GateKeeper: A New Hardware Architecture for Accelerating Pre-Alignment in DNA Short Read Mapping

Bioinformatics, Volume 33, Issue 21, 1 November 2017, Pages 3355–3363.

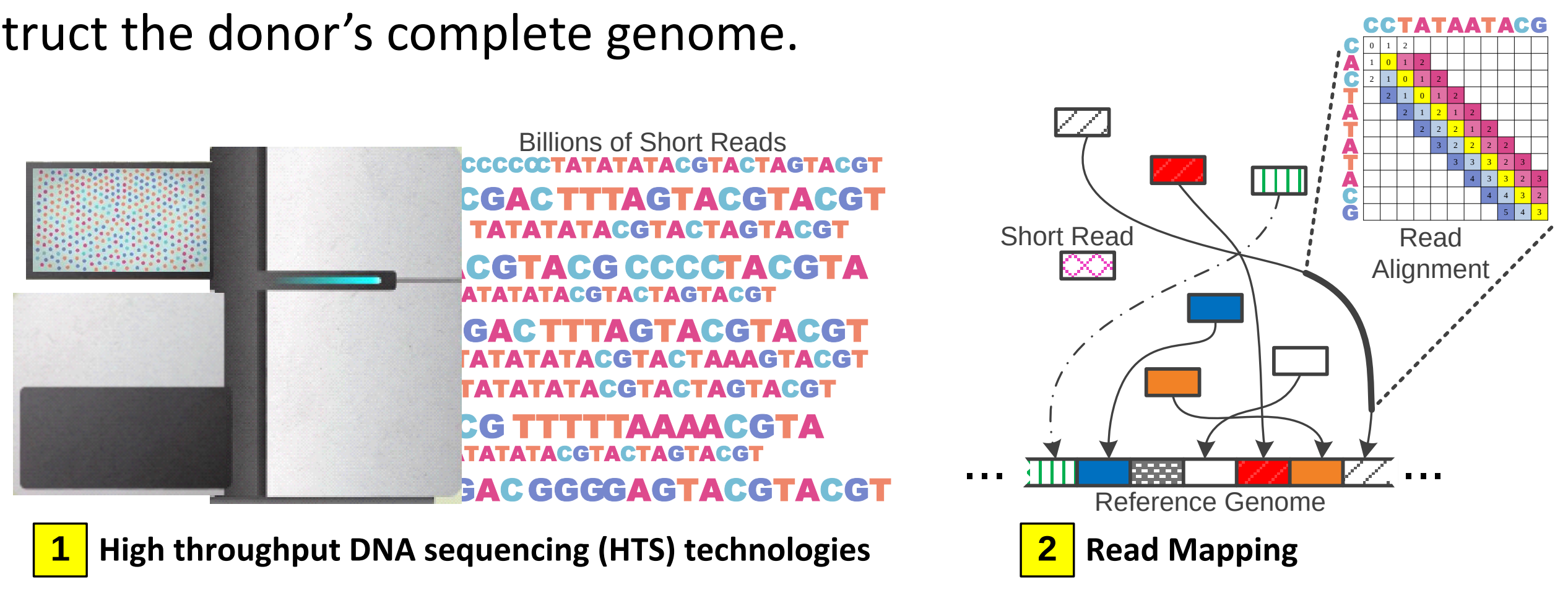
Mohammed Alser[1] Hasan Hassan[2] Hongyi Xin[3] Oğuz Ergin[2] Onur Mutlu[4] Can Alkan[1]

[1]Bilkent University, [2]TOBB University of Economics & Technology, [3]Carnegie Mellon University, [4]ETH Zürich

1: Read Mapping

Fact: until today, it remains challenging to sequence the entire DNA molecule as a whole.

As a workaround: **high throughput DNA sequencing** (HTS) technologies are used to sequence short reads of copies of the original molecule. These technologies are relatively **quick and cost-effective** but result in an **excessive number of short reads**. Reads do not have any information about which part of genome they come from; hence ***read mapping*** is needed. It determines the **optimal alignment** and the potential location of each of the reads within a reference genome to construct the donor's complete genome.



