Today

Tool of the day: Advanced Version Control

GPU performance
Tool of the day: Advanced Version Control

GPU performance
Version control demo time
Tool of the day: Advanced Version Control

GPU performance
- Less control, more data
- GPUs and Latency
- Understanding GPUs
Tool of the day: Advanced Version Control

GPU performance
  Less control, more data
  GPUs and Latency
  Understanding GPUs
Gratuitous Amounts of Parallelism!

Example:

128 instruction streams in parallel
16 independent groups of 8 synchronized streams
Great if everybody in a group does the same thing.
But what if not?
What leads to divergent instruction streams?

Credit: Kayvon Fatahalian (Stanford)
Example:

128 instruction streams in parallel
16 independent groups of 8 synchronized streams

Credit: Kayvon Fatahalian (Stanford)
Example:

128 instruction streams in parallel
16 independent groups of 8 synchronized streams

Great if everybody in a group does the same thing.

But what if not?

What leads to divergent instruction streams?
But what about branches?

```
if (x > 0) {
    y = pow(x, exp);
    y *= Ks;
    refl = y + Ka;
} else {
    x = 0;
    refl = Ka;
}
```

Credit: Kayvon Fatahalian (Stanford)
But what about branches?

```c
if (x > 0) {
    y = pow(x, exp);
    y *= Ks;
    refl = y + Ka;
} else {
    x = 0;
    refl = Ka;
}
```

Credit: Kayvon Fatahalian (Stanford)
Branches

Not all ALUs do useful work!
Worst case: 1/8 performance

if (x > 0) {
    y = pow(x, exp);
    y *= Ks;
    refl = y + Ka;
} else {
    x = 0;
    refl = Ka;
}

Credit: Kayvon Fatahalian (Stanford)
But what about branches?

```
if (x > 0) {
    y = pow(x, exp);
    y *= Ks;
    refl = y + Ka;
} else {
    x = 0;
    refl = Ka;
}
```

Credit: Kayvon Fatahalian (Stanford)
Branch demo time
Tool of the day: Advanced Version Control

**GPU performance**
- Less control, more data
- GPUs and Latency
- Understanding GPUs
GPUs vs Latency

Problem
Memory still has very high latency... ...as do many other things... ...but we’ve removed most of the hardware that helps us deal with that.

We’ve removed
- caches
- branch prediction
- out-of-order execution

So what now?
GPUs vs Latency

Memory still has very high latency... as do many other things... but we've removed most of the hardware that helps us deal with that.

- caches
- branch prediction
- out-of-order execution

So what now?

Version Control GPU performance
GPUs vs Latency

Problem

Memory still has very high latency... as do many other things... but we've removed most of the hardware that helps us deal with that.

- caches
- branch prediction
- out-of-order execution

So what now?

Idea #3

Even more parallelism

+ Some extra memory

= A solution!
GPUs vs Latency

Problem

Memory still has very high latency. . .
. . . as do many other things. . . but we've removed most of the hardware that helps us deal with that.

We've removed

• caches
• branch prediction
• out-of-order execution

So what now?

Idea #3

Even more parallelism

+ Some extra memory

= A solution!
Hiding Memory Latency

Credit: Kayvon Fatahalian (Stanford)
Hiding Memory Latency

Time (clocks)

Frag 1 … 8
Frag 9… 16
Frag 17 … 24
Frag 25 … 32

Credit: Kayvon Fatahalian (Stanford)
Hiding Memory Latency

Credit: Kayvon Fatahalian (Stanford)
Hiding Memory Latency

Credit: Kayvon Fatahalian (Stanford)
Hiding Memory Latency

Credit: Kayvon Fatahalian (Stanford)
Hiding Memory Latency

Increase run time of one group
To maximum throughput of many groups

Credit: Kayvon Fatahalian (Stanford)
GPUs and latency demo
Outline

Tool of the day: Advanced Version Control

GPU performance
  - Less control, more data
  - GPUs and Latency

Understanding GPUs
### Comparing Architectures

<table>
<thead>
<tr>
<th></th>
<th>GF100</th>
<th>GF104</th>
<th>GK104</th>
<th>GCN</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td># Warps/Wavefronts</td>
<td>48</td>
<td>48</td>
<td>64</td>
<td>40</td>
<td></td>
</tr>
<tr>
<td>Warp Size</td>
<td>32</td>
<td>32</td>
<td>32</td>
<td>64</td>
<td>W. Item</td>
</tr>
<tr>
<td>SP FLOP/clock</td>
<td>64</td>
<td>96</td>
<td>384</td>
<td>128</td>
<td>MHz</td>
</tr>
<tr>
<td>Clock</td>
<td>700</td>
<td>650</td>
<td>823</td>
<td>925</td>
<td>MHz</td>
</tr>
<tr>
<td>Reg File</td>
<td>128</td>
<td>128</td>
<td>256</td>
<td>256</td>
<td>kiB</td>
</tr>
<tr>
<td>Lmem</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>64</td>
<td>kiB</td>
</tr>
<tr>
<td>Lmem BW</td>
<td>64</td>
<td>64</td>
<td>128</td>
<td>128</td>
<td>B/clock</td>
</tr>
</tbody>
</table>
Architecture by the numbers demo
Questions?