
CUDA

**Georges-Emmanuel Moulard
Paul Karlshöfer**



Warp-level operations

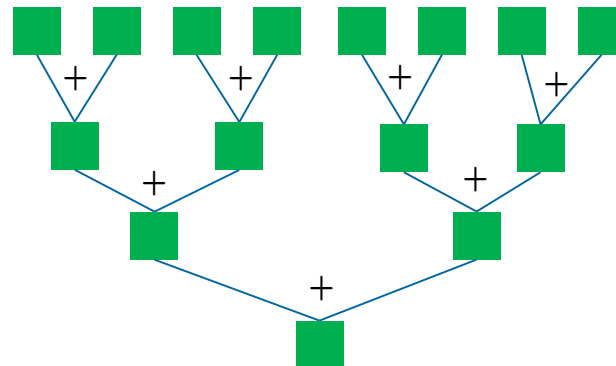
05/08/2020

Recap: Communication between Threads

Communication	Threads in distinct threadblocks	Threads in same threadblock	Threads in same warp
Level of synchronization	Grid-level	Block-level	Warp-level
Reflect in code	Distinct kernel launches	<code>__syncthreads()</code>	warp-level primitives

Warp-level operations

- ▶ CUDA provides some operations for data exchange, synchronization and querying the execution flow on the warp level.
- ▶ Warp-level operations are important, if data dependencies between threads of a warp exists.
- ▶ Example: Reducing 32 values (one per thread) held in registers



Warp-level operations

► Example:

```
T __shfl_sync(unsigned mask, T var, int srcLane, int width=warpSize);
```

- T can be any primitive numeric type (double included)
- Different shuffle operations exist. (e.g. shuffle_down, shuffle_xor)
- Since CUDA 9.0, all warp-level functions take a mask and are suffixed “sync”
 - Non sync functions are deprecated and especially on cc 7.x not valid!
- The shuffle functions allow exchange a variable (4 or 8 byte) between threads within a warp without having to pass by shared memory

Warp-level operations

- ▶ Threads within a warp are called **lanes**. (index from 0 .. (warpsize - 1))
- ▶ The **mask** specifies which threads are participating in the call
 - If a lane is not marked in the mask, but participating, the result is undefined

```
#define FULL_MASK 0xffffffff
__global__ void warp_broadcast(int *ret){
    int val = 0;
    if(threadIdx.x == 0)
        val = 42;
    val = __shfl_sync(FULL_MASK, val, 0, 32);

    if(threadIdx.x == 11)
        printf("%d \n", val);    //prints 42
}
```

Reduction (warp-level)

```
#define FULL_MASK 0xffffffff
```

```
__inline__ __device__
```

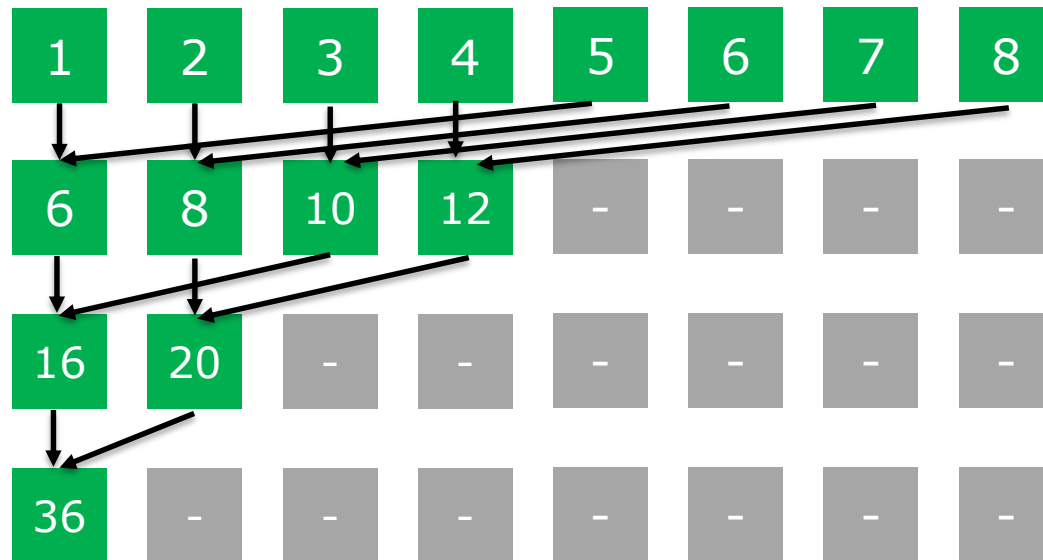
```
void warp_reduction(int *val) {
```

```
    for (int offset = WARPSIZE/2; offset > 0; offset /= 2)
```

```
        *val += __shfl_down_sync(FULL_MASK, *val, offset);
```