1 High throughput DNA sequencing (HTS) technologies

2 Read Mapping

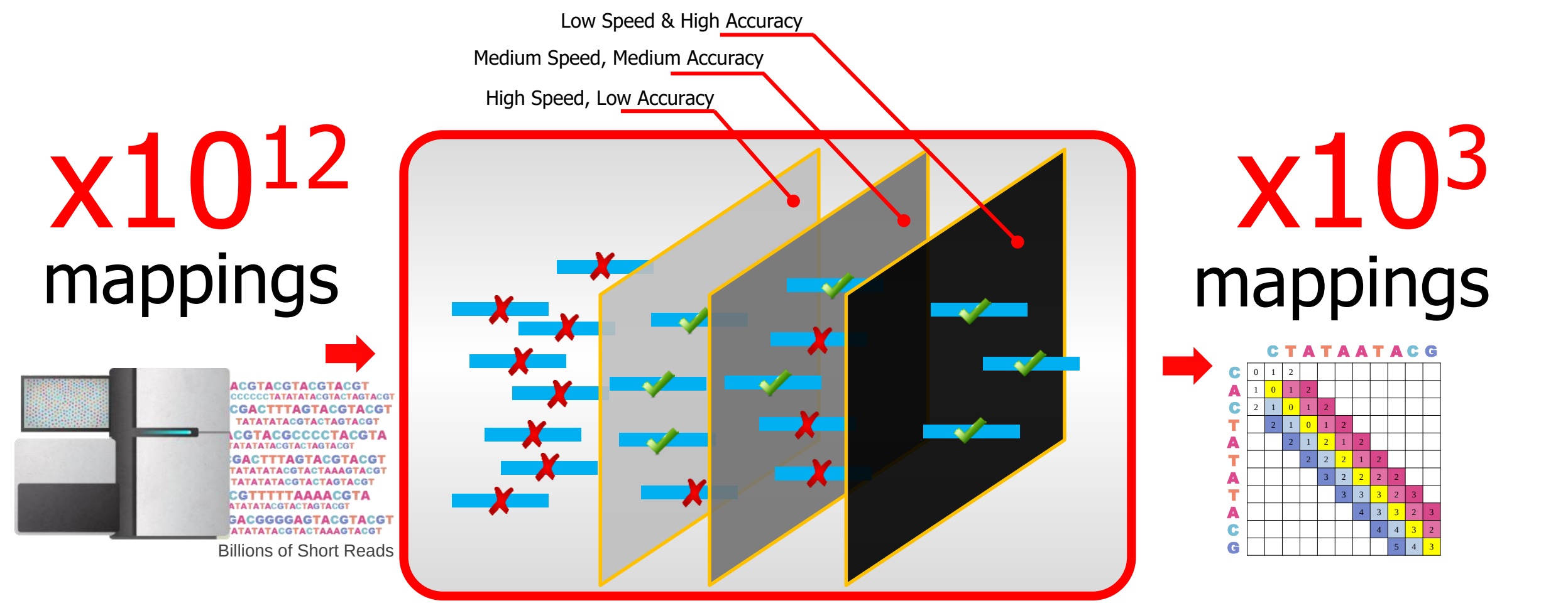
2: Problem

- **Optimal alignment** is **computationally expensive**.
- **Bottlenecked by memory bandwidth**, e.g., Illumina NovaSeq 6000 generates 6 Terabases per 36 hours for each genomic sample.
- **Optimal alignment** algorithms are **unavoidable** as they provide accurate information about the quality of the alignment.
- Majority of **candidate locations** in the reference genome **do not align** with a given read due to **high dissimilarity**. This **wastes execution time and incurs significant computational burden.**

3: Pre-Alignment Filtering

Our Goal: provide the **first hardware accelerator architecture** (as a **pre-alignment** filter) for **quickly** rejecting **incorrect mappings** (highly dissimilar read-reference pairs) that wastes execution time.

