Embedded Systems (ECE340) Project 3

Χριστόδουλος Ζερδαλής & Τσιαντός Δημήτριος

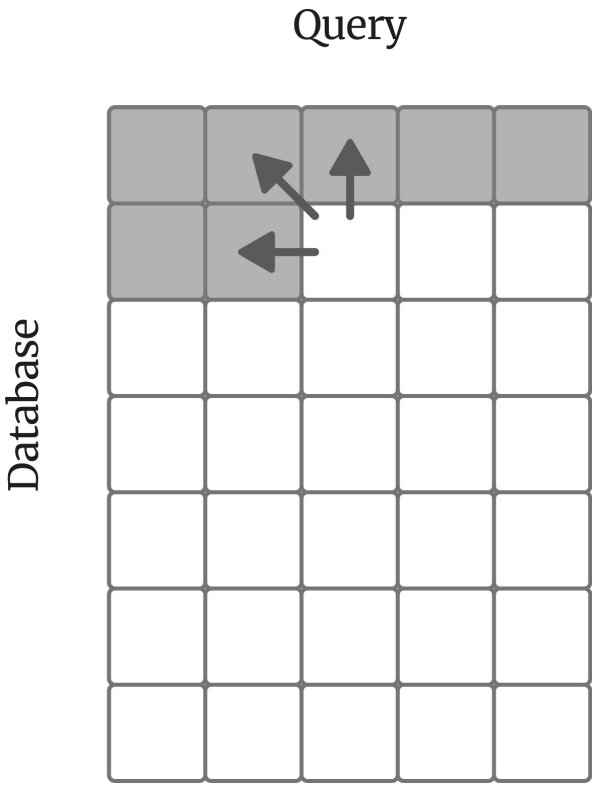
(3531 & 3796)

x86 Software implementation

Base implementation

Algorithm 1 Smith-Waterman Algorithm base

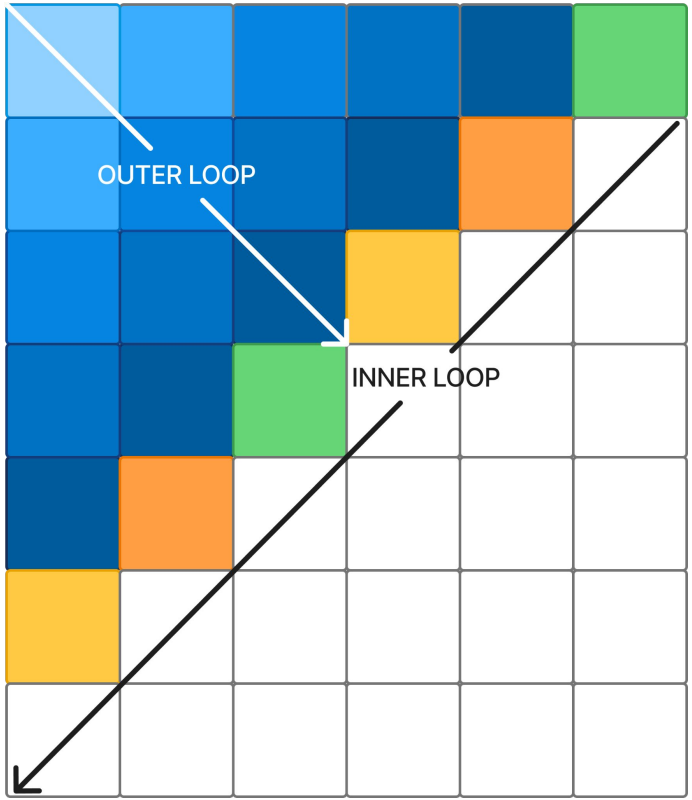
1: **for** $i \leftarrow 1$ to M **do**
2: **for** $j \leftarrow 1$ to N **do**
3: $\text{match} \leftarrow \begin{cases} \text{MATCH}, & \text{if } Q[j] = D[i] \\ \text{MISS}, & \text{else} \end{cases}$
4: $\text{North} \leftarrow S[i-1][j] - 1$
5: $\text{West} \leftarrow S[i][j-1] - 1$
6: $\text{Northwest} \leftarrow S[i-1][j-1] + \text{match}$
7: $S[i][j] \leftarrow \max(0, \text{North}, \text{West}, \text{Northwest})$
8: **if** $S[i][j] > \text{MAX}$ **then**
9: $\text{MAX} \leftarrow s[i][j]$
10: $\text{MAX\_POS} \leftarrow (i, j)$
11: **end if**
12: **end for**
13: **end for**



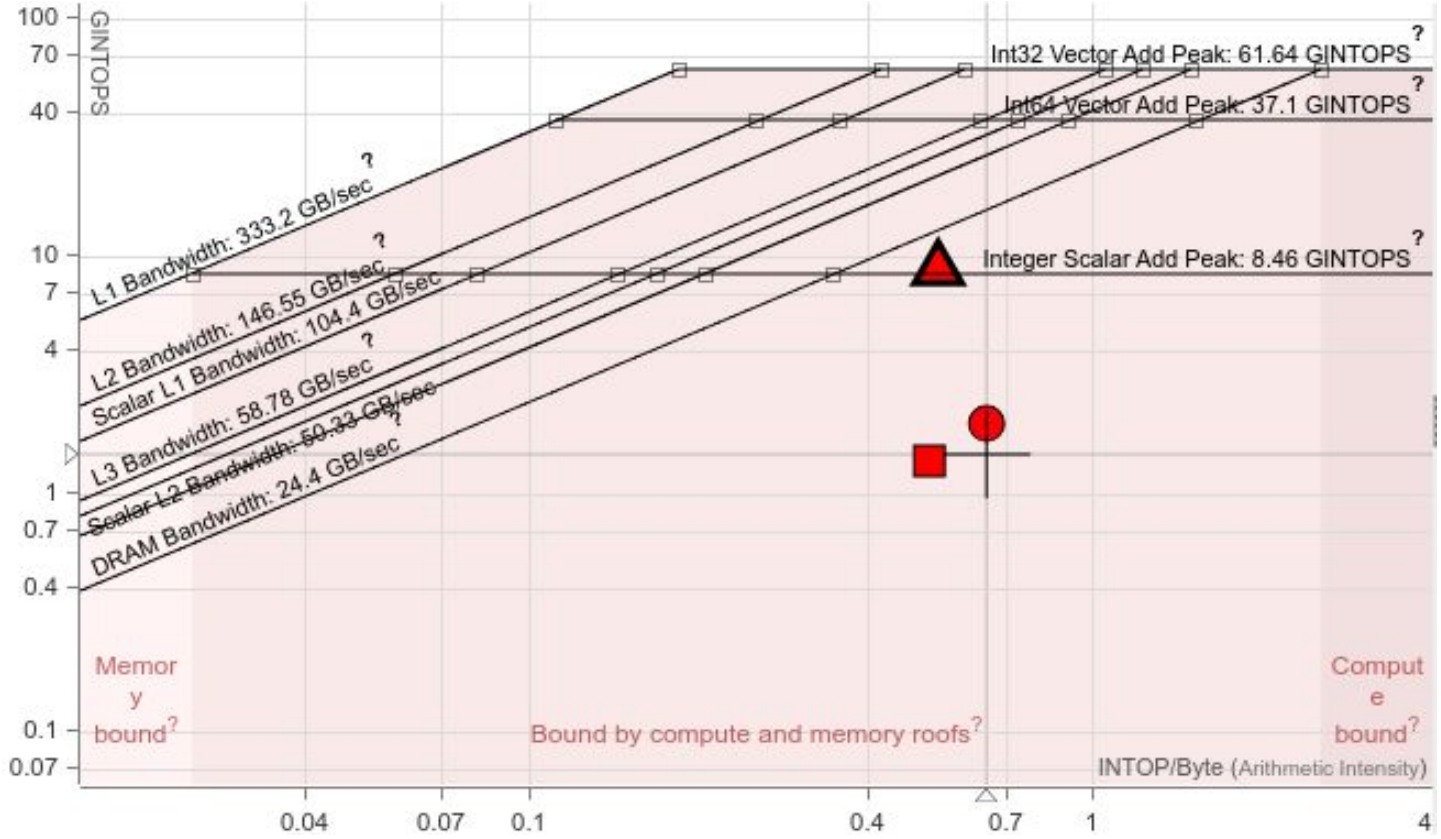
OpenMP implementation

Algorithm 2 Parallel Smith-Waterman with OpenMP

```
1:  #pragma omp parallel
2:  for sum ← 0 to M + N − 2 do
3:      Set k ← max(1, sum − N + 1)
4:      #pragma omp for
5:      for i ← k to N + k − 1 do
6:          Set j ← sum − i
7:          ...
8:          if MAX > local_max then
9:              L_MAX ← MAX
10:             L_MAX_index ← (i, j)
11:         end if
12:     end for
13: end for
14: #pragma omp critical
15: if L_MAX > G_MAX then
16:     G_MAX ← L_MAX
17:     MAX_index ← L_MAX_index
18: end if
```



