1.A Systolic array based architecture:

Features

Not for Blastn

- Implemented in VHDL
- Algorithm
    - Finding hits-Searches for exact matches of those k words on FPGA
    - Ungapped Extension- Do ungapped extension using BLOSUM50
    - Gapped Extension- Using Needleman-Wunsh
- Architecture consists of three layers
    - Top layer-i7 processor and 4GB RAM
    - Middle Layer- two 1GB SDRAMs- subject sequence and HSP list
    - Basic layer- FPGA Virtex 5 embedded on Xilinx ML509
- Hits Finder array detects multiple hits at a time
- Hits combinational logic combines overlapping hits
- PCIe-500MB/s throughput
- Implemented for query lengths of 1024, 2048, 3072

Claims

- Implement every step of Blast on FPGA to avoid interface issue and save execution time
- Use NW because of its ability to find optimal global gapped alignment
- Using WPRBS at most one hit can be found in one clock cycle
- Memory Constraints
    - Mercury Blastn built a hash table from query
        - An accelerator checks words in S db against hash table from Q
        - Hash table stored in external SRAM which create timing issue like long pipeline cycle time

        - Proposes a two hit method for acceleration
    - FPGA/Flash
        - Database is formatted as an index- every word and its position in sequence and its adjacent environment so ungapped alignments could be computed simultaneously, avoiding random accesses
        - Size of index is very large
    - Multiengines
        - Adopted 64 identical computing units on single chip
- Proposed architecture detects multiple hits in one clock cycle
- Tree Blast could only have query size of 600 due to BRAM restrictions
- FPGA based accelerators have complex processing unit –
    - require more registers to get match information
    - Data stream needs to be shut down to record hit addresses
- Systolic array based architectures require less storage than many WPRBS architectures-3408Kbits
    - RC-blast spends 64K X 64 bits to store query index
    - Mercury has external BRAM to store hash table
- Word Scanning Speed
    - Mercury-96M matches/s
    - Multiengine-6400M
    - Proposed- 14450M
- Proposed architecture
    - Tree Blast-twice the array size for less FPGA resources
    - Needs less memory space- no hash table or database index stored
    - Word Scanning Speed
    - Architecture more suitable for dealing with longer Q lengths

2.Gapped Blast and two hit

Features

Not for Blastn

- Architecture parameterized in terms of length, match scores, gap penalties , cut off and threshold values
- Implemented in Handel C
- Use BLOSUM50
- Pre Processing, Hits and extension
    - Two hit method and gapped blast for extension
- Two modifications to NW to make local alignment
- Pre Processing done in high level software
- 8 32K X 5 bits S memory
- Sw implemented on Intel Centrino Duo 2.2Ghz with 2GB ram

Claims

- Fpga clocked at 15Mhz
- Upto 44x speed up

3.CAAD Blastn

Features

- 2 HW modules and 3 SW modules
- Pre filtering done on database using Tree-Blastn and Smith Waterman for ungapped and gapped

Claims

- Greater than 12X speed
- 100% accuracy

- 120 s for software and 10 s for proposed

4.RC-Blastn:Implementation of Blastn Scan Function

Features

- Hardware designed to reduce initial comparison latencies between multiple short Q and a S db
- Implemented to provide spatial scalability
- 8 letter word
- Components
    - Input and output FIFO, main hit controller
    - Controller is a FSM coordinating all functions of HW core
- First State- Pop the data from input FIFO to last eight bytes(total nine) of subject buffer
- Each lookup for each byte

Claims

- Maintains same result as the software
- Blast_Nt_Scan is computationally intensive part- consumed 30 – 70 %
- Achieves 4X speed up compared to software
- Mercury- 98 to 99% fidelity
- Tree Blast reports extra alignments
- Trade off of complement FPGA implementations- Restriction on Q length due to limited memory

5.Single Pass

Features

- Only pre-processing is loading the query string

Claims

- Two new algorithm to emulate seeding and extension phase

- Achieve high sensitivity without impact on performance
- Query size 1024
- Cycle time 9ns for up to 600 Q length
- The clock delay is 5.6ns for a throughput of 178 Maa/sec. This last design uses 90% of the slices, 88% of the block RAMs, and 78% of the lookup tables
- a transfer rate from disk to FPGA of 55MB/sec, and memory to FPGA of 320 MB/sec

6.Mitrion: Accelerating NCBI blast

Features

- Mitrion Virtual Processor acts as the core
- Implemented in Mitrion-C
- Software development kit- Mc compiler, graphic simulator and debugger, processor configuration unit

Claims

- 10x – 20x performance improvement
- FPGA memory BW is 10- 20 GB/s as compared to 3-6GB/s for a host system
- BW as high as 0.5 TB/s but limited to only 750KB of storage
- Processor provides a sustained lookup rate of 16 memory loads per clock cycle for a 100k query and 64 memory loads per cycle for a 10k query
- The throughput of the first stage is 400 Megabases per second for a 100k query and 1.6 Gigabases per second for a 10k query r provides a sustained lookup rate of 16 memory loads per clock cycle for a 100k query and 64 memory loads per cycle for a 10k query
- The throughput of the first stage is 400 Megabases per second for a 100k query and 1.6 Gigabases per second for a 10k query