- Obtain **low runtime**.
- **Highly-parallel hardware** accelerator design.
- Reject most of **incorrect** mappings (**low false positives**).
- Accept all **correct** mappings (**zero false negatives**).



1 High throughput DNA sequencing (HTS) technologies

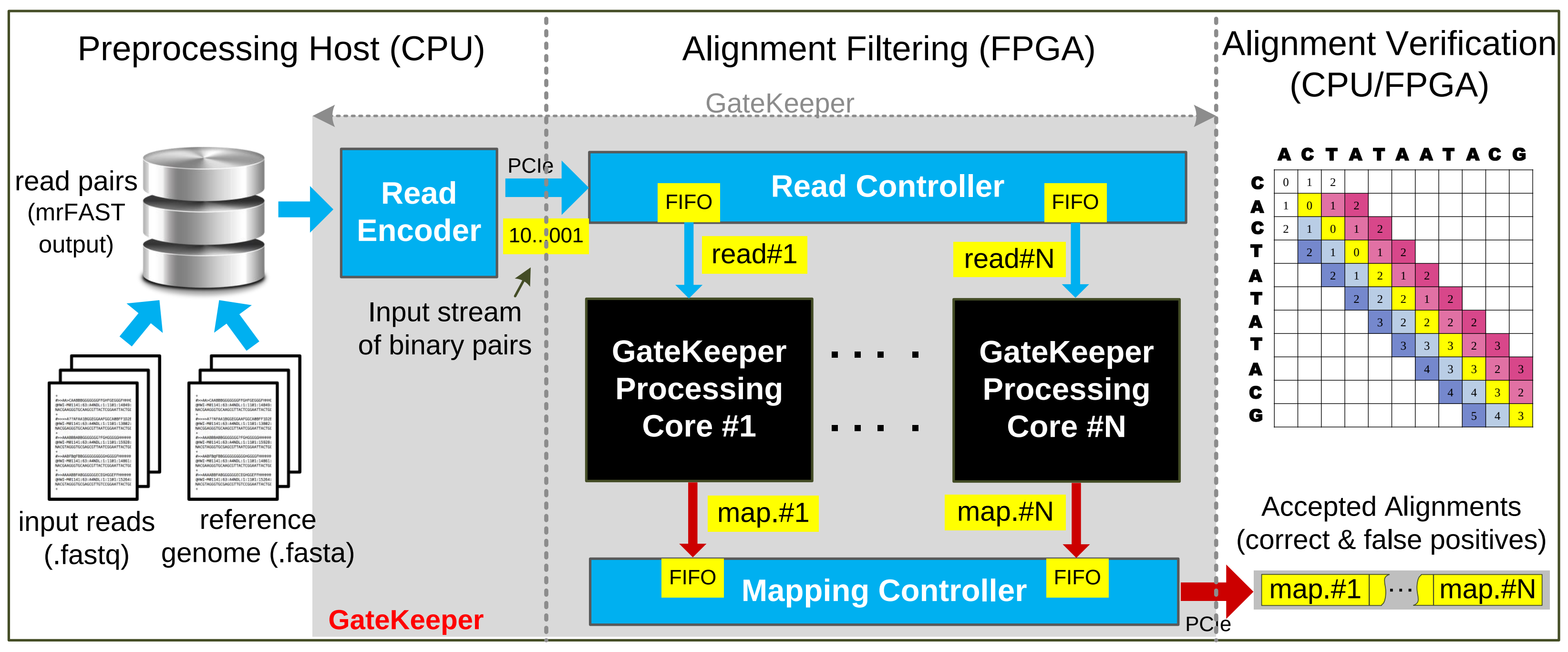
2 Read Pre-Alignment Filtering
Fast & Low False Positive Rate