Roofline model



ARM Software implementation

Roofline model

Roof y-axis:

$$\text{Peak INTOPS/s} = f_{\text{CPU}} \times \left(\frac{\#\text{Integer Ops}}{\text{Cycle}} \right)$$

Roof x-axis:

$$\text{RAM Bandwidth} = \text{Bus Width (Bytes)} \times f_{\text{RAM}} \times 2$$

Roofline model

Roof y-axis:

$$\text{Peak INTOPS} = 0.667 \times 2 = 1.334 \text{ GINTOPS}$$

Roof x-axis:

$$\text{Bandwidth} = 8 \text{ bytes} \times 867 \times 10^6 \text{ Hz} \times 2 = 13.872 \times 10^9 \text{ bytes/s} = 13.872 \text{ GB/s}$$

Roofline model

Program y-axis:

$$\text{Operational Intensity} = \frac{\#\text{Integer Instructions}}{\#\text{Memory Loads} \times \text{Bus Width (Bytes)}}$$

Program x-axis:

$$\text{GINTOPS} = \frac{\#\text{Integer Instructions}}{10^9 \times \text{Total Time (seconds)}}$$

Roofline model

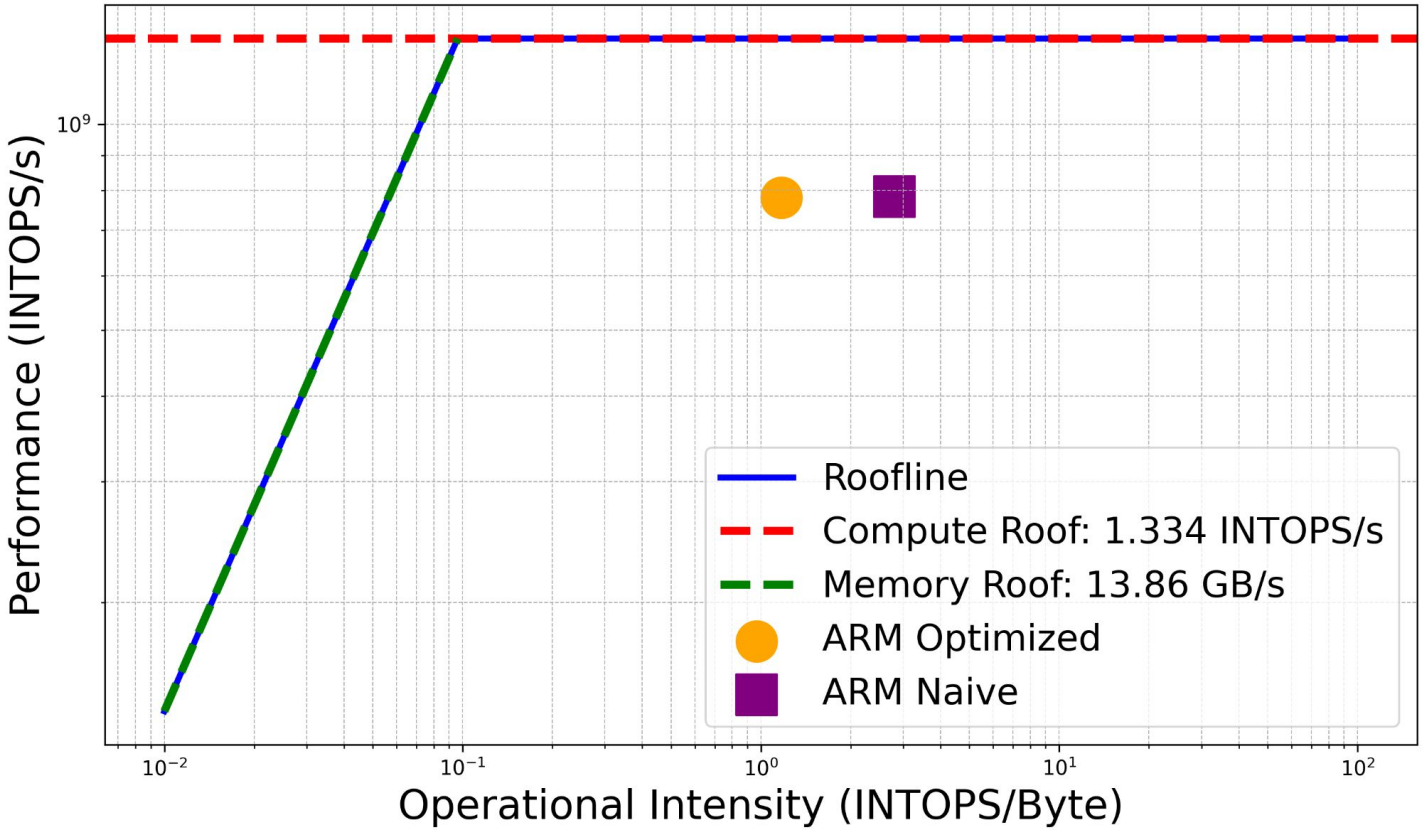
Program y-axis:

$$\mathrm{OI} = \frac{\#\mathrm{Instructions}}{\#\mathrm{Loads} \times \mathrm{Bus\ Width}} = \frac{3{,}099{,}341{,}072}{662{,}432{,}510 \times 8} = 0.585\ \mathrm{INTOPS/Byte}$$

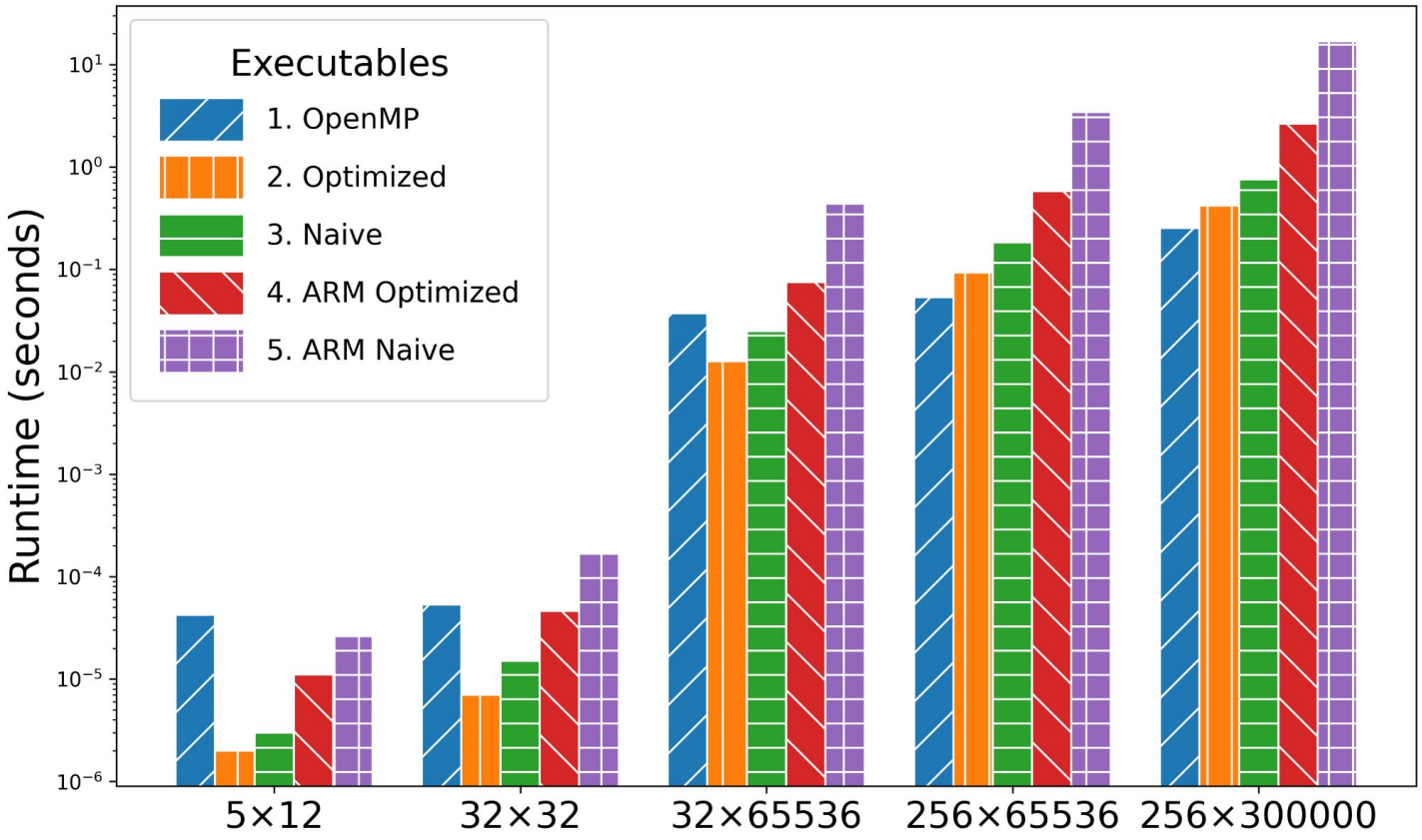
Program x-axis:

$$\mathrm{GINTOPS} = \frac{\#\mathrm{Instructions}}{10^9 \times \mathrm{Time}} = \frac{3{,}099{,}341{,}072}{10^9 \times 3.965} \approx 0.782\ \mathrm{GINTOPS}$$

