```
__global__ void work_efficient_scan_kernel(float *X, float *Y, int InputSize) {

  __shared__ float XY[SECTION_SIZE];

  int i = blockIdx.x*blockDim.x + threadIdx.x;
  if (i < InputSize) {
    XY[threadIdx.x] = X[i];
  }

  for (unsigned int stride = 1; stride < blockDim.x; stride *= 2) {
    __syncthreads();
    int index = (threadIdx.x+1) * 2* stride -1;
    if (index < blockDim.x) {
      XY[index] += XY[index - stride];
    }
  }

  for (int stride = SECTION_SIZE/4; stride > 0; stride /= 2) {
    __syncthreads();
    int index = (threadIdx.x+1)*stride*2 - 1;
    if(index + stride < BLOCK_SIZE) {
      XY[index + stride] += XY[index];
    }
  }

  __syncthreads();

  Y[i] = XY[threadIdx.x];

}
```