Democratizing Sequencing with Ion S5™ Sequencers Powered by GPUs

*Mohit Gupta

*GPU work is combined effort with Jakob Siegel

For Research Use Only. Not for use in diagnostic procedures.
Why sequence DNA?
### Why Targeted Sequencing

**More cost effective, more time efficient and simpler to analyze**

<table>
<thead>
<tr>
<th></th>
<th>Targeted Sequencing</th>
<th>Whole Exome</th>
<th>Whole Genome</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Variants generated per run</strong></td>
<td>10s to 100s</td>
<td>~50,000</td>
<td>~3,000,000</td>
</tr>
<tr>
<td><strong>Likely number of variants for follow-up</strong></td>
<td>1-10s</td>
<td>1-10s</td>
<td>1-10s</td>
</tr>
<tr>
<td><strong>Time to analyze</strong></td>
<td>Hours to Days</td>
<td>Days to Weeks</td>
<td>Weeks to Months</td>
</tr>
<tr>
<td><strong>Total cost including analysis</strong></td>
<td>$</td>
<td>$$</td>
<td>$$$</td>
</tr>
</tbody>
</table>
Low Cost, Simple, Scalable, Real Time Sequencing

Wafer
Semiconductor Manufacturing

Chip
Semiconductor Packaging

Single Sensor
Chemical to Digital Sequence

TCGACC...

Millions of Sensors
Semiconductor Design
Transistor as a pH meter

Rothberg J.M. et al Nature doi:10.1038/nature10242

Compute Intensive signal processing
Ion S5™ and Ion S5™ XL Systems

Simplest and fastest targeted sequencing system with the lowest capital investment

- Simplest NGS workflow for targeted sequencing - <15mins to set up a sequencing run and <45mins total hands-on time from DNA to data with Ion Chef™ System
- Fastest run time – as little as 3.5 hours from sequence to BAM files.
- Lowest capital investment - Single platform for all targeted applications with flexibility to scale from 5M - 80M reads
- Lowest DNA/RNA input requirements – as little as 1ng using Ion AmpliSeq™ technology
- Easy setup and training – single day installation and plug and play cartridge-based reagents
Ion S5™ System – Low Cost

- Gene panels to exomes and transcriptomes on a single low cost platform
- Built-in Informatics - no external server required
- Upgradable from S5™ to S5™ XL configuration
- NVIDIA GTX 970

Ion 520™ Chip
5 M Reads

Ion 530™ Chip
15-20 M Reads

Ion 540™ Chip
60-80 M Reads
 Ion S5™ XL System – Speed

- Rapid sequencing and analysis for labs requiring more output or multiple runs per day
- The computing power to enable rapid turnaround times from any size of experiment
- 1 hr analysis time for gene panels, 5 hr analysis for transcriptomes
- NVIDIA Tesla K40

Ion 520™ Chip
5 M Reads

Ion 530™ Chip
15-20 M Reads

Ion 540™ Chip
60-80 M Reads
## Ion S5™ and Ion S5™ XL Systems
Scalable performance on a single platform

### Ion S5™ XL System
![Ion S5™ XL System](image)

<table>
<thead>
<tr>
<th>200bp Sequencing Time</th>
<th>200bp Analysis Time</th>
<th>Output</th>
<th>No. of Reads</th>
<th>Maximum Read Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>S5/S5 XL</td>
<td>Ion S5™ XL System</td>
<td>1-2 Gb</td>
<td>3-5 M</td>
<td>400bp</td>
</tr>
<tr>
<td>2.5 hr</td>
<td>~1 hr</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5M to 80M reads</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Ion S5™ System
![Ion S5™ System](image)

<table>
<thead>
<tr>
<th>200bp Sequencing Time</th>
<th>200bp Analysis Time</th>
<th>Output</th>
<th>No. of Reads</th>
<th>Maximum Read Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ion 520™ Chip</td>
<td>~2.5 hr</td>
<td>3-5 Gb</td>
<td>15-20 M</td>
<td>400bp</td>
</tr>
<tr>
<td>2.5 hr</td>
<td>~2.5 hr</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| Ion 530™ Chip         | ~8 hr               | 10-15 Gb | 60-80 M | 200bp |
| 2.5 hr                | ~5 hr               |        |        |       |

| Ion 540™ Chip         | ~16.5 hr            | 10-15 Gb | 60-80 M | 200bp |
| 2.5 hr                | ~5 hr               |        |        |       |
Data Processing Pipeline