Roofline model



Execution time comparisons



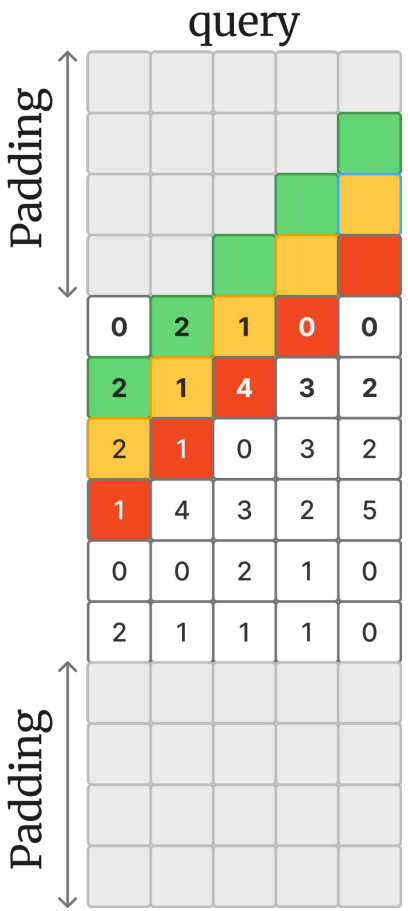
Hardware implementation

Base implementation on FPGA

Latency on FPGA (ms)	4562.5097
Loop Name	OUTER_LOOP
Min Latency (cycles)	297088813
Max Latency (cycles)	435367289
Latency Range	4529 – 6637
Iteration Latency	–
Initiation Achieved	–
Initiation Target	–
Trip Count	65597
Pipelined	no

First hardware optimizations

- Remove **similarity matrix**
- Add **buffers**
- Add **padding**
- **Memory** optimizations
- **Pipeline** and **unrolling**



First hardware optimizations

```
1: for k ← 0 to M − N do (Outer_loop)
2:     #pragma HLS PIPELINE
3:     for i ← 0 to N − 1 do (Inner_loop)
4:         j ← N − 1 − i
5:         match ← (query_buff[j] = database_buff[i + k])?MATCH : MISS_MATCH
6:         North ← diag_array_2[i] − 1
7:         Northwest ← match + diag_array_1[i + 1]
8:         West ← diag_array_2[i + 1] − 1
9:         ...
10:        Direction_buff[i] ← Direction
11:        Diag_array_3[i] ← Max_value
12:        ...
13:    end for
14:    Copy diag_array_2 to diag_array_1
15:    Copy diag_array_3 to diag_array_2
16:    for i ← 0 to N − 1 do (Dir_loop)
17:        j ← N − 1 − i
18:        direction_matrix[(i + k) × N + j] ← direction_buff[i]
19:    end for
20: end for
```

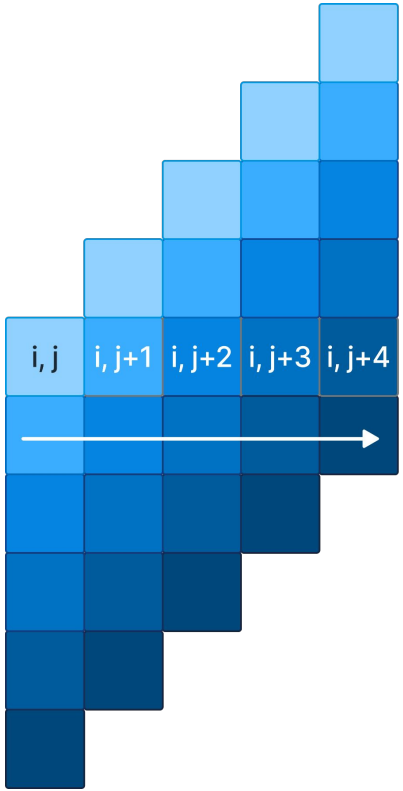
Memory optimizations

```
#pragma HLS ARRAY_PARTITION diag_array_1 dim=1 factor=32 cyclic
#pragma HLS BIND_STORAGE diag_array_1 type=ram_t2p impl=bram

#pragma HLS ARRAY_PARTITION diag_array_2 dim=1 factor=32 cyclic
#pragma HLS BIND_STORAGE diag_array_2 type=ram_t2p impl=bram

#pragma HLS ARRAY_PARTITION diag_array_3 dim=1 factor=32 cyclic
#pragma HLS BIND_STORAGE diag_array_3 type=ram_t2p impl=bram

#pragma HLS ARRAY_PARTITION query_buff dim=1 factor=16 cyclic
#pragma HLS BIND_STORAGE query_buff type=ram_1wnr impl=bram

#pragma HLS ARRAY_PARTITION database_buff dim=1 factor=16 cyclic
#pragma HLS BIND_STORAGE database_buff type=ram_1wnr impl=bram

#pragma HLS ARRAY_PARTITION direction_buff dim=1 factor=16 cyclic
#pragma HLS BIND_STORAGE direction_buff type=ram_t2p impl=bram
```

First hardware optimizations

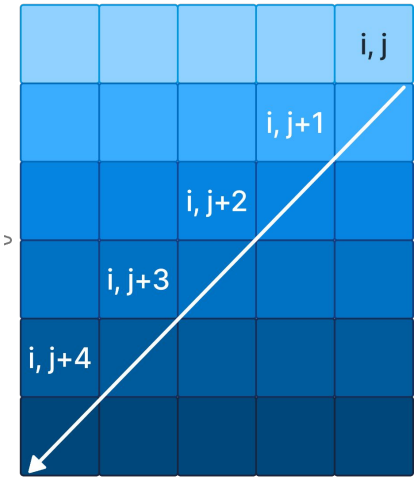
Latency on FPGA (ms)	21.641
Loop Name	OUTER_LOOP
Min Latency (cycles)	2,098,215
Max Latency (cycles)	2,098,215
Iteration Latency	104
Initiation Achieved	32
Initiation Target	1
Trip Count	65,567
Pipelined	yes

Flattening the direction array



for $i \leftarrow 0$ to $N-1$ **do** (Dir_loop)
    $j \leftarrow N-1-i$
    direction_matrix$[(i+k) \times N + j] \leftarrow$ direction_buff$[i]$
end for

Flattening the direction array



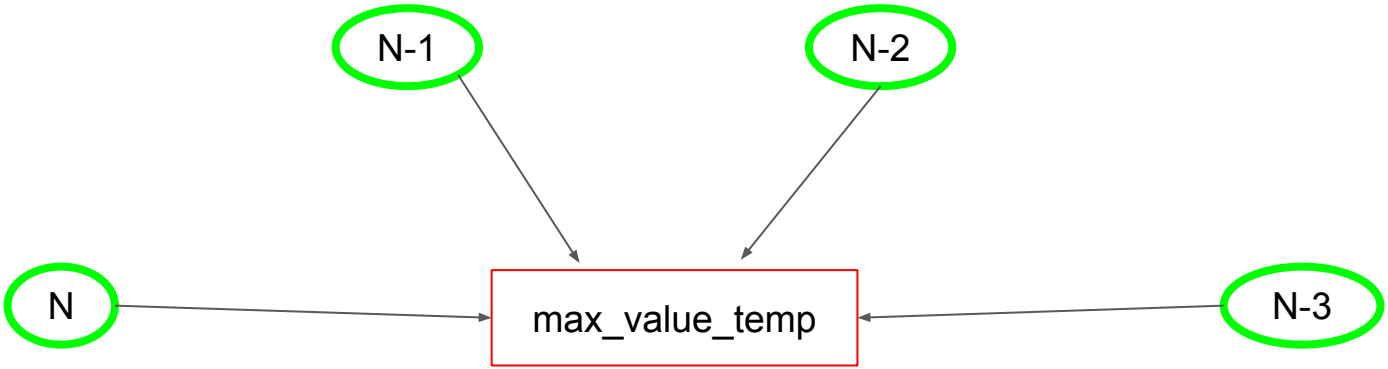
for $j \leftarrow 0$ to $N-1$ **do** (Dir_loop)
    direction_matrix$[i * N + j] \leftarrow$ direction_buff$[j]$
end for

