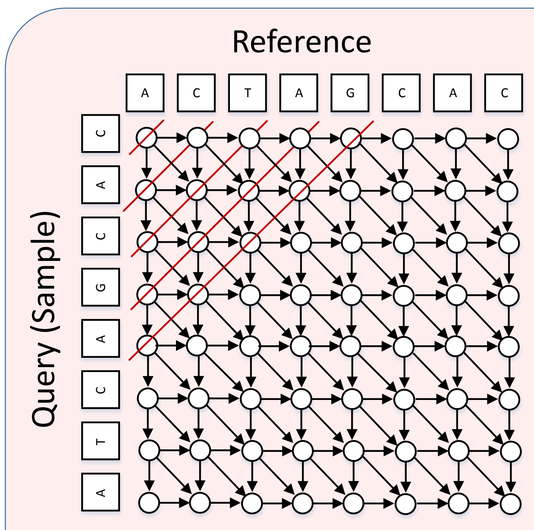
FPGA Based OpenCL Acceleration of Genome Sequencing Software

Ashish Sirasao, Elliott Delaye, Ravi Sunkavalli, Stephen Neuendorffer
Xilinx Inc., 2100 Logic Drive, San Jose, CA

Introduction

The Smith-Waterman algorithm [1], which produces the optimal pairwise alignment between two sequences of proteins or nucleotides, is frequently used in genomics but computationally expensive. In this work an efficient and scalable implementation of the Smith-Waterman algorithm is demonstrated for nucleotides using OpenCL and implemented on a Xilinx Virtex-7 FPGA PCIe accelerator card. A linear systolic array is implemented for the wavefront computation with local memory caching and multiple simultaneous queries. A performance of 77 GCUPS was achieved 30% faster than GPU implementations and 290% faster than a 12-core CPU and 60-core coprocessor. Architectural choices are discussed to achieve the results.

Smith-Waterman Algorithm



$$H_{i,j} = \max \begin{cases} H_{i-1,j-1} + s(a_i, b_i) \\ H_{i-1,j} - W_k \\ H_{i,j-1} - W_l \\ 0 \end{cases}$$

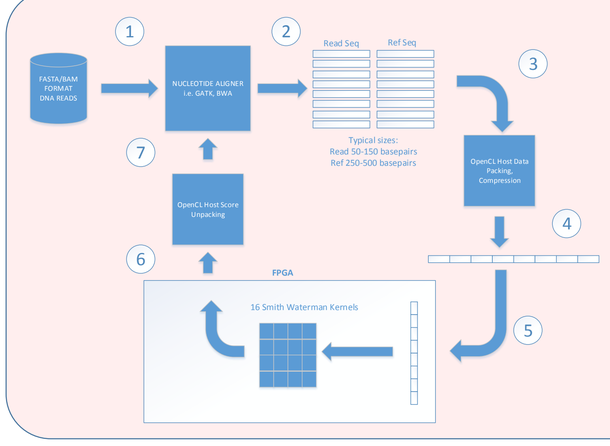
* Goal: Find the optimal substring match using score values for matches and mismatches of nucleotides of proteins
* Performance measure in GCUPS: Billion Cell Updates/Sec

STEPS:
For each cell *i,j*
1) **Compare characters at Q(*i*),R(*j*)**
2) **Add match/mismatch score to Northwest**
3) **Subtract mismatch score from North, West**
4) **Store max value at $H_{i,j}$**

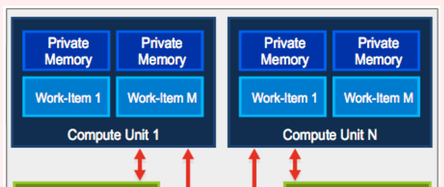
Results of the Smith-Waterman calculation typically include max value of all $H_{i,j}$ scores along with the location (*i,j*). Some implementations will include backtracking which can produce an alignment showing where the two substrings have differences, i.e.

ACT–AGCAC
A–TCAGCAC

FPGA Acceleration Platform

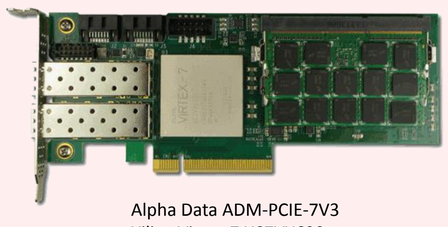


Genome sequencing software typically starts with samples and reference files (1) and using various techniques can identify areas of the genomes where initial alignment may have been sub-optimal. These initial alignments may use algorithms such as the Burrows-Wheeler Transform. The software then produces the reference and sample data (2) for these suspect areas for full Smith-Waterman alignment. Our acceleration platform begins at this step (3) where the sample and reference data have been identified and are sent through the OpenCL framework (3). The sample and references of various alignments are combined to minimize memory transfer overhead to the global memory area on the FPGA accelerator card (4,5). Inside the OpenCL compute unit, 16 sample/reference pairs are run simultaneously and results sent back to the OpenCL host (6). Alignment locations can be used by the alignment software for further sequencing.



