mapeo correcto, la lectura corta y el segmento del genoma de referencia al cual debe alinearse, deben compartir algunas regiones pequeñas de apareamiento exacto o aproximado, esas regiones cortas compartidas, las cuales indican alta similitud entre las cadenas reciben el nombre de semillas, *k*-mers o *q*-grams.

En la etapa de siembra, el alineador obtiene las semillas a partir de la lectura y las localiza en el genoma de referencia para posicionarla globalmente en forma aproximada. El proceso de alineación implica la creación previa de un índice enorme a partir del genoma de referencia, para facilitar la consulta rápida y eficiente de toda la secuencia, buscando siempre que se utilice la mínima cantidad de memoria. En este paso pueden utilizarse diversos algoritmos, no obstante, la mayoría de los programas actuales de alineación utilizan tablas hash, árboles de sufijos o índices de FM [12].

En la etapa de filtrado, el alineador utiliza diferentes heurísticas para examinar en forma rápida la similaridad entre la lectura y cada una de las posiciones encontradas en el paso previo, descartando aquellas con posibilidades nulas de alineación con *e* ediciones, esta etapa es muy importante puesto que al eliminar gran cantidad de posiciones que no representan una alineación real, disminuye el trabajo de la etapa de extensión, siendo esta la más absorbente en términos computacionales.

La etapa de extensión revisa cuidadosamente todas las formas posibles de alineación entre la lectura y segmentos de la cadena de referencia extraídos desde las posiciones que han superado la etapa de filtrado, reportando las mejores. Debido a la exactitud que se requiere, estos algoritmos están basados principalmente en programación dinámica, lo cual los hace muy complejos tanto espacial como temporalmente. Los dos algoritmos más comunes encontrados en esta etapa son el Smith-Waternan [14] y el Needleman-Wush [15], en su primera fase ambos algoritmos llenan mediante una función de actualización, una matriz con puntuaciones de alineaciones parciales entre segmentos de las cadenas, luego en una segunda fase, a partir de la mejor puntuación se realiza un recorrido en retroceso que permite determinar la forma de alineación entre ambas cadenas, este último paso es conocido como *bactracking*.

A pesar de los esfuerzos en el desarrollo de estas herramientas, los tiempos de ejecución están muy por arriba de los de las máquinas NGS que producen las lecturas cortas que deben alinearse. Por ejemplo, un secuenciador NovaSeq X plus de la empresa Illumina, es capaz de secuenciar 128 genomas humanos en tan solo dos días, mientras que una herramienta de alineación en el estado del arte, corriendo en un sistema de cómputo de alto rendimiento, necesita varias horas y aun días para alinear un solo genoma humano [16]. Esto ha motivado al uso de plataformas de cómputo alterno como las Unidades de Procesamiento Gráfico (GPUs), las arquitecturas de Procesamiento en Memoria (PIM) y los Arreglos de Compuertas Programables en el Campo (FPGAs).

Los FPGAs son semiconductores que contienen millones de bloques lógicos combinacionales (CLBs), interconexiones programables, módulos de entrada/salida, multiplicadores, procesadores digitales de señales (DSPs), etc., que pueden personalizarse en forma muy eficiente para realizar una función específica requiriendo menor consumo energético en relación a las GPUs. El flujo de trabajo del FPGAs es más complejo en relación al que corresponde a las GPUs y PIM, requiriendo lenguajes de descripción de hardware y conocimientos solidos de electrónica de conmutación. No obstante, las herramientas de síntesis de alto nivel (HLS) han avanzado considerablemente, permitiendo la programación del FPGA utilizando lenguajes de alto nivel convencionales como C y C++, lo cual simplifica el proceso y disminuye el tiempo de colocación en el mercado.

Objetivo

El objetivo de este trabajo es revisar el uso de cómputo reconfigurable basado en FPGAs en la aceleración de herramientas de alineación o mapeo de ADN.