Flattening the direction array

Latency on FPGA (ms)	12.8369
Loop Name	OUTER_LOOP
Min Latency (cycles)	1,049,075
Max Latency (cycles)	1,049,075
Iteration Latency	20
Initiation Achieved	16
Initiation Target	1
Trip Count	65,567
Pipelined	yes

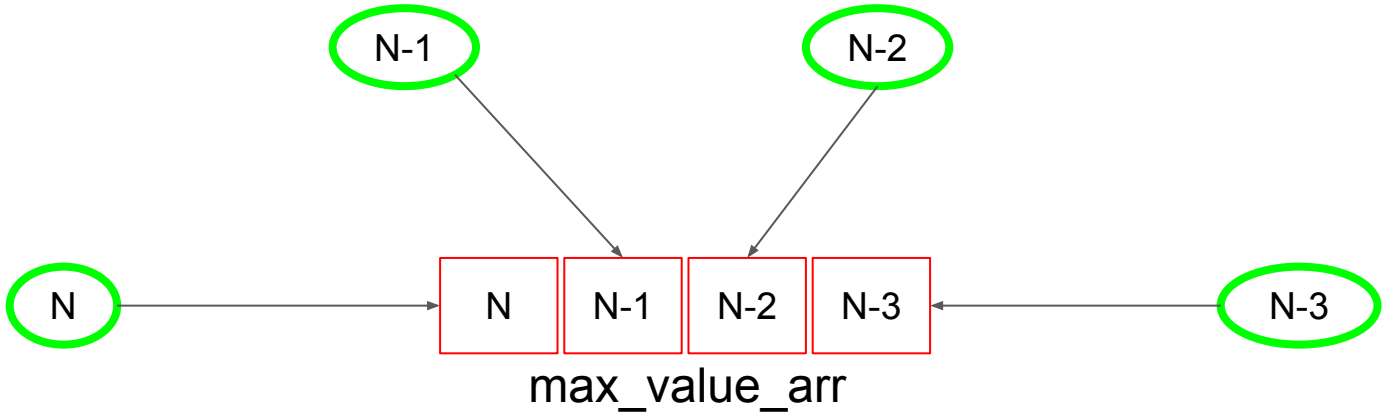
Adding max value array

if max_value $>$ max_value_temp **then**
    max_value_temp $\leftarrow$ max_value
    max_index_temp $\leftarrow ((i + k) * N + j)$
end if



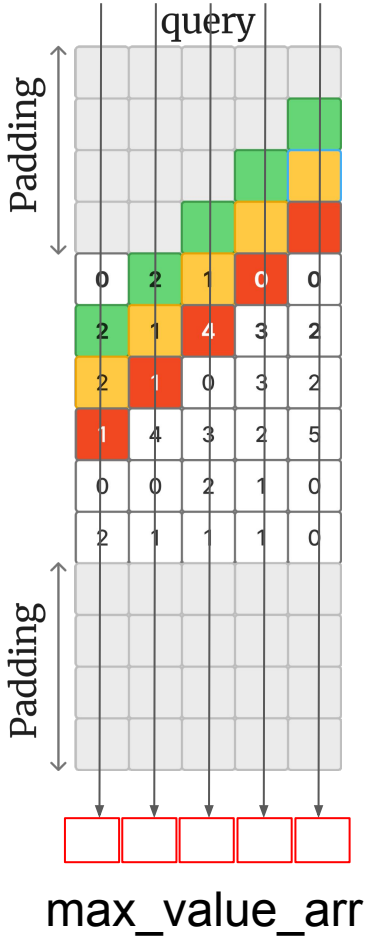
Adding max value array

if max_value $>$ max_value_arr[i] **then**
    max_value_arr[i] $\leftarrow$ max_value
    max_index_arr[i] $\leftarrow ((i + k) * N + j)$
end if



Adding max value array

if max_value $>$ max_value_arr[i] **then**
    max_value_arr[i] $\leftarrow$ max_value
    max_index_arr[i] $\leftarrow ((i + k) * N + j)$
end if



Adding max value array

```
for i ← 0 to N − 1 do (MAX_LOOP)
    if max_value_arr[i] > max_value_temp then
        max_value_temp ← max_value_arr[i]
        max_idx_temp ← max_index_arr[i]
    end if
end for
*max_index ← max_idx_temp
```

Only one write to main memory

Adding max value array

```
#pragma HLS ARRAY_PARTITION variable=max_value_arr dim=1 factor=16 cyclic
#pragma HLS BIND_STORAGE variable=max_value_arr type=ram_t2p impl=bram

#pragma HLS ARRAY_PARTITION variable=max_index_arr dim=1 factor=16 cyclic
#pragma HLS BIND_STORAGE variable=max_index_arr type=ram_t2p impl=bram
```

Adding max value array

Latency on FPGA (ms)	4.96728
Loop Name	OUTER_LOOP
Min Latency (cycles)	262,268
Max Latency (cycles)	262,268
Iteration Latency	4
Initiation Achieved	4
Initiation Target	1
Trip Count	65,567
Pipelined	yes

Optimizing if statements

```
1:  max_value ← north
2:  direction ← NORTH
3:  if northwest > max_value then
4:      max_value ← northwest
5:      direction ← NORTH_WEST
6:  end if
7:  if west > max_value then
8:      max_value ← west
9:      direction ← WEST
10: end if
11: if max_value ≤ 0 then
12:     max_value ← 0
13:     direction ← CENTER
14: end if
```

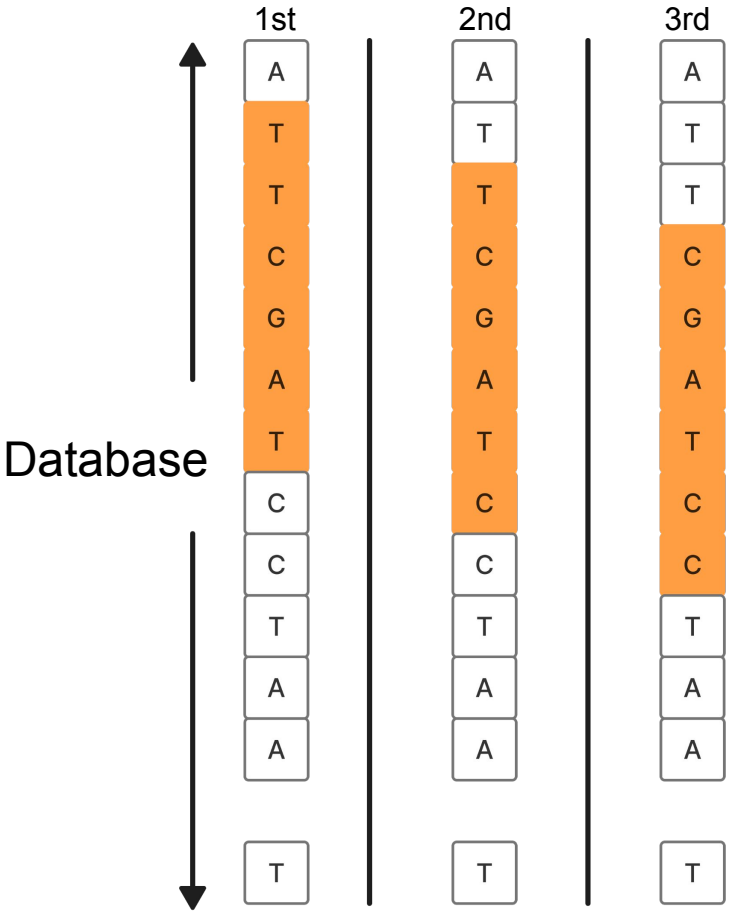
Many if statements

Optimizing if statements

```
1: if north < northwest and northwest < west and west ≠ −1 then
2:     max_value ← west
3:     direction ← WEST
4: else if north < northwest and northwest ≠ −1 then
5:     max_value ← northwest
6:     direction ← NORTH_WEST
7: else if north < west and west ≠ −1 then
8:     max_value ← west
9:     direction ← WEST
10: else if north = −1 then
11:     max_value ← 0
12:     direction ← CENTER
13: else
14:     max_value ← north
15:     direction ← NORTH
16: end if
```

One if-else-if chain

Database Buffer Shift




```
1: for k ← 0 to M − N do
2:     for i ← 1 to N − 1 do
3:         database_buff[i − 1] ← database_buff[i]
4:     end for
5:     database_buff[N − 1] ← database[k + N − 1]
6:     for i ← 0 to N − 1 do
7:         // ... inner loop body
8:     end for
9: end for
```

Database Buffer Shift

Latency on FPGA (ms)	3.78776
Loop Name	OUTER_LOOP
Min Latency (cycles)	196,841
Max Latency (cycles)	196,841
Iteration Latency	144
Initiation Achieved	3
Initiation Target	1
Trip Count	65,567
Pipelined	yes

Why did the Iteration Interval dropped?

