SIMD implementation

Example: exclusive scan 32-element array

32-wide GPU execution (SPMD program)

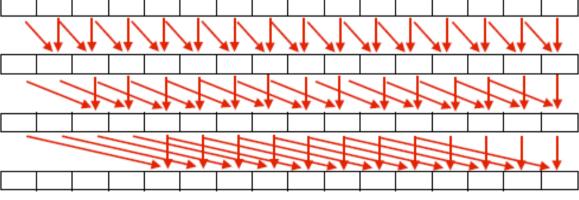
Exclusive result is returned (note: state of ptr[] is inclusive result) CUDA thread index

```
template < class OP, class T>
   __device__ T scan_warp(volatile T *ptr, const unsigned int idx)
{
   const unsigned int lane = idx & 31; // index of thread in warp (0..31)

   if (lane >= 1)    ptr[idx] = OP::apply(ptr[idx - 1],    ptr[idx]);
   if (lane >= 2)    ptr[idx] = OP::apply(ptr[idx - 2],    ptr[idx]);
   if (lane >= 4)    ptr[idx] = OP::apply(ptr[idx - 4],    ptr[idx]);
   if (lane >= 8)    ptr[idx] = OP::apply(ptr[idx - 8],    ptr[idx]);
   if (lane >= 16)    ptr[idx] = OP::apply(ptr[idx - 16],    ptr[idx]);

   return (lane>0) ? ptr[idx-1] : OP::identity();
}
```

Work: ??



• • •