PHAS0102: Techniques of High-Performance Computing

GPUs and CPUs

- CPU = central processing unit
 - Designed to be good at everything a computer needs to do
- GPU = graphical processing unit
 - Designed to be good at processing 3D graphics

GPUs and CPUs CPU GPU A few powerful cores A lot of smaller cores

GPUs

- Can do many parallel flops at the same time
- Usually only fast for single precision computations
- Copying memory to a GPU can take time, and GPUs have limited storage

Common GPU manufacturers

- Nvidia
- AMD
- Intel

Programming on a GPU

- Cuda
 - Nvidia specific GPU API
 - Can be used from Python, C, C++, Fortran, Matlab, Julia, and more
- OpenCL
 - Open \rightarrow can be used on all platforms
 - Can run on GPUs or CPUs
- SYCL
 - More modern open standard

Programming on a GPU

- OneApi
 - Developed by Intel
 - Cros platform
- OpenACC
 - Can be used from C, C++ and Fortran
 - Used on many large HPC systems
- OpenMP
 - Can be used from C, C++ and Fortran
 - First developed for CPUs but more recent versions also support GPUs

Programming on a GPU

• My personal recommendation

if you're using an Nvidia GPU: Cuda elif you're using C++: SYCL else: OpenCL

pycuda and pyopencl

- These two libraries allow you to use Cuda and OpenCL directly from Python.
- There are examples in the lecture notes.

Using Cuda with Numba

from numba import cuda

[live Numba & CUDA demo]

Cuda device model

- Streaming multiprocessor (SM)
 GPUs are made up of multiple SMs
- Warps
 - A collection of blocks
 - Each thread in a warp must follow the same execution path
- Blocks
 - A collection of threads
- Thread
 - Threads are where calculations are actually done

			L1 Instr	ucti	ion Cac	he								
L0 instru	uction C	ache						L0 In	struc	tion C	ache			
Warp Schedu		Warn Scheduler (32 thread/clk)												
Dispatch Ur		Dispatch Unit (32 thread/clk)												
Register File	(16,38	4 x 32-bit)					Reg	ister	File (16,38	4 x 32	2-bit)		
INT32 INT32 FP32 P32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64	TENSOR CORE		I	INT32 INT32 FP32 FP32 FF		P64	1						
INT32 INT32 FP32 FP32	FP64			Ι	INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 P32	FP64				INT32	INT32	FP32	FP32	FF	FP64		TENSOR CORE		
INT32 INT32 FP32 P32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT3 FP32 1 32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	P64				
LD/ LD/ LD/ LD/ LD/ ST ST ST ST ST	/ LD/ ST	LD/ LD/ ST ST	SFU	I	LD/ ST	LD/ ST	LD/ ST	LD/ ST	LD/ St	LD/ ST	LD/ ST	LD/ ST	SFU	
				וה										
L0 Instru	iction C	ache			L0 Instruction Cache									
Warp Schedu	ler (32 t	hread/clk)			Warp Scheduler (32 thread/clk)									
Dispatch Ur	it (32 th	read/clk)					וט	spater	i Unit	(32 th	iread/	CIK)		
Register File	(16,38	4 x 32-bit)			L		Reg	ister	File (16,38	4 x 32	2-bit)		
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	FP64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	P64	TENSOR CORE			
INT32 INT32 FP32 FP32	FP64	TENSO	R CORE		INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	FP64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	P64				
INT32 INT32 FP32 FP32	FP64				INT32	INT32	FP32	FP32	FF	FP64				
INT32 INT32 FP32 FP32	/ LD/	LD/ LD/	OFU		LD/	LD/	LD/	LD/	LD/	-64 LD/	LD/	LD/	OF	
		ST	ST	ST	ST	ST	ST	ST	ST	SFU				
ST ST ST ST ST				·										
		192K	B L1 Data C	ach	ie / Sha	red M	emory	/						

Cuda device model

- Threads for integer calculations -
- Threads for float calculations
- Tensor threads