Latency on FPGA (ms)	4.96728
Loop Name	OUTER_LOOP
Min Latency (cycles)	262,268
Max Latency (cycles)	262,268
Iteration Latency	4
Initiation Achieved	4
Initiation Target	1
Trip Count	65,567
Pipelined	yes

Latency on FPGA (ms)	3.78776
Loop Name	OUTER_LOOP
Min Latency (cycles)	196,841
Max Latency (cycles)	196,841
Iteration Latency	144
Initiation Achieved	3
Initiation Target	1
Trip Count	65,567
Pipelined	yes

Why did the Initiation dropped?

II=4

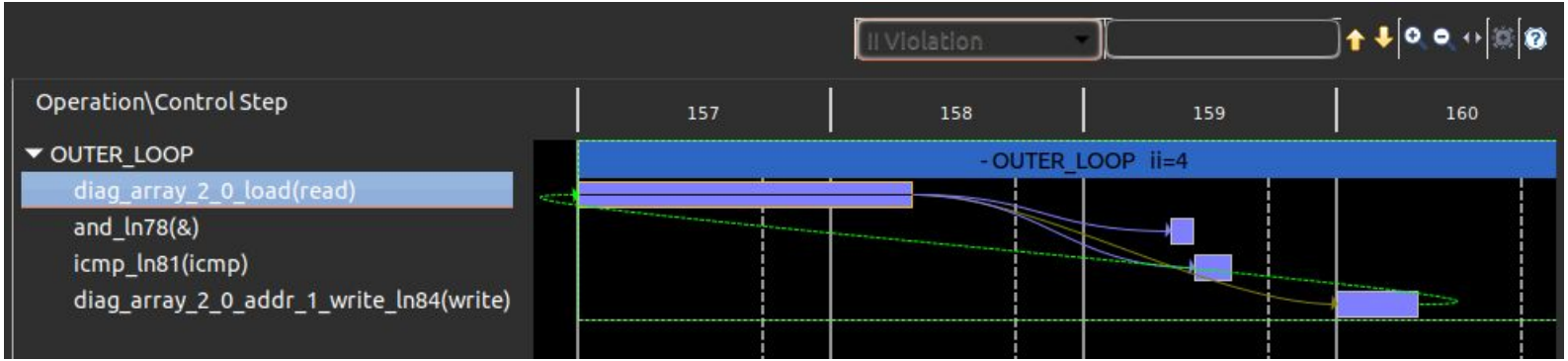
[HLS 200-880]	LINK	The II Violation in module 'compute_matrices' (loop 'OUTER_LOOP'): Unable to enforce a carried dependence constraint (II = 1, distance = 1, offset = 1) between 'store' operation ('diag_array_2_0_addr_1_write_ln84', diag_dir_max/lsal.cpp:84) of variable 'diag_array_3_load_0', diag_dir_max/lsal.cpp:84 on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17 and 'load' operation ('diag_array_2_0_load', diag_dir_max/lsal.cpp:65) on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17.
[HLS 200-880]	LINK	The II Violation in module 'compute_matrices' (loop 'OUTER_LOOP'): Unable to enforce a carried dependence constraint (II = 2, distance = 1, offset = 1) between 'store' operation ('diag_array_2_0_addr_1_write_ln84', diag_dir_max/lsal.cpp:84) of variable 'diag_array_3_load_0', diag_dir_max/lsal.cpp:84 on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17 and 'load' operation ('diag_array_2_0_load', diag_dir_max/lsal.cpp:65) on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17.
[HLS 200-880]	LINK	The II Violation in module 'compute_matrices' (loop 'OUTER_LOOP'): Unable to enforce a carried dependence constraint (II = 3, distance = 1, offset = 1) between 'store' operation ('diag_array_2_0_addr_1_write_ln84', diag_dir_max/lsal.cpp:84) of variable 'diag_array_3_load_0', diag_dir_max/lsal.cpp:84 on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17 and 'load' operation ('diag_array_2_0_load', diag_dir_max/lsal.cpp:65) on array 'diag_array_2[0]', diag_dir_max/lsal.cpp:17.

II=3

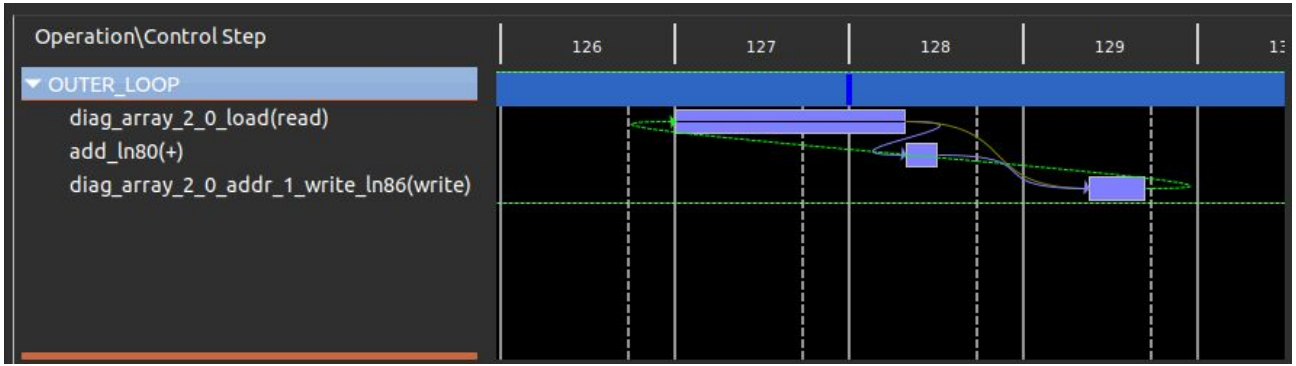
[HLS 200-880]	LINK	The II Violation in module 'compute_matrices' (loop 'OUTER_LOOP'): Unable to enforce a carried dependence constraint (II = 1, distance = 1, offset = 1) between 'store' operation ('diag_array_2_0_addr_1_write_ln86', storage/lsal.cpp:86) of variable 'diag_array_3_load_0', storage/lsal.cpp:86 on array 'diag_array_2[0]', storage/lsal.cpp:17 and 'load' operation ('diag_array_2_0_load', storage/lsal.cpp:80) on array 'diag_array_2[0]', storage/lsal.cpp:17.
[HLS 200-880]	LINK	The II Violation in module 'compute_matrices' (loop 'OUTER_LOOP'): Unable to enforce a carried dependence constraint (II = 2, distance = 1, offset = 1) between 'store' operation ('diag_array_2_0_addr_1_write_ln86', storage/lsal.cpp:86) of variable 'diag_array_3_load_0', storage/lsal.cpp:86 on array 'diag_array_2[0]', storage/lsal.cpp:17 and 'load' operation ('diag_array_2_0_load', storage/lsal.cpp:80) on array 'diag_array_2[0]', storage/lsal.cpp:17.

Why did the Initiation dropped?