3 Read Alignment
Slow & Zero False Positives

4: GateKeeper

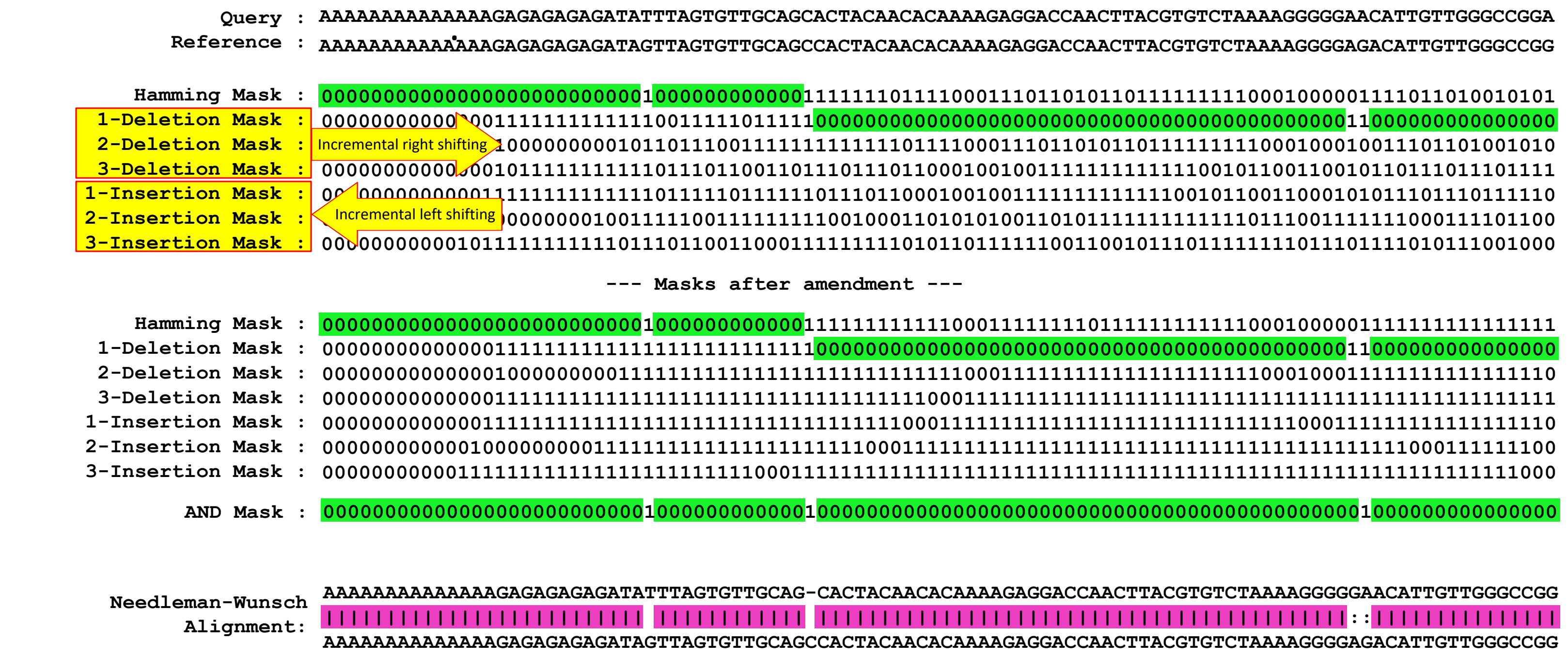
Key Ideas:

- Build a **processing core** that is based on only **parallel bitwise operations** to examine a single mapping.
- Introduce **parallelism** to the pre-alignment step by integrating **many hardware processing cores** for examining many mappings in a parallel fashion.
- Exploit the large amounts of parallelism offered by **FPGA*** architectures to **accelerate the performance of our processing cores**.


Preprocessing Host (CPU)
Alignment Filtering (FPGA)
Alignment Verification (CPU/FPGA)
GateKeeper
read pairs (mrFAST output)
input reads (.fastq)
reference genome (.fasta)
Read Encoder
PCIe
10...001
Input stream of binary pairs
Read Controller
FIFO
FIFO
read#1
read#N
GateKeeper Processing Core #1
GateKeeper Processing Core #N
map.#1
map.#N
FIFO
Mapping Controller
FIFO
PCIe
GateKeeper
ACTATAATACG
CACTATATACG
Accepted Alignments (correct & false positives)
map.#1
map.#N

* **FPGAs (field-programmable gate arrays)** are **the most commonly used** reconfigurable hardware engines today and their computational capabilities are greatly increasing every generation due to increased number of transistors on the FPGA chip. An FPGA can be configured to include a **large number of hardware execution units** that are custom-tailored to the problem at hand (Aluru and Jammula, 2014; Herbordt et al., 2007; Trimberger, 2015).