Metodología y proceso de desarrollo

Este trabajo corresponde a una investigación documental donde se ha aplicado el proceso de revisión sistemática de la Literatura. El proceso consistió de tres partes: Planeación, ejecución y análisis de resultados.

Planeación de la revisión. En esta fase se definieron las preguntas de investigación, las cual fueron las siguientes: ¿Es recurrente el uso de tecnologías de cómputo reconfigurable en la aceleración de programas de alineación?, ¿En qué etapas dentro del programa se ha aplicado tal tecnología? y ¿Qué factores de aceleración se han conseguido? ¿Cuál ha sido la tendencia al aplicar esta tecnología? así mismo se determinó el motor de búsqueda a utilizar, siendo éste Google Académico. La elección del mismo se debe a que este motor de búsqueda proporciona un enfoque de indexación inclusivo con cobertura de documentos científicos y académicos, más completa que otras bases de datos más especializadas como SCOPUS o WoS (web of Science). Las cadenas de búsqueda fueron: a) “Cómputo reconfigurable” + “Alineación de ADN”, b) “Cómputo reconfigurable” + “Mapeo de ADN”, c) “Aceleración Hardware” + “Alineación de ADN” y d) “Aceleración hardware” + “Mapeo de ADN”, e) “Alineación de ADN” + “FPGAs”, f) “Mapeo de ADN” + “FPGAs”.

Ejecución de la revisión. A partir de más de 200 resultados obtenidos inicialmente, se identificaron y seleccionaron estudios primarios eliminando falsos positivos a partir del título e información proporcionada por la base de datos, aplicando entonces criterios de inclusión considerando la lectura del resumen, eliminando las fuentes duplicadas y eligiendo las fuentes que aseguraron la calidad, el proceso resulto en un conjunto de 32 artículos. La búsqueda de las cadenas se realizó en inglés y en español configurando los siguientes filtros: desde el año 2003 hasta el 2023, ordenados por relevancia, sin incluir patentes ni citas.

Análisis de resultados. En esta fase se revisaron a profundidad cada uno de los artículos que pasaron la fase depuración en el paso previo, obteniendo las respuestas a las preguntas de investigación y las conclusiones, mismas que se muestran a continuación.

Resultados

Las publicaciones revisadas a profundidad, muestran la recurrente aplicación del cómputo reconfigurable dentro de los programas de alineación, utilizando no solo el proceso de diseño clásico que utiliza Lenguajes de Descripción de Hardware, sino también la tecnología de Síntesis de Alto Nivel. Los FPGAs se han utilizado en las tres etapas de los programas de alineación, sin embargo, la mayoría se han centrado en la aceleración de la etapa de extensión, la cual puede absorber cerca del 90% del tiempo de ejecución del programa cuando no se incluye una etapa de filtrado [17]. La estrategia está basada principalmente en el cálculo simultáneo de las celdas en las anti diagonales de la matriz de puntuaciones, mediante arreglos sistólicos, disminuyendo la complejidad temporal de O(mn) a $O(m+n)$. Cada celda en la matriz de puntuaciones depende de sus celdas vecinas superior, izquierda y superior-izquierda lo que limita el cálculo de filas o columnas simultáneamente. Sin embardo después de calcular una celda, por ejemplo A(1,1), es posible calcular simultáneamente las celdas A(1,2) y A(2,1).

Los primeros trabajos reportados estaban limitados al cálculo del algoritmo SW o NW con el modelo de penalidad lineal de espacios [18-20], sin embargo, rápidamente se reportaron diseños con el modelo con penalidad Afín de espacios [21-24], que se adapta de una mejor manera al proceso evolutivo de las especies. Adicionalmente, se han buscado diseños más flexibles, que permitan realizar alineaciones con diferentes parámetros de entrada e incluso permitiendo tanto alineaciones globales como locales [25]. Sus resultados experimentales muestran que comparados con implementaciones software contemporáneas logran acelerar la etapa en factores entre 50 y 100x. Para obtener mejoras adicionales, los diseños más actuales realizan también el backtracking de la matriz de puntuaciones en el FPGA [26-27], en lugar de dejárselo al procesador