Data acquisition and compression in FPGA

20 TB (540™ chip)

2 TB

Signal Processing

180 GB

BaseCalling and Alignment to reference genome

2 TB

10 20 30
GPU to the rescue

• Removed main hotspot in signal processing pipeline
• **Speedups of more than 250x over a CPU core!**

![Graph showing CPU and GPU processing times with significant speedup for GPU.](image-url)
GPU’s Impact

- Multiple sequencing runs a day possible
- Swift pace of Research and Development
- Accelerated product innovation

On Instrument Analysis Time with and without GPU
Signal Processing
Signal Processing Flow

1. Reading flow data
2. Raw Data Processing
3. Regional Parameter Estimation (Common to all wells)
4. Parameter Estimation unique to each well (LM fitting)
5. Post Fit Processing
6. Writing signal values
Mathematical model

- Sophisticated model
  - Background correction
  - Incorporation
  - Buffering
- Regional Parameters
  - Enzyme kinetics, nucleotide rise, diffusion etc.
- Well Parameters
  - Hydrogen ions generated, buffering, DNA copies etc.
GPU Acceleration
Current Execution Model

- Based on Original CPU implementation: Process Level

- 96 blocks
- depending on hardware 4 to 6 processes in parallel
- work on available data during experiment

*Heat-map and timing from a S5™ XL 540™ with Nvidia Tesla K40 GPU
Current Execution Model

- Based on Original CPU implementation: Thread Level
Current Implementation

- Stream based to hide PCIe transfer

- Resources needed for stream execution are pre-allocated and obtained from a resource pool.

- If resources to create a Stream Execution Unit (SEU) are available the Stream Manager will try to poll a new job from a job queue.

- The Stream Manager can drive multiple SEUs which can be of different types.

- Theoretically up to 16 SEUs can be spawned in one Stream Manager if enough resources are available.
• ~400 MB GPU / 150MB Host memory per stream
• no room for persistent data (36 regions)
• huge allocation and copy overhead
• data transpose overhead
• varying bead count and frames per region, reallocation and slowdown in absolute worst case.
• synchronization steps
Why are further optimizations needed?

- Current pipeline utilizes GPU (more or less) efficiently during bkgmodel fitting
  - Generating empty and bead traces a bottleneck
    - Big chunk of CPU time spent in these computations
    - Mostly memory bound and a natural step to be performed on GPU as a precursor to fitting
- Raw data processing is another big compute hog
  - This pipeline will enable it to be easily streamlined in the new flow
- Many unnecessary data transformations and memcopies
- Complex execution model
Pipeline performance

**Timing S5™ XL 540 (500 flows)**

- K20: 212 minutes
- K40 with boost: 207 minutes
- Current: 192 minutes
- Optimized with MPS: 161 minutes

**GPU Utilization**

- K20: 99% current, 99% optimized with MPS
- K40 with boost: 76% current, 98% optimized with MPS
Signal Processing Flow

1. Reading flow data
2. Raw Data Processing
3. Regional Parameter Estimation (Common to all wells)
4. Parameter Estimation unique to each well (LM fitting)
5. Post Fit Processing
6. Writing signal values
Current Optimization Work

- Expand scope of GPU implementation
  - Modifications in intermediate data layout
  - Removed need for addition copies and transposes
  - Changes in spatial and temporal data subdivision
  - Use of Nvidia MPS to hide PCIe transfers
- Algorithm Changes
  - This is a huge one.
  - Daunting gold standard of current pipeline accuracy to overcome
  - Main focus now
Optimized after 20 flows

- Per block fixed amount ~270 MB GPU memory
- Almost no additional host memory
- Persistent data only copied/generated once on device, no additional transposes.
- No host side copy overhead (use of MPS to hide PCIe)
- Fixed max block size, no need for re-allocation
- No synchronization steps
Optimization Summary

- **520™/530™/540™ block level signal processing**
  - Concept of bkgmodel regions internal to the GPU
  - Easy to experiment with different region sizes
  - Regions can talk to each other.
- Streamlined flow from raw data processing to signal processing
  - Sequential execution of the pipeline steps
  - Final output to be written to 1.wells
  - Fewer data copies and reduced memory footprint
  - Freed up CPU resources
- Reduced context switches on the GPU
- Better utilization of PCIe bandwidth
Thank You

NVIDIA
specifically the DevTech Team
Nikolai Sakhkarnykh
Jonathan Bentz
Andrew Vandergrift
Bob Keating
Mark Berger
Kimberley Powell

Our supervisor Eugene Ingerman
and
The entire Ion Torrent R&D team