II=4



II=3

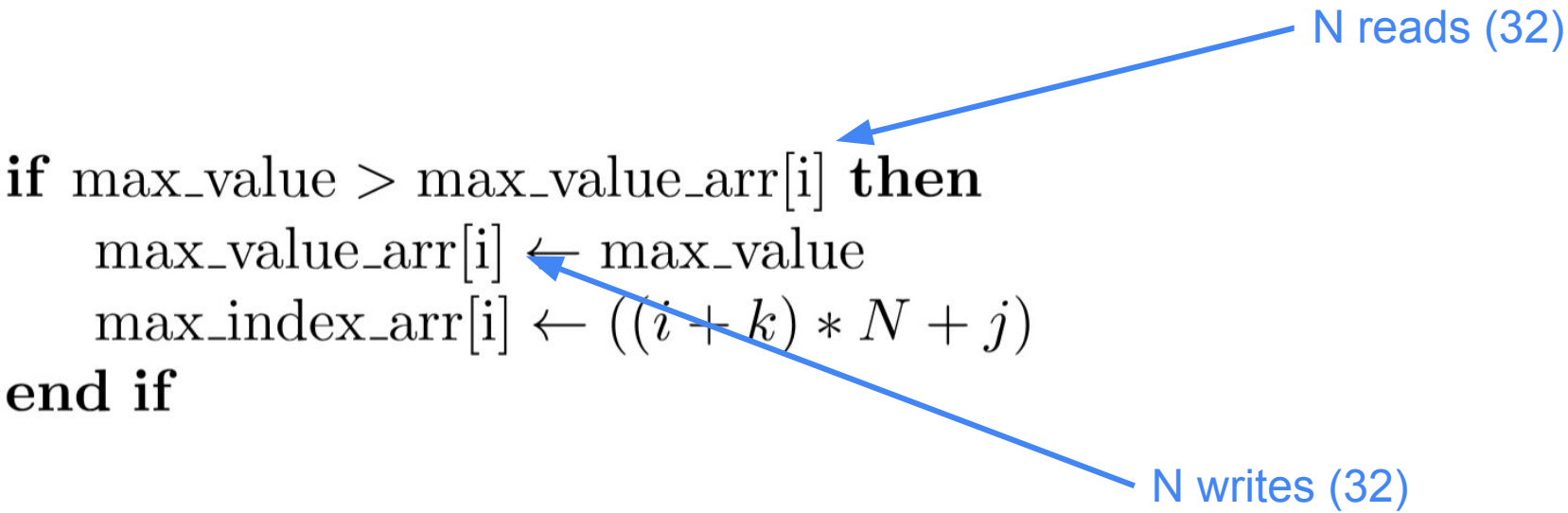


Diagonal rotation

```
1: for k ← 0 to M − N do
2:     for i ← 0 to N − 1 do
3:         // ... inner loop body
4:     end for
5:     Copy diag_array_2 into diag_array_1
6:     Copy diag_array_3 into diag_array_2
7: end for
```

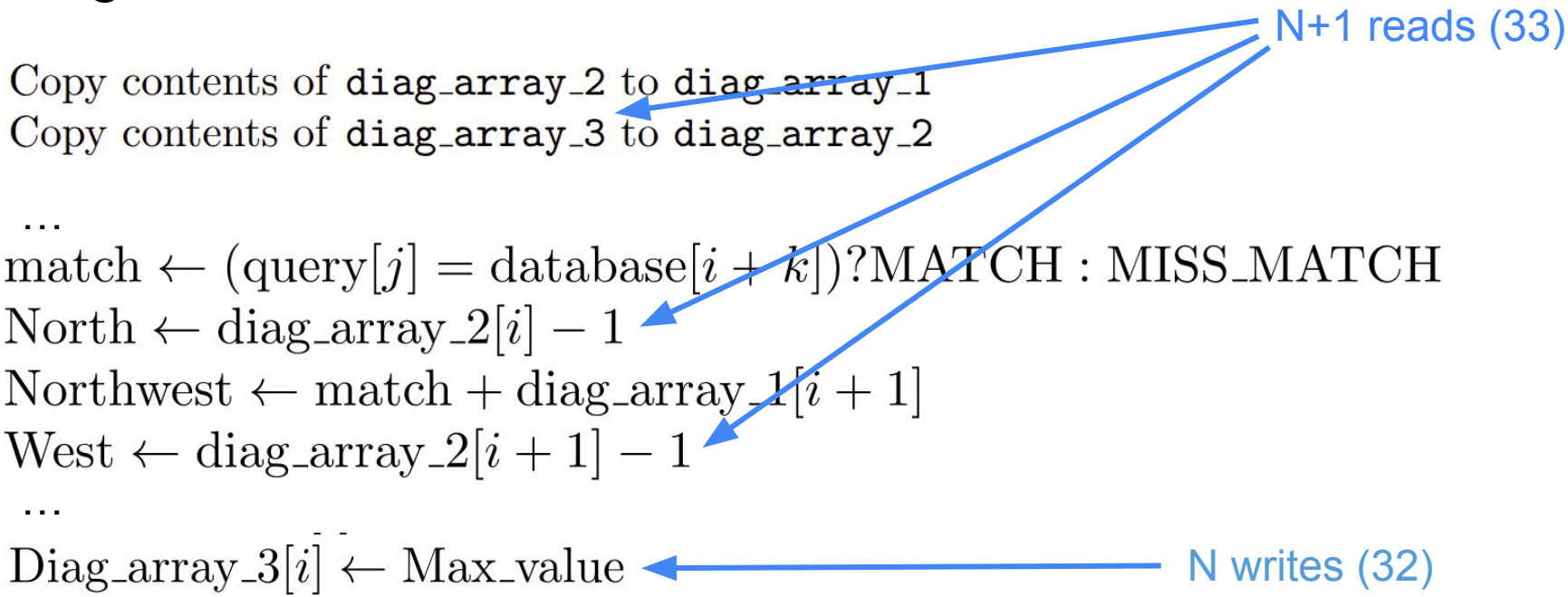
```
1: for k ← 0 to M − N do
2:     Copy diag_array_2 into diag_array_1
3:     Copy diag_array_3 into diag_array_2
4:     for i ← 0 to N − 1 do
5:         // ... inner loop body
6:     end for
7: end for
```