anfitrión, aunque es un proceso simple esto evita la transmisión de grandes cantidades de información entre el FPGA y el CPU que funciona como anfitrión. Por ejemplo, FPGASW [27] es una reciente implementación del algoritmo SW con backtracking y el modelo de penalidad afín de espacios, capaz de procesar lecturas ultra-largas debido a su estrategia de partición de tareas. FPGASW obtiene un factor de aceleración de 23.5x y 10.2x en relación a los alineadores basados en GPU SW-CUDA y CUDASW++ respectivamente. Las pruebas en ese trabajo se realizaron utilizando el FPGA Xilinx XC7VX485T y la NVidia Geforce GTX680 GPU. De acuerdo a sus autores FPGASW tiene un ahorro en el consumo de potencia cercano al 75% en relación a las implementaciones GPU.

Entre los trabajos que han incursionado en el uso de técnicas de síntesis de alto nivel, se encuentra SWIFOLD [28], una implementación FPGA del algoritmo SW para lecturas de ADN sin restricciones de tamaño, en un Intel's Arria 10 FPGA. El diseño gana portabilidad entre plataformas y disminuye considerablemente el tiempo de desarrollo, compitiendo en razón de procesamiento con alineadores basados en GPU tal como CUDAling utilizando GPUs en el estado de arte, ejecutándose apenas 1.6x más lento en promedio.

La etapa de filtrado también se ha beneficiado de su aceleración a partir de FPGAs, debido al uso de operaciones masivas simples, como desplazamientos, decrementos, comparaciones y concatenaciones, ideales para realizar en estos dispositivos. GateKeeper [29] es el primer diseño para acelerar la etapa de pre-alineación utilizando FPGAs. Utilizando un solo FPGA GateKeeper logra factores de aceleración de 90x y 130x en relación a las principales técnicas de filtrado en software Adjacency Filter and Shifted Hamming Distance (SHD), respectively. La adición de GateKeeper a la herramienta de alineación mrFAST [6] reduce el tiempo de la etapa de extensión por un factor de 10x. Otras implementaciones muy atractivas de algoritmos de prefiltrado en FPGAs son Shouji [30] y SneakySnake [31], las cuales mejoran en varios ordenes de magnitud la exactitud del filtrado de alineación en relación a GateKeeper. En [32] se implementan y optimizan varios algoritmos de pre-alineación utilizando síntesis de alto nivel para expandir su portabilidad a sistemas soportando openCL runtime.

Respecto a la etapa de siembra, son pocos los trabajos que reportan la aceleración de los algoritmos dentro de este grupo. Dentro de estos, en [34] se desarrolla una arquitectura reconfigurable en tiempo de ejecución basada en FPGA que implementa una versión modificada del algoritmo índices de FM. La arquitectura soporta alineaciones tanto exactas como aproximadas hasta con *e* ediciones. Sus creadores utilizan la tarjeta MPC-X2000 que incluye 8 FPGAs Altera Stratix-V, y reportan factores de aceleración de 28x en relación a Bowtie2 corriendo con 16 hilos sobre un procesador Intel Xeon E5-2640 y 9x en relación a Soap3-dp corriendo sobre una GPU NVIDIA Tesla C2070. En [35] se presenta otra arquitectura que acelera el algoritmo índice de FM, en esta se proponen varias técnicas que logran reducir el número de accesos a memoria hasta en un 60% y el número de operaciones en un 50% para aumentar la razón de procesamiento. Los beneficios de las técnicas, así como el uso de un FPGA (el Xilinx Alveo U250) le permite alinear más de 1000 millones de lecturas en tan solo 35.4 minutos.