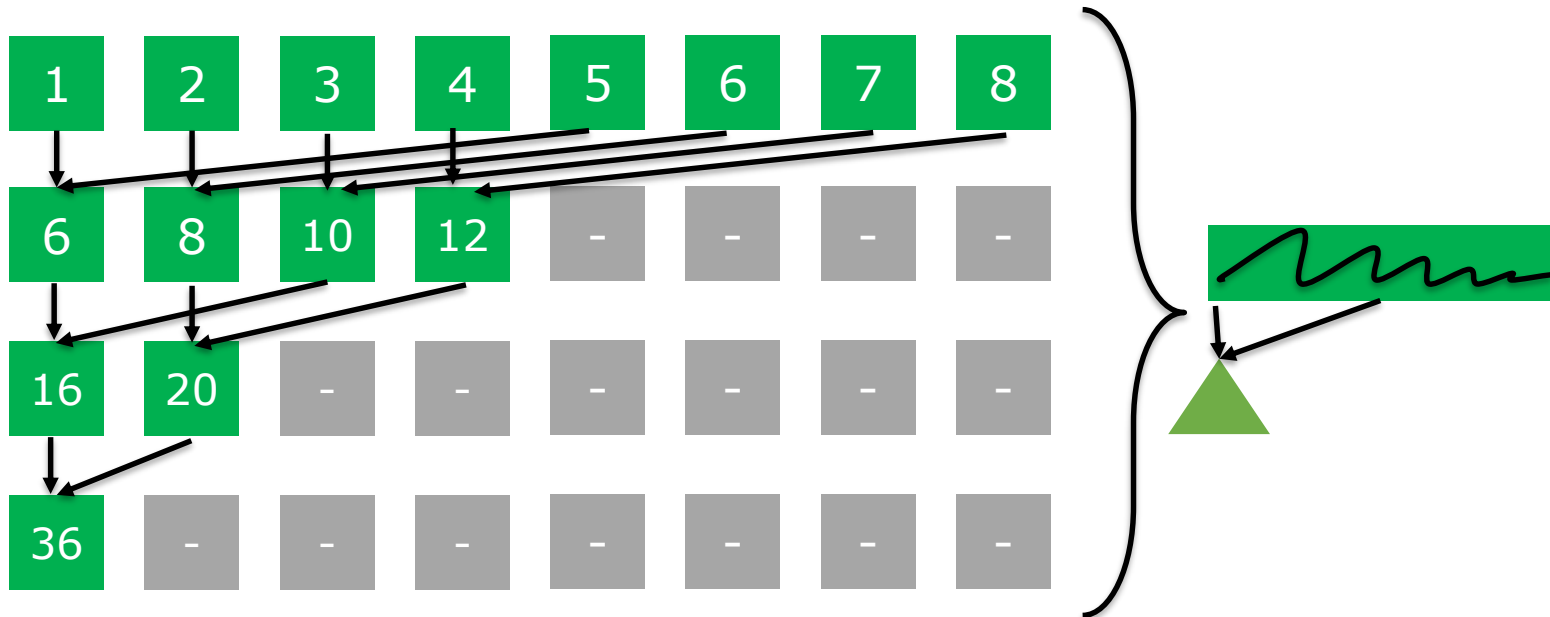
```
}
```

`__inline__` similar to C/C++ `inline`.
A hint to the compiler.