Diagonal rotation



We have ram_t2p cyclic 16 : NOT ENOUGH

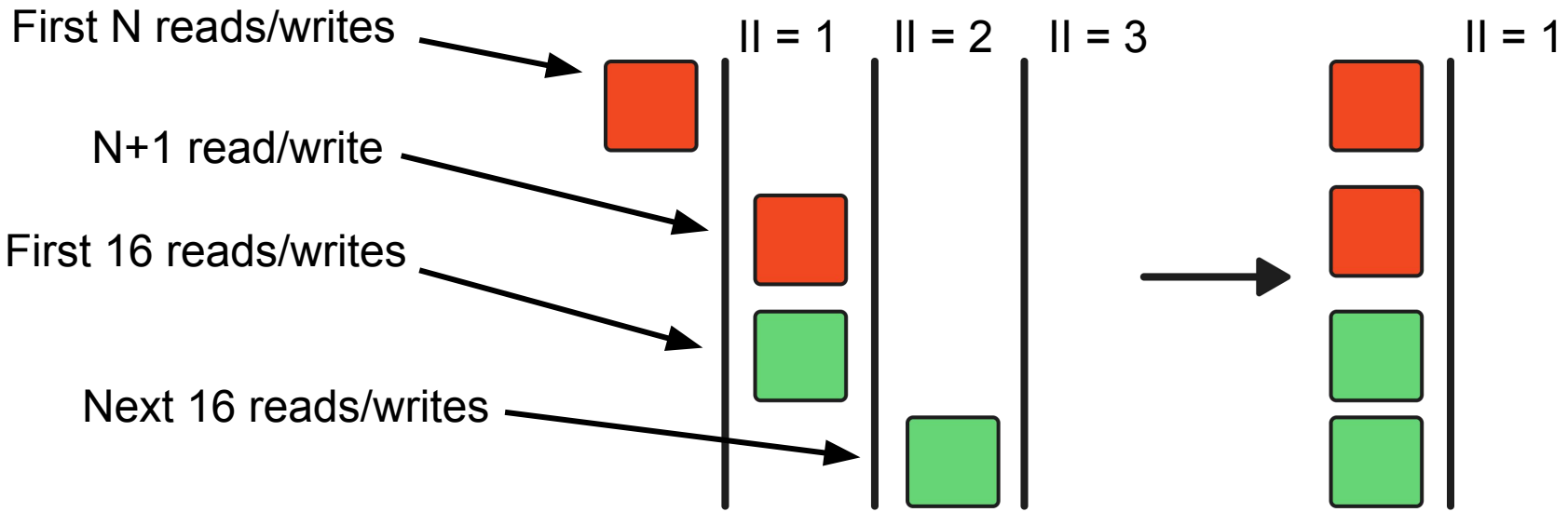
Diagonal rotation



We have ram_t2p cyclic 32 : NOT ENOUGH

Diagonal rotation

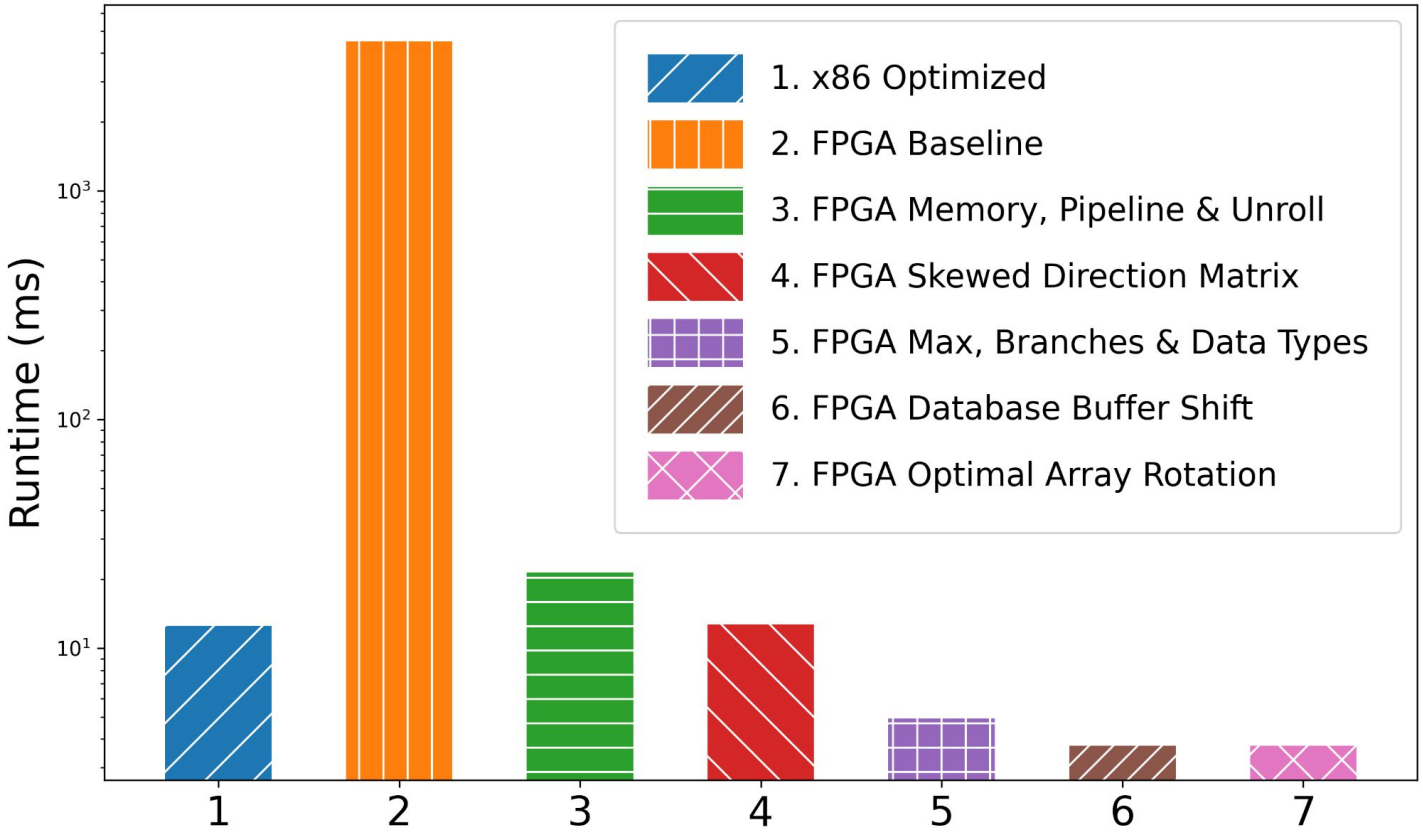
```
#pragma HLS ARRAY_PARTITION diag_array_3 dim=1 complete

#pragma HLS ARRAY_PARTITION max_value_arr dim=1 complete
```



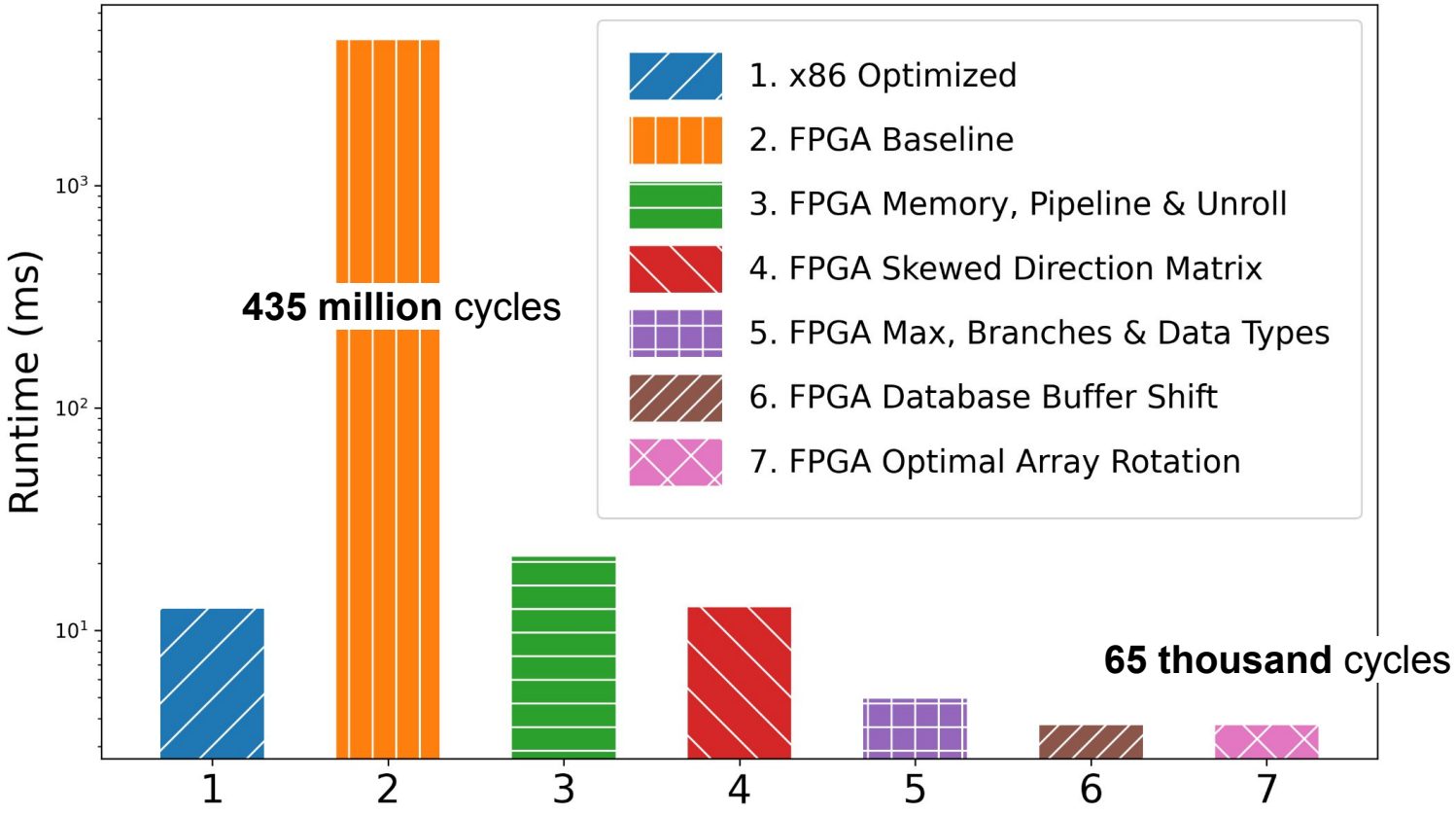
Diagonal rotation

Latency on FPGA(ms)	3.77532
Loop Name	OUTER_LOOP
Min Latency (cycles)	65,740
Max Latency (cycles)	65,740
Iteration Latency	175
Initiation Achieved	1
Initiation Target	1
Trip Count	65,567
Pipelined	yes

