An-Najah National University
Computer Engineering Department

Computer Architecture II HW #1
Loop Unrolling

Loop unrolling is a compiler optimization technique designed to enhance the execution speed of repetitive loops. In this study, we investigated the effects of loop unrolling on the performance of a simple loop algorithm. By varying the unrolling degree and array size, we aimed to understand the interplay between these factors and the resulting impact on running time and speedup.

Using a C++ programming , we implemented loop unrolling with degrees of 4, 8, 16, and 64. We meticulously tested three different array sizes: 1024, 1024*1024, and 16*1024*1024. Precise timing functions were employed to measure the execution time of both the unoptimized (normal) and optimized (unrolled) loops.

We run code 12 times for every size, and exclude max and min values, then take average of 10 values to increase accuracy.

Results:

		Unrolling Time			
Input-size	**Normal**	**Degree-4**	**Degree-8**	**Degree-16**	**Degree-64**
1024	0.000010	0.000003	0.0000031	0.0000037	0.0000043
1024*1024	0.0020429	0.0008862	0.0007872	0.0007619	0.0007875
16*1024*1024	0.0327831	0.0154541	0.0148363	0.0144612	0.0144381

		Unrolling Speedup			
Input-size	**Normal**	**Degree-4**	**Degree-8**	**Degree-16**	**Degree-64**
1024	-	3.333	3.225	2.702	2.325
1024*1024	-	2.305	2.595	2.681	2.594
16*1024*1024	-	2.121	2.209	2.266	2.270

Discussion:

Impact of Unrolling Degree: For smaller arrays (1024), unrolling the loop with a degree of 4 resulted in a notable speedup, increasing the processing speed by approximately 3.33 times compared to the normal loop. However, the speedup gains diminished as the unrolling degree increased. Higher degrees such as 16 and 64 offered only marginal improvements, indicating diminishing returns in optimization efforts for smaller input sizes.

Effect of Array Size: Larger arrays (1024*1024 and 16*1024*1024) exhibited more significant improvements with higher unrolling degrees. For example, with a degree of 16, the processing speed for 1024*1024 array increased by approximately 2.68 times, and for 16*1024*1024 array, the speedup was around 2.27 times compared to the normal loop. These results suggest that loop unrolling becomes particularly advantageous for processing large datasets, where the impact on performance is more pronounced.

Conclusion:

Clearly, that the impact of loop unrolling on performance is contingent on both the size of the array and the degree of unrolling. For smaller datasets, moderate unrolling degrees suffice to yield noticeable improvements. However, for larger datasets, especially those encountered in real-world applications, higher unrolling degrees significantly enhance performance.