Zorua: A Holistic Approach to Resource Virtualization in GPUs

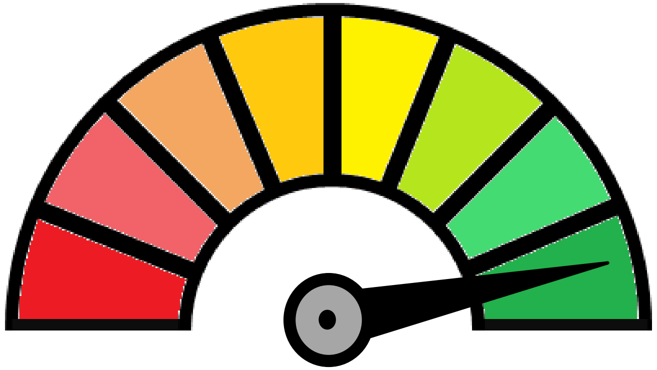
Session 2A

Monday, 5:20 PM

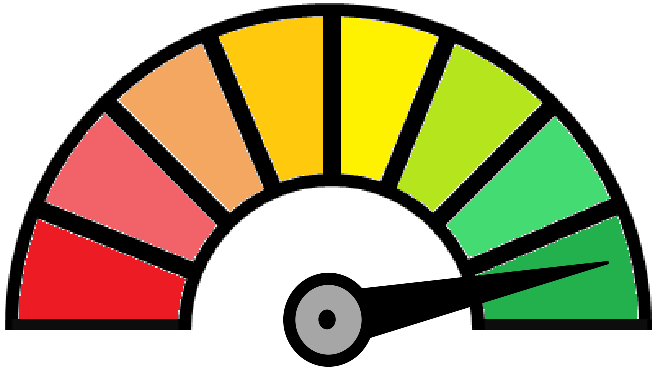
Nandita Vijaykumar

Kevin Hsieh, Gennady Pekhimenko, Samira Khan,
Ashish Shrestha, Saugata Ghose, Adwait Jog, Phillip B. Gibbons, Onur Mutlu

Carnegie Mellon University Microsoft Research University of Virginia William & Mary ETH Zürich



High
Performance

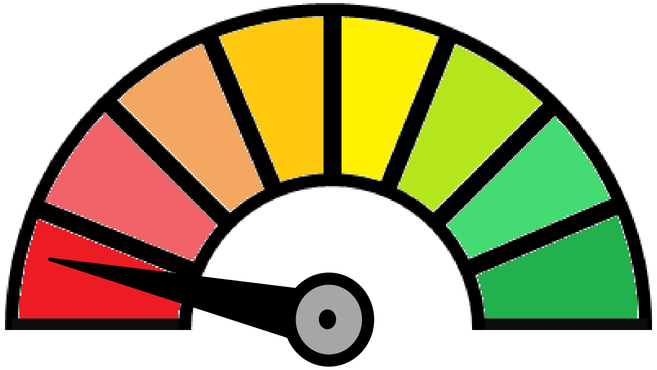


High Performance

GPUs

```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
            bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
            dst[i *I] = bl_ptr[i * X]; }
```

```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
            bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
            dst[i *I] = bl_ptr[i * X]; }
```



Low Performance!

The programmer has to statically allocate 3 major resources:

The programmer has to statically allocate 3 major resources:

- ***Registers*** R

The *programmer* has to statically allocate 3 major resources:

- **Registers** R
- **Scratchpad Memory** S

The ***programmer*** has to statically allocate 3 major resources:

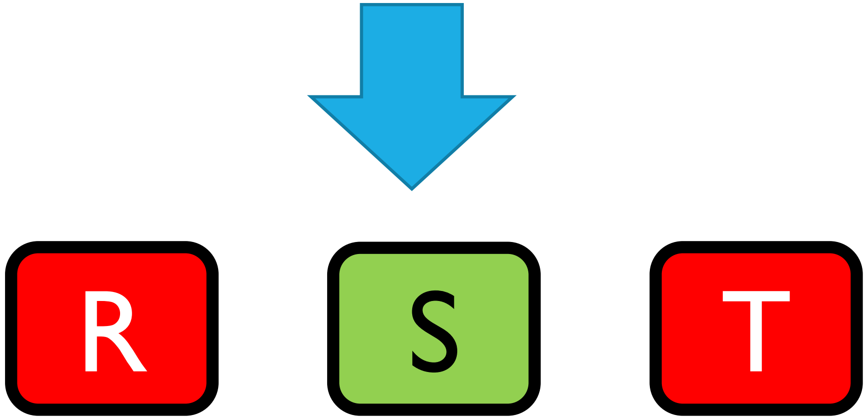
- ***Registers*** R
- ***Scratchpad Memory*** S
- ***Thread Slots*** T

The programmer has to statically allocate 3 major resources:

- ***Registers*** R
- ***Scratchpad Memory*** S
- ***Thread Slots*** T

Imperfect Allocation ⇨ Low Performance

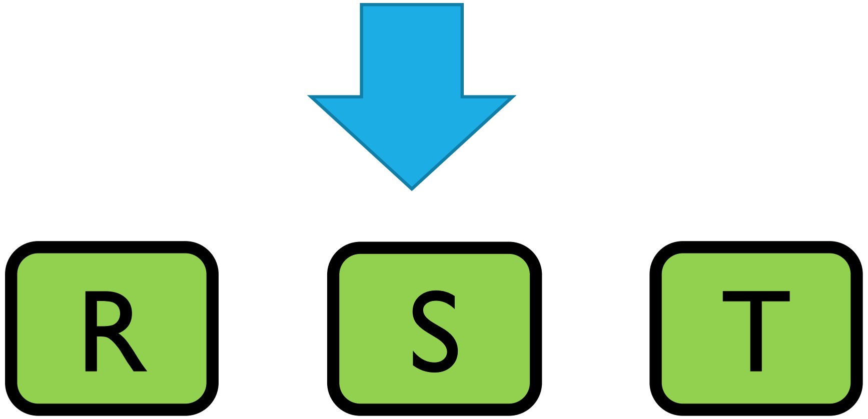
```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
            bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
            dst[i *I] = bl_ptr[i * X]; }
```

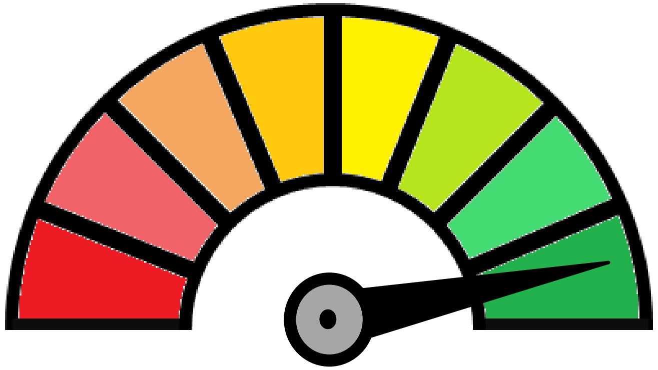


Tune Code

FIX: ***Usage of Registers, Scratchpad and Thread Slots***

```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
            bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
            dst[i *I] = bl_ptr[i * X]; }
```





High Performance

Problem: Programming Effort

```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
            bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
            dst[i *I] = bl_ptr[i * X]; }
```

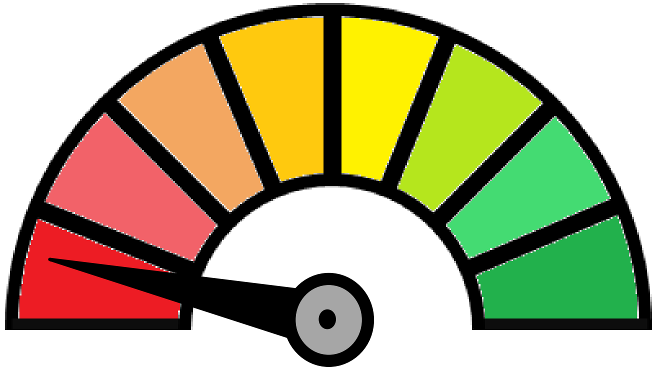
GPU 1

GPU 2

🟩 R 🟩 S 🟩 T

🟥 R 🟥 S 🟩 T

```
__global__ void CUDAkernel2DCT(float *dst,
float *src, int I){
    int OffsThreadInRow = threadIdx.y * B +
threadIdx.x;
    for(unsigned int i = 0; i < B; i++)
        bl_ptr[i * X] = src[i * I];
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);
    __syncthreads();

    CUDAsubroutineInplaceDCTvector(…);

    for(unsigned int i = 0; i < B; i++)
        dst[i *I] = bl_ptr[i * X]; }
```



Low Performance!

Problem: Performance Portability

***Programmer-specified* resource allocation leads to 3 key issues with:**

- ***Programming ease***
- ***Performance portability***
- ***Performance for optimized code***

Our Approach

Decouple

Programmer-specified resource usage

↕

Allocation in the hardware

Zorua:

A Framework to Virtualize On-chip Resources in GPUs

Zorua: A Holistic Approach to Resource Virtualization in GPUs

Session 2A

Monday, 5:20 PM

Nandita Vijaykumar

Kevin Hsieh, Gennady Pekhimenko, Samira Khan,
Ashish Shrestha, Saugata Ghose, Adwait Jog, Phillip B. Gibbons, Onur Mutlu

Carnegie Mellon University

Microsoft Research

University of Virginia

William & Mary

ETH Zürich