5: GateKeeper Walkthrough



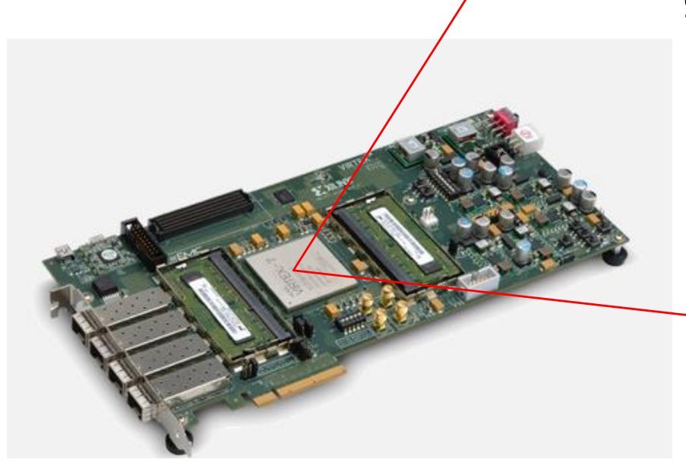
Mechanism:

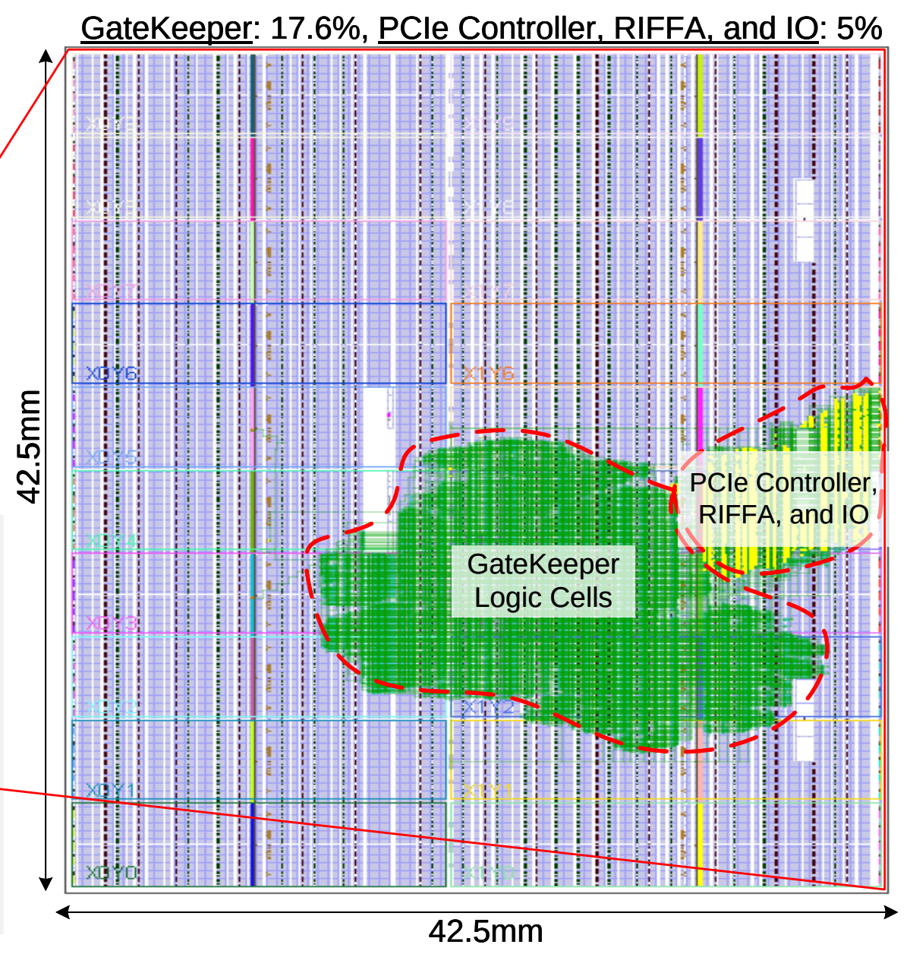
1) **Fast detection of base-substitutions:** If Hamming distance is less than or equal to E, the user-defined edit distance threshold, then this mapping is accepted.
2) **Fast detection of insertions and deletions:**
   - Generate 2E deletion and insertion masks, by incrementally shifting the query to right or left, respectively, then compare against the reference segment.
3) **Apply bit-vector optimization using fast architecture design:**
   - Pre-process all the (2E+1) bit-vectors by encoding them into shorter binary format.
   - Amend each 2 or less consecutive 0's into 1's as they are likely to be random matches.
4) **Calculate shifted Hamming distance (Xin et al., 2015):** By ANDing all bit-vector masks, then conservatively count the 1's in the AND mask. If their number is less than or equal to E then the mapping is accepted and passed to the alignment step.

