

performance/compiler_optionen/intel/example_vec_report_stream

Example: Intel legacy compiler optimization report for benchmark stream

- stream source code snippet

```
/* ---
Tuned vector scale:  b[] = scalar * c[]

In:  STREAM_ARRAY_SIZE_thread, scalar, c[]
Out: b[]
--- */
void static inline tuned_STREAM_Scale(const STREAM_TYPE scalar) {
    #pragma omp parallel default(none) shared(scalar, STREAM_ARRAY_SIZE_thread)
    {
        #ifdef __INTEL_COMPILER
            // Instructs the compiler to use non-temporal (that is, streaming) stores
            #pragma vector nontemporal
        #endif
        #pragma omp simd aligned(b, c : alignment_bytes)
        for (long int j = 0; j < STREAM_ARRAY_SIZE_thread; j++) {
            b[j] = scalar * c[j]; // Line: 349
        }
    }
}

/* ---
Tuned vector add:  c[] = a[] + b[]

In:  STREAM_ARRAY_SIZE_thread, a[], b[]
Out: c[]
--- */
void static inline tuned_STREAM_Add() {
    #pragma omp parallel default(none) shared(STREAM_ARRAY_SIZE_thread)
```

```

    {
        #ifdef __INTEL_COMPILER
            // Instructs the compiler to use non-temporal (that is, streaming) stores
            #pragma vector nontemporal
        #endif
        #pragma omp simd aligned(a, b, c : alignment_bytes)
        for (long int j = 0; j < STREAM_ARRAY_SIZE_thread; j++) {
            c[j] = a[j] + b[j]; // Line: 369
        }
    }
}

```

- Prepare environment

```

module purge
module add compiler/intel/2022

```

- Compile benchmark with optimization report enabled

```

icc -std=c11 -Ofast -xHost -ipo -qopenmp \
    -qopt-report=5 \
    -qopt-report-phase=vec \
    -qopt-report-stdout \
    stream.OpenMP.c -o stream

```

- Output

```

...
LOOP BEGIN at stream.OpenMP.c(348,9) inlined into stream.OpenMP.c(679,5)
remark #15388: vectorization support: reference *b[j] has aligned access [ stream.
remark #15388: vectorization support: reference *c[j] has aligned access [ stream.
remark #15412: vectorization support: streaming store was generated for b [ stream.
remark #15305: vectorization support: vector length 4
remark #15309: vectorization support: normalized vectorization overhead 0.200
remark #15301: SIMD LOOP WAS VECTORIZED
remark #26013: Compiler has chosen to target XMM/YMM vector. Try using -qopt-zmm-usa
remark #15448: unmasked aligned unit stride loads: 1
remark #15449: unmasked aligned unit stride stores: 1
remark #15467: unmasked aligned streaming stores: 1
remark #15475: --- begin vector cost summary ---
remark #15476: scalar cost: 7
remark #15477: vector cost: 1.250
remark #15478: estimated potential speedup: 5.580
remark #15488: --- end vector cost summary ---
LOOP END

LOOP BEGIN at stream.OpenMP.c(348,9) inlined into stream.OpenMP.c(679,5)
<Remainder loop for vectorization>

```

```
...  
LOOP END
```

```
...  
LOOP BEGIN at stream.OpenMP.c(368,9) inlined into stream.OpenMP.c(680,5)  
  remark #15388: vectorization support: reference *c[j] has aligned access [ stream.  
  remark #15388: vectorization support: reference *a[j] has aligned access [ stream.  
  remark #15388: vectorization support: reference *b[j] has aligned access [ stream.  
  remark #15412: vectorization support: streaming store was generated for c [ stream.  
  remark #15305: vectorization support: vector length 4  
  remark #15301: SIMD LOOP WAS VECTORIZED  
  remark #26013: Compiler has chosen to target XMM/YMM vector. Try using -qopt-zmm-usa  
  remark #15448: unmasked aligned unit stride loads: 2  
  remark #15449: unmasked aligned unit stride stores: 1  
  remark #15467: unmasked aligned streaming stores: 1  
  remark #15475: --- begin vector cost summary ---  
  remark #15476: scalar cost: 8  
  remark #15477: vector cost: 1.250  
  remark #15478: estimated potential speedup: 6.400  
  remark #15488: --- end vector cost summary ---  
LOOP END
```

```
LOOP BEGIN at stream.OpenMP.c(368,9) inlined into stream.OpenMP.c(680,5)  
<Remainder loop for vectorization>  
...  
LOOP END
```

- Report on successful vectorization
- Report on data alignment
- Report on vector length
- Report on loads, stores and streaming store