Conclusiones

El resultado de esta investigación permitió definir el panorama general del uso del cómputo reconfigurable en la aceleración de los programas de alineación. Esta tecnología se aplicado de manera recurrente en las últimas dos décadas siendo muy utilizada principalmente en la aceleración de la etapa de extensión, pero también ampliamente usada en la implementación de algoritmos de filtrado donde se han logrado factores de aceleración superiores a 100x en relación a filtros implementados en software en el estado del arte, y en menor proporción en la etapa de siembra. Los principales algoritmos acelerados mediante esta tecnología incluyen: Smith Waterman, Needleman Wush, índices de FM, filtros de adyacencias, distancia de haming desplazada y Shouji. Se ha identificado también el uso cada vez mayor de las tecnologías HLS y se prevé que en un futuro cercano tanto la etapa de filtrado como la etapa de extensión puedan ser aceleradas en conjunto disminuyendo el flujo de datos entre el FPGA y el procesador anfitrión.

Referencias

[1] Satam H, Joshi K, Mangrolia U, Waghoo S, Zaidi G, Rawool S, ... & Malonia S K. (2023). Next-generation sequencing technology: current trends and advancements. *Biology*, *12*(7), 997.

[2] Logsdon, GA, Vollger, MR, Eichler EE. (2020). Long-read human genome sequencing and its applications. *Nature Reviews Genetics*, *21*(10), 597-614.

[3] Langmead B, Trapnell C, Pop M, Salzberg SL. Ultrafast and memory-efficient alignment of short DNA sequences to the human genome, Genome Biol. 10(3) (2009) R25. 23

[4] Campagna D *et al*. PASS: a program to align short sequences, *Bioinformatics* **25**(7) (2009) 967–968. 25

[5] Weese D, Emde AK, Rausch T, Döring A, Reinert K. RazerS—fast read mapping with sensitivity control, *Genome Res.* **19**(9) (2009) 1646–1654. 26

[6] Alkan C *et al*., Personalized copy number and segmental duplication maps using next-generation sequencing, *Nat. Genet.* **41**(10) (2009) 1061–1067. 27

[7] Rizk G, Lavenier D. GASSST: global alignment short sequence search tool, *Bioinformatics* **26**(20) (2010) 2534–2540. 28

[8] Li H, Durbin R. Fast and accurate long-read alignment with Burrows–Wheeler transform, *Bioinformatics* **26**(5) (2010) 589–595. 29

[9] Lee WP *et al*., MOSAIK: a hash-based algorithm for accurate next-generation sequencing short-read mapping, *PloS One* **9**(3) (2014) e90581. 31

[10] Luo R, Wong T, Zhu J, Liu C M, Zhu X, Wu E, Lee LK, Lin H, Zhu W, Cheung DW et al. "Soap3-dp: fast accurate and sensitive gpu-based short read aligner", *PloS one*, vol. 8, no. 5, pp. e65632, 2013.

[11] Liu B *et al*., deSALT: fast and accurate long transcriptomic read alignment with de Bruijn graph-based index, *Genome Biol.* **20**(1) (2019) 274. 32

[12] Chakraborty A, Bandyopadhyay S. conLSH: Context based Locality Sensitive Hashing for mapping of noisy SMRT reads, *Comput. Biol. Chem.* **85** (2020) 107206. 33

[13] Ferragina P, Manzini G, Mäkinen V, Navarro G. (2004, October). An alphabet-friendly FM-index. In *International Symposium on String Processing and Information Retrieval* (pp. 150-160). Berlin, Heidelberg: Springer Berlin Heidelberg.

[14] Smith TF, Waterman MF. Identification of common molecular subsequences, *J. Mol. Biol.* **147**(1) (1981) 195–197.

[15] Needleman SB, Wunsch CD. A general method applicable to the search for similarities in the amino acid sequence of two proteins, *J. Mol. Biol.* **48**(3) (1970) 443–453.

[16] Marçais G *et al*., MUMmer4: A fast and versatile genome alignment system, *PLoS Comput. Biol.* **14**(1) (2018) e1005944.

[17] Xin H et al., Accelerating read mapping with FastHASH, BMC Genom. 14(1) (2013) S13.