6: Results & Conclusion

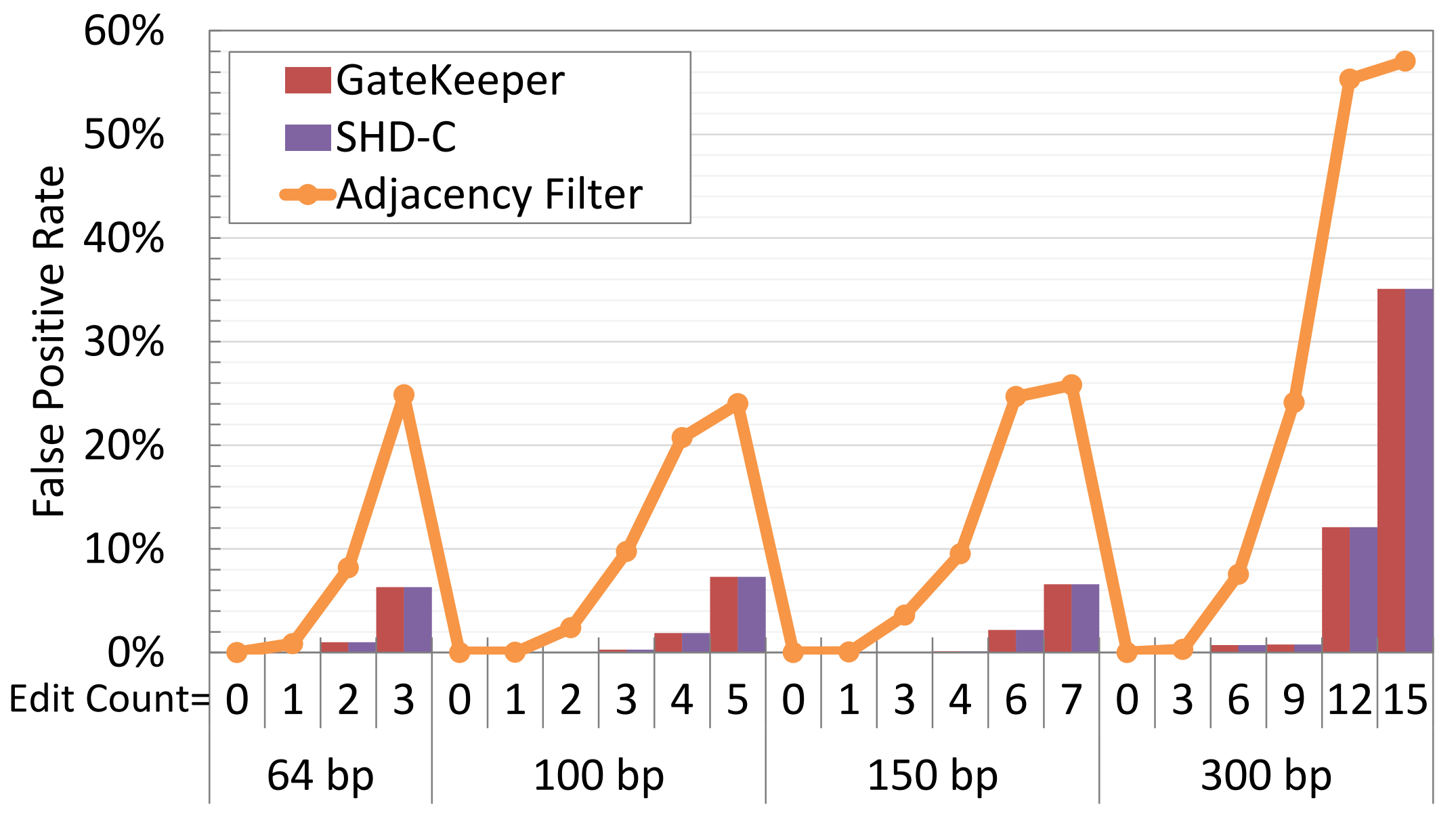
Chip Layout

Xilinx VC709 FPGA Chip layout and breakdown of the chip area for GateKeeper (for a read length of 300 bp and E=15).




GateKeeper: 17.6%, PCIe Controller, RIFFA, and IO: 5%
42.5mm
42.5mm
PCIe Controller, RIFFA, and IO