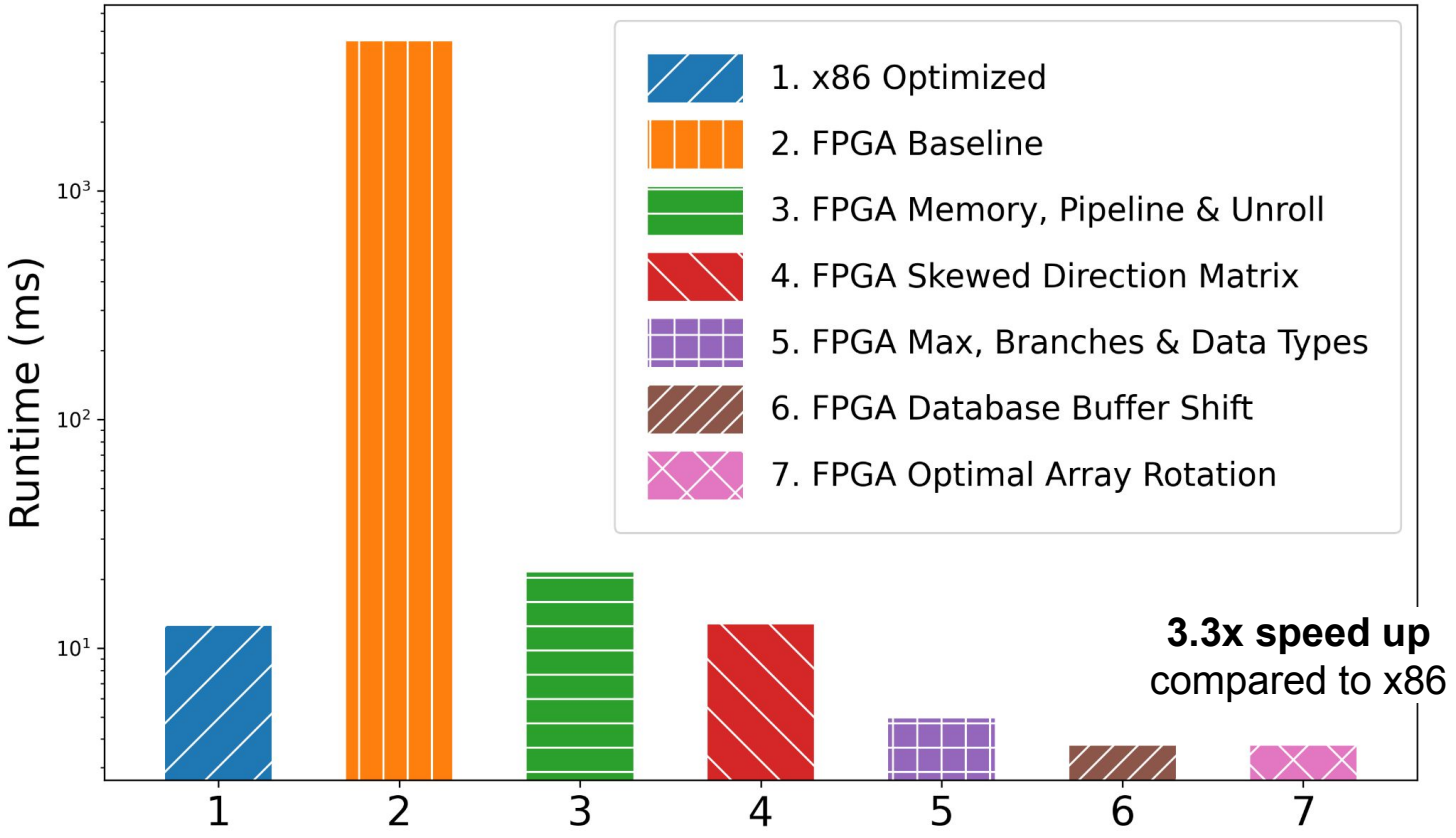
Time improvement with different optimizations

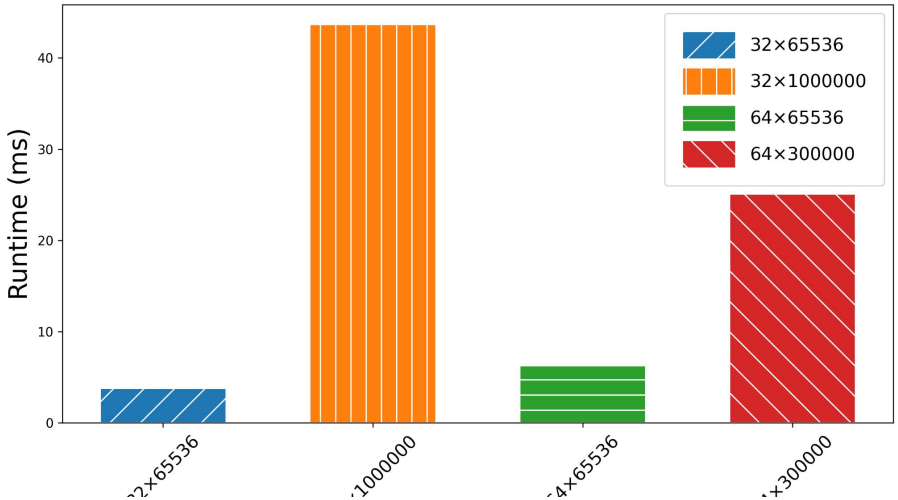


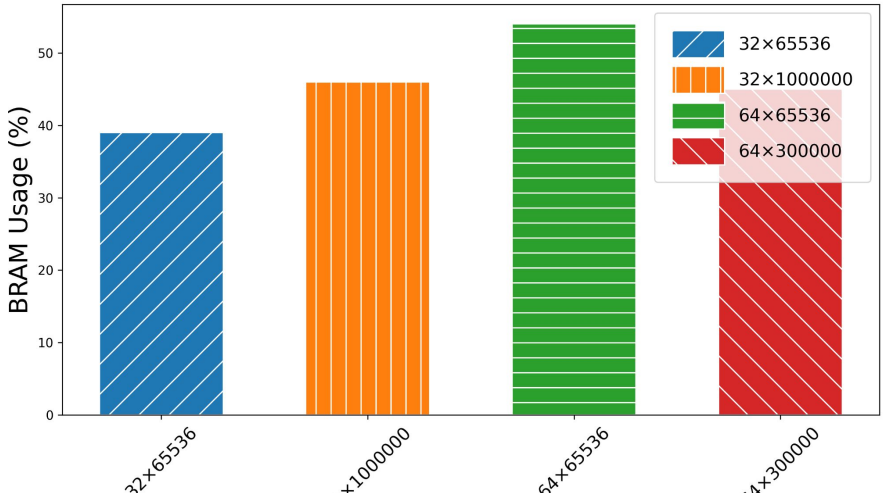
Time improvement with different optimizations

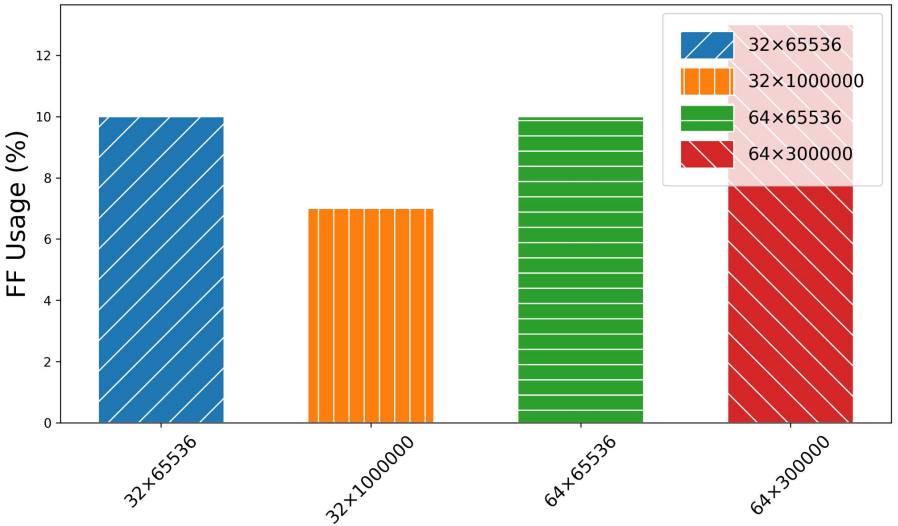


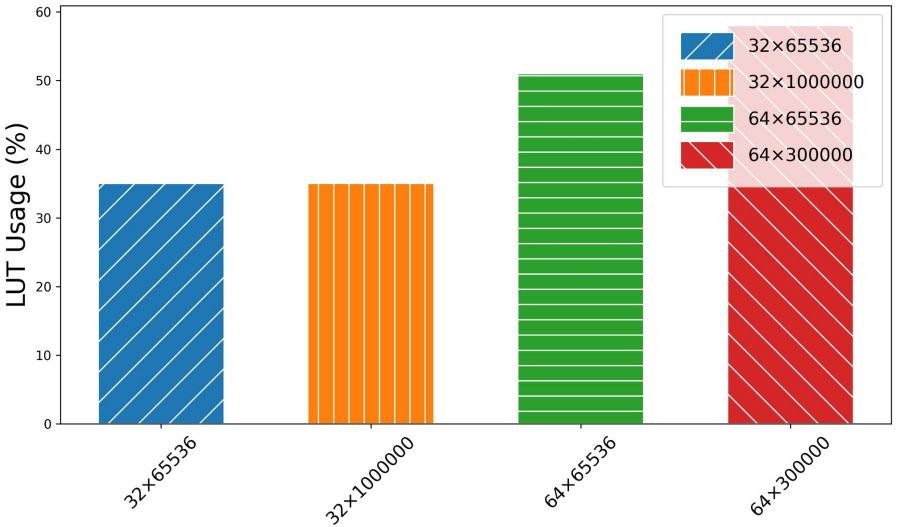
Time improvement with different optimizations




Runtime (ms)
0
10
20
30
40
32×65536
32×1000000
64×65536
64×300000


BRAM Usage (%)
0
10
20
30
40
50
32×65536
32×1000000
64×65536
64×300000


FF Usage (%)
0
2
4
6
8
10
12
32×65536
32×1000000
64×65536
64×300000


LUT Usage (%)
0
10
20
30
40
50
60
32×65536
32×1000000
64×65536
64×300000