[18] Puttegowda K *et al*., A run-time reconfigurable system for gene-sequence searching, in *Proc 16th Int Conf VLSI Design*, New Delhi, India (2003), pp. 561–566.

[19] Yu CW, Kwong KH, Lee KH, Leong PHW, A Smith-Waterman systolic cell. In *Proc. Field Programmable Logic and Application: 13th International Conference, Lisbon, Portugal (2003),* pp. 375-384.

[20] Caffarena G, Pedreira C, Carreras C, Bojanic S, Nieto-Taladriz O. FPGA acceleration for DNA sequence alignment, *J. Circuits Syst. Comput.* **16**(02) (2007) 245–266.

[21] Oliver TF, Schmidt B, Maskell DL. Reconfigurable architectures for bio-sequence database scanning on FPGAs, *IEEE Trans Circuits Syst II: Express Br.* **52**(12) (2005) 851–855.

[22] Van-Court T, Herbordt MC, Families of FPGA-based accelerators for approximate string matching, *Microprocess Microsyst* **31**(2) (2007) 135–145.

[23] Jiang X, Liu X, Xu L, Zhang P, Sun N. A reconfigurable accelerator for smith–waterman algorithm, *IEEE Trans. Circuits Syst. II: Express Br.* **54**(12) (2007) 1077–1081.

[24] Li IT, Shum W, Truong K. 160-fold acceleration of the Smith-Waterman algorithm using a field programmable gate array (FPGA), *BMC Bioinform.* **8** (2007) 185.

[25] Benkrid K, Liu Y, Benkrid A. A highly parameterized and efficient FPGA-based skeleton for pairwise biological sequence alignment, *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **17**(4) (2009) 561–570.

[26] Pacheco-Bautista D, Carreño-Aguilera R, Aguilar-Acevedo F, Algredo-Badillo I. Bit-Vector-Based Hardware Accelerator for DNA Alignment Tools, *J. Circuits Syst. Comput.* **30**(5) (2021) 2150087.

[27] Fei X, Dan Z, Lina L, Xin M, Chunlei Z. FPGASW: accelerating large-scale Smith–Waterman sequence alignment application with backtracking on FPGA linear systolic array, *Interdiscip. Sci.* **10**(1) (2018) 176–188.

[28] Rucci E *et al*., M, SWIFOLD: Smith-Waterman implementation on FPGA with OpenCL for long DNA sequences, *BMC Syst. Biol.* **12**(5) (2018) 43–53.

[29] Alser M *et al*., GateKeeper: a new hardware architecture for accelerating pre-alignment in DNA short read mapping, *Bioinformatics* **33(21)** (2017) 3355–3363.

[30] Alser M, Hassan H, Kumar A, Mutlu U, Alkan C. Shouji: a fast and efficient pre-alignment filter for sequence alignment, *Bioinformatics* **35**(21) (2019) 4255–4263.

[31] Alser M, Shahroodi T, Gómez-Luna J, Alkan C, Mutlu. SneakySnake: a fast and accurate universal genome pre-alignment filter for CPUs, GPUs and FPGAs, *Bioinformatics* **36**(22–23) (2020) 5282-5290.

[32] Castells-Rufas D., Marco-Sola S, Moure JC, Aguado Q, Espinosa A. FPGA Acceleration of Pre-Alignment Filters for Short Read Mapping With HLS, *IEEE Access* **10** (2022) 22079–22100.

[33] Arram J, Kaplan T, Luk W, Jiang P. (2016). Leveraging FPGAs for accelerating short read alignment. *IEEE/ACM transactions on computational biology and bioinformatics*, *14*(3), 668-677.

[34] Yang CH, Wu YC, Chen YL, Lee CH, Hung JH, Yang CH. (2023). An FM-Index Based High-Throughput Memory-Efficient FPGA Accelerator for Paired-End Short-Read Mapping. *IEEE Transactions on Biomedical Circuits and Systems*.