GateKeeper Logic Cells

Filtering Accuracy vs. Existing Filters



False positive rate (rate of incorrect mappings that are falsely accepted) of GateKeeper, SHD, and the Adjacency Filter across different edit distance thresholds (E) and read lengths.

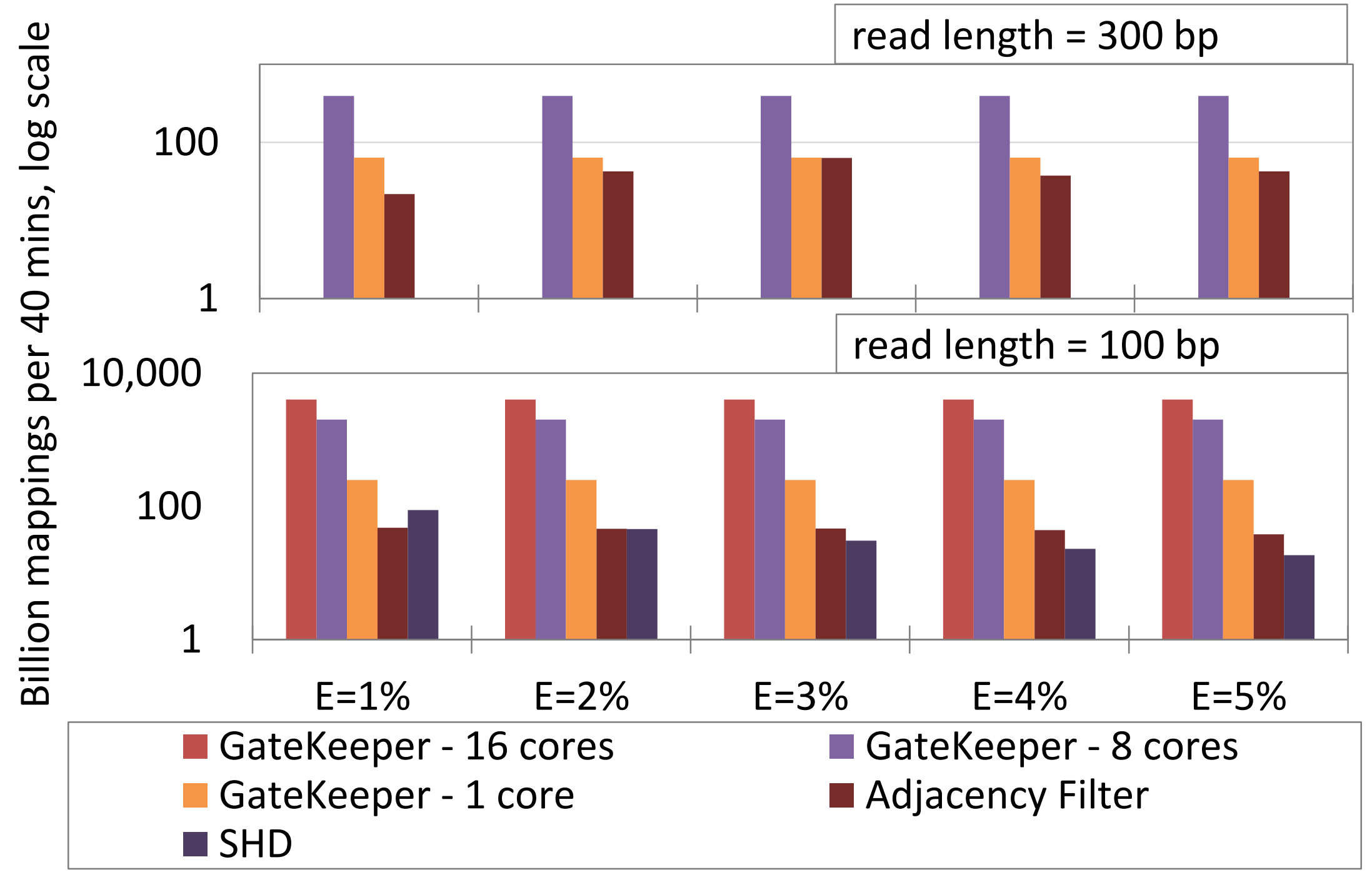
Key Results of GateKeeper:

- **90x-130x faster** than SHD (Xin et al., 2015) and the Adjacency Filter (Xin et al., 2013).
- **4x fewer false positives** (falsely accepted incorrect mappings) the Adjacency Filter (Xin et al., 2013).
- **10x speedup** with the addition of GateKeeper to the mrFAST mapper (Alkan et al., 2009).
- **The first open-source FPGA-based filter** for genome analysis. Github.com/BilkentCompGen/GateKeeper. As such, we hope that it catalyzes the development and adoption of such hardware accelerators in genome sequence analysis, which are becoming increasingly necessary to cope with the processing requirements of greatly increasing amounts of genomic data.

Other Results:

- The number of processing cores is determined by the **maximum data throughput** (~13.3 billion bases per second provided by RIFFA (Jacobsen et al., 2015)) and the **available FPGA resources**.
- GateKeeper can examine up to **8 or 16 mappings concurrently** (at 250 MHz) for an input read length of 300 and 100 bp, respectively.
- GateKeeper **occupies 50%** of the available FPGA slice LUTs and **91% of the available registers** for an input read length of 100 and 300 bp, respectively.
- Pre-alignment filter does not replace alignment verification.
- Integrating the FPGA accelerators with the sequencer can help to hide the complexity and details of the underlying hardware.

Speedup vs. Existing Filters



Number of examined mappings by GateKeeper, SHD, and the Adjacency Filter across different read lengths and E thresholds. SHD does not support 300 bp long reads

Pre-alignment + Alignment Steps

Overall mrFAST mapping time (in hours) with and without a pre-alignment step, with an edit distance threshold of 5%.

Read length / E	mrFAST version / pre-alignment type	Filtering & Verification time (speed-up)	Overall mapping time (speed-up)
100 bp / 5 edits	2.1 / No Pre-alignment	22.60 h (1x)	24.27 h (1x)
	2.6 / Adjacency Filter	5.65 h (4x)	7.31 h (3.3x)
	2.1 / GateKeeper	0.55 h (41x)	2.50 h (9.7x)
300 bp / 15 edits	2.1 / No Pre-alignment	0.94 h (1x)	1.02 h (1x)
	2.6 / Adjacency Filter	0.04 h (24x)	0.12 h (8x)
	2.1 / GateKeeper	0.003 h (279x)	0.09 h (11x)