					L1 Instru	ucti	on Cad	che							
	L0 Instru	ction C	ache			ור			L	_0 In:	struct	ion C	ache		
V	Varp Schedul	er (32 t	hread	/clk)					Warp	Sche	eduler	(32 t	hread	/clk)	_
	Dispatch Uni	it (32 th	read/c	:lk)			Dispatch Unit (32 thread/clk)								
R	Register File (16,384 x 32-bit)						Register File (16,384 x 32-bit)								
INT32 INT32 FP	32 FP32 F	P64					INT32	INT32	FP32 FF	932	FP	64			
INT32 INT32 FP	32 FP32 F	P64					1	INT32	FP32 FF	- 32	FP	64			
	32 FP32 F	P64				1	INT32	INT	-022 FF	932	FP	64			
INT32 INT32 FP	32 FP32 F	P64					INT32	INT3∠	FP32 FF	9 32	FP	64			
	32 EP32				CORE		INT32	INT32	ED32 E	232	ED	64	TE	INSO	R CORE
		-													
IN T32 FP	32 FP32 F	P64					INT32	INT32	CDoo						
INT32 INT32 FP	32 FP32 F	P64					INT32	INT32	FP32 FF	- 32	FP	64			
	32 FP32 F	P64					INT32	INT32	FP32 FF	- 32	FP	64			
LD/ LD/ LI	D/ LD/ LD/	LD/	LD/	LD/	0511		LD/	LD/	LD/ L	_D/	LD/	LD/	LD/	LD/	OFU
ST ST S	T ST ST	ST	ST	ST	SFU		ST	ST	ST	ST	ST	ST	ST	ST	SFU
	I A Instruction Cache									0 In	struct	ion C	ache		
L0 Instruction Cache Warp Scheduler (32 thread/clk)											Suuce		aure		
	Varn Schedul	er (32 t	hread	(clk)				_	Warn	Sche	eduler	(32 f	hread	/clk)	
V	Varp Schedul Dispatch Uni	er (32 tl it (32 th	hread read/c	/clk) :lk)				_	Warp Disp	Sche atch	eduler Unit (<mark>. (32 t</mark> (32 th	hread read/o	/clk) :lk)	
R	Varp Schedul Dispatch Uni egister File	ler (32 th it (32 th (16,384	hread/c read/c 4 x 32	/clk) :lk) !-bit)					Warp Disp Regis	Sche atch ter F	eduler Unit (-ile (1	(32 th (32 th 6,384	hread read/o 4 x 32	/clk) :lk) ?-bit)	
R INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F	er (32 th it (32 th (16,384 P64	hread read/c 4 x 32	/clk) :lk) :-bit)			INT32	INT32	Warp Disp Regis FP32 FF	Sche atch ter F 932	eduler Unit (=ile (1 FP(6,384	hread read/o 4 x 32	/clk) :lk) ?-bit)	
R INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64	hread, read/c 4 x 32	/clk) :lk) :-bit)			INT32 INT32	INT32	Warp Disp Regis FP32 FF	Sche atch ter F 932 932	eduler Unit (-ile (1 FP(FP((32 th (32 th 6,384 64 64	hread/o read/o 4 x 32	/clk) :lk) ?-bit)	
NT32 INT32 FP INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64	hread, read/c 4 x 32	/clk) :lk) :-bit)			INT32 INT32 INT32	INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF	Sche atch ter F 232 232 232	eduler Unit (FP(FP(FP((32 th (32 th 6,384 64 64 64	hread read/c 4 x 32	/clk) :lk) 2-bit)	
INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64 :P64	hread/c read/c 4 x 32	(clk) :lk) :-bit)	P COBE		INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF	Sche atch ter F 932 932 932 932	eduler Unit (File (1 FP(FP(FP((32 th (32 th 6,384 64 64 64 64	hread read/o 4 x 32	/clk) :lk) 2-bit)	
INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64 :P64	hread/c read/c 4 x 32 TE	(cik) :ik) :-bit)	R CORE		INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF	School Sc	eduler Unit (FP(FP(FP(FP((32 th (32 th 6,384 64 64 64 64 64	hread read/c 4 x 32 TE	/clk) :lk) ?-bit) ENSO	R CORE
INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64 :P64 :P64	hread, read/c 4 x 32	(c1k) :1k) :-bit)	R CORE		INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF	Scha atch ter F 232 232 232 232 232 232	eduler Unit (File (1 FP(FP(FP(FP(FP(r (32 th (32 th 6,384 64 64 64 64 64 64	hread read/c 4 x 32 TE	/cik) :ik) 2-bit) ENSO	R CORE
INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64 :P64 :P64 :P64	hread, read/c 4 x 32	(clk) llk) l-bit)	R CORE		INT32 INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 Ff FP32 Ff FP32 Ff FP32 Ff FP32 Ff FP32 Ff	Sche atch ter F -32 -32 -32 -32 -32 -32 -32 -32	eduler Unit (File (1 FPC FPC FPC FPC	(32 t) (32 t) (32 t) (33 t) (33 t) (34 t) (35 t) (36 t) (3	hread read/o 4 x 32	/clk) (lk) t-bit)	R CORE
INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F	er (32 th it (32 th (16,384 :P64 :P64 :P64 :P64 :P64 :P64 :P64	hread/c read/c 4 x 32	(clk) llk) -bit)	R CORE		INT32 INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF	Sche atch ter F 232 232 232 232 232 232 232 232 232 23	eduler Unit (File (1 FPR FPR FPR FPR FPR FPR FPR	(32 t 32 th 6,38 6 6 6 6 6 6 6 6	hread read/o 4 x 32	/clk) :lk) 2-bit)	R CORE
INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F	er (32 tt i (36,384))))))))))))))))))))))))))))))))))))	tread/c 4 x 32 TE	(clk) ilk) i-bit) :NSO	R CORE		INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF	Sche atch ter F -32 -32 -32 -32 -32 -32 -32 -32 -32 -32	eduler Unit (FPG FPG FPG FPG FPG FPG FPG FPG FPG FPG	(32 th (32 th 6,384 64 64 64 64 64 64 64 64 64 64 64	hread read/ 4 x 32 TE	/clk) :lk) ?-bit) ENSO	R CORE
INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 22 FP32 F 22 FP32 F 32 FP32 F	I (32 th (16,384 IP64 IP64	hread/c 4 x 32 TE	(clk) ilk) i-bit) :NSO	R CORE SFU	l	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF	Sche atch ter F 232 232 232 232 232 232 232 232 232 23	eduler Unit (FPI FPI FPI FPI FPI FPI FPI FPI FPI FPI	(32 th (32 th 6,384 64 64 64 64 64 64 64 64 64 51	hread read/c 4 x 32 TE LD/ ST	/clk) :lk) 2-bit) ENSO	R CORE SFU
INT32 INT32 FP INT32 INT32 FP	Varp Schedul Dispatch Uni egister File 32 FP32 F 32 FP32 F 33 F 34 F 34 F 34 F 34 F 34 F 34 F 34	er (32 tt (16,384 P64 P64 P64 P64 P64 P64 P64 P64 LD/ ST	hread/c 4 x 32 TE	(clk) (lk) (lk) (lb) (st) (lb) (st) (lb) (st) (lb) (st) (lb) (st) (lb) (st) (lb) (st) (lb) (st) (lb) (lb) (lb) (lb) (lb) (lb) (lb) (lb	R CORE SFU		INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Warp Disp Regis FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF FP32 FF CP32 FF CP32 FF CP32 FF	Sche atch ter F -32 -32 -32 -32 -32 -32 -32 -32 -32 -32	eduler Unit (FPIe (1) FPI FPI FPI FPI FPI FPI FPI	(32 th (32 th 6,384 64 64 64 64 64 64 64 64 64 64 64 57	hread read/c 4 x 32 TE LD/ ST	/cik) :ik) 2-bit) ENSO	R CORE