One of the key aspects of achieving performance was managing the various memory systems inside the FPGA. The OpenCL Platform Model separates memory into global, local and private but within private memory, Xilinx FPGAs have three different types. During step 4 from above, samples are transferred to the accelerator card DDR3. In step 5, the FPGA begins a burst DDR3 read and copies all samples/reference pairs into RAMB blocks. Within the FPGA, each of our 16 Smith-Waterman kernels then reads from RAMB into LUTRAMs. Finally to perform the PE computation, all intermediate values are stored in Flip-Flops.

Type	KBits	Memory Update rate @200MHz in Tb/s	Usage
FF	538	105.25	Internal PE scores values
LUTRAM	7078	21.60	Store Sample/Reference strings per PE
RAMB	34380	13.12	Read/Write Between DDR3 and FPGA
DDR3	8388608	0.08	Read/Write between host



Alpha Data ADM-PCIE-7V3
Xilinx Virtex-7 XC7VX690t

```
for(i in rows) {
  __attribute__((xcl_pipeline_loop))
  for(j in MAXPE) {
    executePE(i, j, &myPE[j], &myPE[j-1]);
  }
}
```

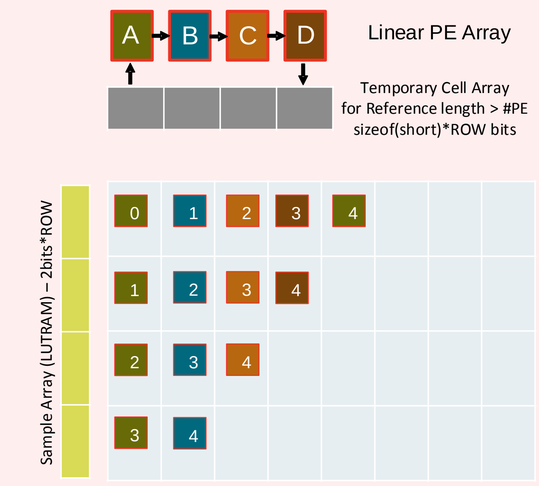
Attribute to allow full anti-diagonal of PEs to run in parallel

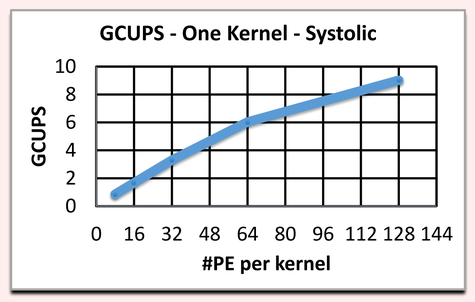
Architectural Evaluations

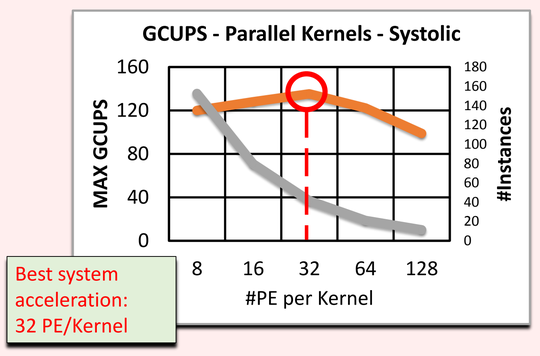
Approaches to Smith-Waterman on SIMD processors include striped approaches, tiled approaches, intra and inter task parallelism. Algorithm selection in OpenCL let us try various approaches. The Striped Smith-Waterman [2] approach was prototyped in OpenCL on the FPGA and compared with the linear systolic array. The linear systolic array was approximately 5.3x faster. The architecture chosen for FPGA must be computational efficiency in that every clock cycle must maximize the number of cells being updated. Striped Smith-Waterman while performing well on SIMD processors compared to other algorithms has the drawback of separating horizontal and vertical dependencies. During the vertical dependency calculation, the vector units become idle losing efficiency and performance.