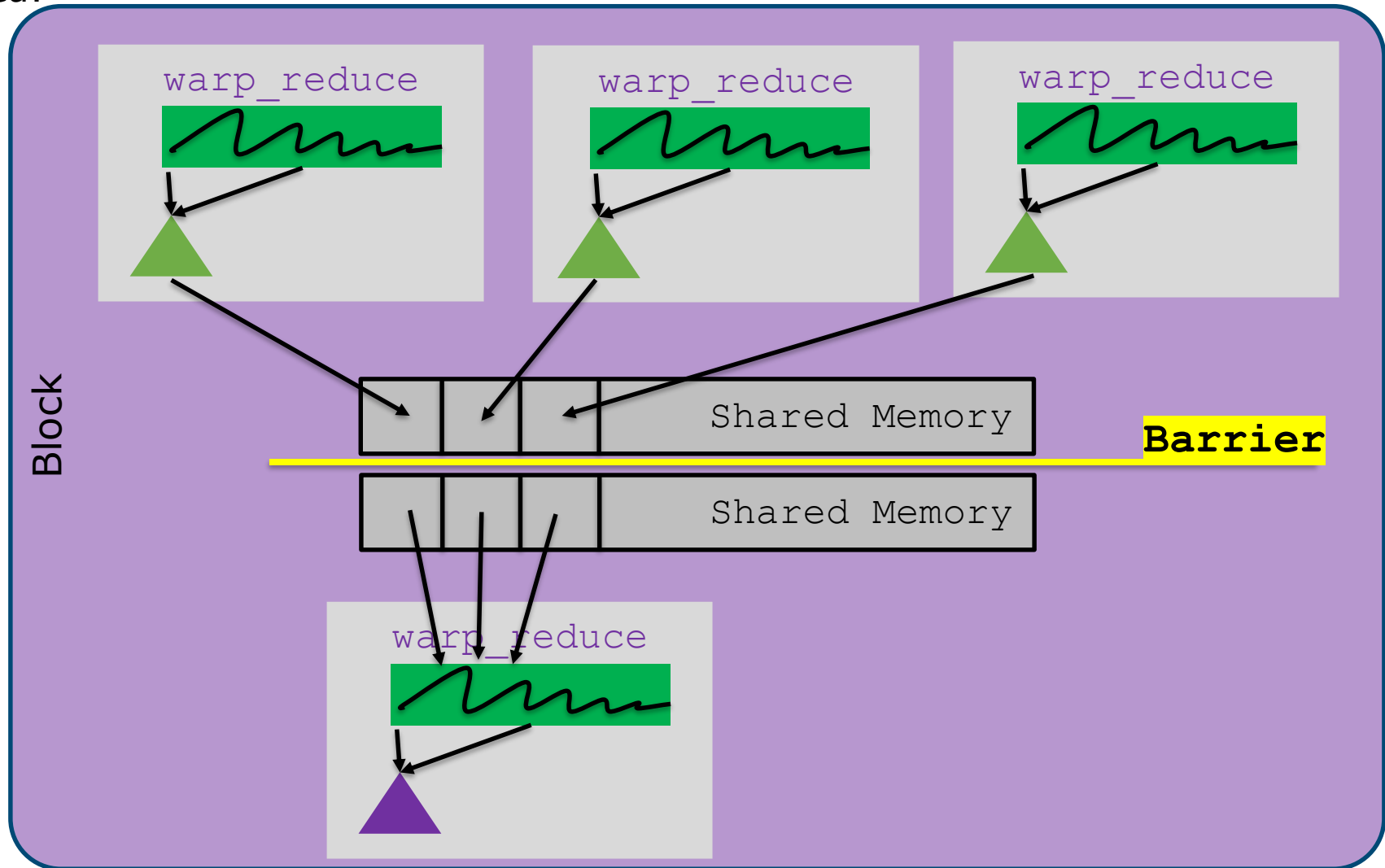
Reduction (block-level)

► Idea



Reduction (block-level)

► Idea:



Copyright

Copyright Bull, an Atos Company. All rights reserved.

Users Restricted Rights - Use, duplication or disclosure restricted.

Any copy of these documents should keep all copyright, logos and other proprietary notices contained herein.

This publication may include technical inaccuracies or typographical errors.

This publication is provided "AS IS" without any warranty either expressed or implied including but not limited to the implied warranties of merchantabilities or fitness of the described product.

Course Material Licensing Terms : No sublicensing rights.

For other licensing needs, please contact Bull, an Atos Company.

Thanks

For more information please contact:

Paul Karlshöfer

paul.karlshoefer@atos.net

Atos, the Atos logo, Atos Consulting, Atos Worldgrid, Worldline, BlueKiwi, Bull, Canopy the Open Cloud Company, Yunano, Zero Email, Zero Email Certified and The Zero Email Company are registered trademarks of the Atos group. September 2016. © 2016 Atos. Confidential information owned by Atos, to be used by the recipient only. This document, or any part of it, may not be reproduced, copied, circulated and/or distributed nor quoted without prior written approval from Atos.

29-10-2018