Cuda device model

- Global memory
 - Access from threads to global memory is slow
- Shared memory
 - Shared within a block
- Private memory
 - Used by a thread during calculations

			c1 instruc	tion Cache							
		action	Cache		L0 Ir	nstruction C	ache				
	warp Sch	eduler (32	thread/clk)	Warp Scheduler (32 thread/clk)							
	Dispatch	1 Unit (32 t	hread/clk)	Dispatch Unit (32 thread/clk)							
	Register	File (16,3	34 x 32-bit)		Register	File (16,384	4 x 32-bit)				
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
NT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
NT32 INT32	FP32 FP32	FP64	TENSOR CORE	INT32 INT32	FP32 FP32	FP64	TENSOR CORE				
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
INT32 INT32	FP32 FP32	FP64		INT32 INT32	FP32 FP32	FP64					
LD/ LD/ ST 07	LD/ LD/ ST	LD/ LD/ ST ST	LD/ LD/ ST ST SFU	LD/ LD/ ST ST	LD/ LD/ ST ST	LD/ LD/ ST ST	LD/ LD/ ST ST SFU				
	1.0.1r	struction	Cache	L 0 Instruction Cache							
	Warp Sch	eduler (32	thread/clk)	Warp Scheduler (32 thread/clk)							
	Dispatch	1 Init (32 t									
		1 01111 (02 1	hread/clk)		Dispatcl	h Unit (32 th	read/clk)				
	Register	File (16,3	hread/clk) 34 x 32-bit)		Dispatch Register	h Unit (32 th File (16,384	read/clk) 4 x 32-bit)				
INT32 INT32	Register	File (16,34	hread/cik) 84 x 32-bit)	INT32 INT32	Dispatch Register FP32 FP32	h Unit (32 th File (16,384 FP64	read/clk) 4 x 32-bit)				
NT32 INT32 NT32 INT32	Register FP32 FP32 FP32 FP32	File (16,34 FP64 FP64	14 x 32-bit)	INT32 INT32 INT32 INT32	FP32 FP32	h Unit (32 th File (16,384 FP64 FP64	read/clk) 4 x 32-bit)				
INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32	File (16,34 FP64 FP64 FP64	hread/cik)	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	h Unit (32 thi File (16,384 FP64 FP64 FP64	read/clk) 4 x 32-bit)				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	File (16,3) FP64 FP64 FP64 FP64	HTENCOR CORE	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	h Unit (32 thi File (16,384 FP64 FP64 FP64 FP64	read/clk) 4 x 32-bit)				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	File (16,34 FP64 FP64 FP64 FP64 FP64	tensor core	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	r Unit (32 th File (16,384 FP64 FP64 FP64 FP64 FP64	read/clk) 4 x 32-bit) TENSOR CORE				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	File (16,33 FP64 FP64 FP64 FP64 FP64 FP64	tensor core	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Dispatch Register FP32 FP32	File (16,384 FP64 FP64 FP64 FP64 FP64 FP64 FP64	read/clk) 4 x 32-bit) TENSOR CORE				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32	File (16,34 FP64 FP64 FP64 FP64 FP64 FP64 FP64	tensor core	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Dispatch Register FP32 FP32	File (16,384 FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP6	read/clk) 4 x 32-bit) TENSOR CORE				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32	File (16,34 FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP6	tensor core	INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Dispatch Register FP32 FP32	File (16,384 FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP6	read/clk) 4 x 32-bit) TENSOR CORE				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP33 FP32 FP34 FP3	File (16, 3) FP64 FP64 FP64 FP64 FP64 FP64 FP64 EP64 LD/ ST LD/ ST	TENSOR CORE	INT32 INT32 INT32 INT32	Dispatcl Register FP32 FP32 FP32 FP32 FP32 FP32	FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP64	tead/clk) t x 32-bit) TENSOR CORE LD/ ST LD/ ST SFU				
INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32 INT32	Register FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 FP32 CD' CD' ST	File (16,3) FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP64	TENSOR CORE	INT32 INT32 INT32 INT32	Dispatch Register FP32 FP32 FP32 FP32 <	File (16,384 FP64 FP64 FP64 FP64 FP64 FP64 FP64 FP6	tensor core				

Thread numbering

gridDim.x = 4096threadIdx.x threadIdx.x threadIdx.x threadIdx.x 255 255 255 2 255 0 1 2 ••• blockIdx.x = 0blockIdx.x = 1 blockIdx.x = 2 blockIdx.x = 4095index = blockIdx.x * blockDim.x + threadIdx.x

index = (2) * (256) + (3) = 515

• In this example, threads are arranged in a line. Threads could also be arranged into a 2D or 3D array.