Algorithm (FPGA)	GCUPS
Striped Smith-Waterman [2]	0.6
Linear Systolic (w/32 PEs)	3.2

Systolic solutions to Smith-Waterman have various implementations from 2-D mesh to linear arrays. Linear arrays model the wavefront nature of a single query. The number of processing elements on the wavefront can vary from 1 up to the maximum diagonal. Performance on an FPGA is directly tied to area efficiency measured by chip area used per kernel. Single kernel performance, both queries per second and GCUPS, increases with number of PEs per kernel however total system performance shows a maximal GCUPS at 32 PEs per kernel. At 32 PEs per kernel, there was a balance between the idle time of larger number of PEs and the area overhead of kernel communication.



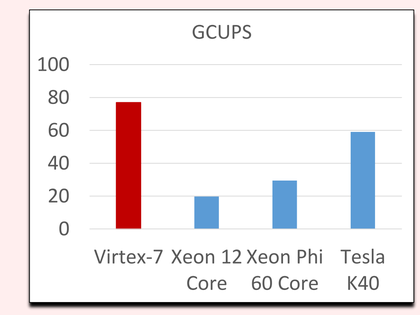


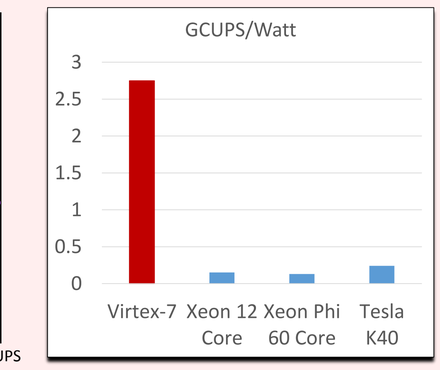


Performance Results

Platform Achitecture	Device	GCUPS	Watts	GCUPS/Watt
Systolic FPGA / OpenCL	Virtex-7 690T	77.0	28	2.8
SSW [2] / SSE2	Intel® Xeon E5-2697 12 core	19.7	130	0.15
SWAPHI-LS [3] / AVX-512	Intel® Xeon Phi 5110P 60 core	29.5	225	0.13
GSWABE [4] / CUDA	nVidia® Tesla K40 2880 core	59.1	245	0.24
	FPGA vs Xeon E5-2697	**3.9x**	**0.22x**	**18.7x**
	FPGA vs Xeon Phi 5110P	**2.6x**	**0.12x**	**21.5x**
	FPGA vs Tesla K40	**1.3x**	**0.11x**	**11.6x**

* Power Based on TDP

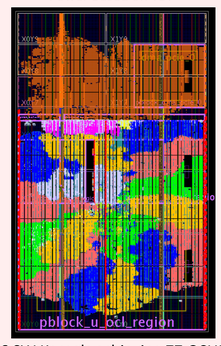




Future Optimizations:
- Adjusting score bitwidths
- Improving DDR3 memory interface
- Reducing kernel area
- Inter-Task vs Intra-Task Parallelism

Hardware:
- Xilinx Virtex-7 XC7VX690T
- Alpha Data ADM-PCIE-7V3

Software:
- OpenCL in Xilinx SDAccel 2015.1



16 SW Kernels achieving 77 GCUPS

Conclusions

- OpenCL can model systolic arrays on FPGAs previously only accessible by writing RTL
- SIMD Striped Smith-Waterman on FPGA is inferior to systolic array implementation
- Memory Bandwidth internal and external to accelerator was measure and optimized
- Significant performance increase achieved compared to published work using a single device and order of magnitude better GCUPS/watt as a power efficiency metric
- System performance on the FPGA requires optimizing an area/throughput curve for choosing kernel sizes and balancing

Future Work:
Smith-Waterman is one of many algorithms used in platforms and tools such as GATK/Gamgee, BWA-MEM, Bowtie2. Similar optimization techniques will be required to accelerate the full genomics workflow.

XILINX
ALL PROGRAMMABLE™

References: [1] Smith, T.F. and Waterman, M.S. 1981, Identification of common molecular subsequences. *J. Mol. Biol.*, 147, 195-197 [2] Zhao, M., et al., 2013, SSW library: an SIMD Smith-Waterman C/C++ library for use in genomic applications. PLoS One, vol. 8, pp. e82138, 2013. [3] Liu, Y., et al., 2014, SWAPHI-LS: Smith-Waterman Algorithm on Xeon Phi Coprocessors for Long DNA Sequences, *IEEE International Conf. on Cluster Computing 2014*, 257-265 [4] Liu, Y., Schmidt, B., 2014, GSWABE: faster GPU-accelerated sequence alignment for optimal alignment retrieval for short DNA sequences, Concurrency and Computat.: Pract. Exper. 2015; 27:958-972