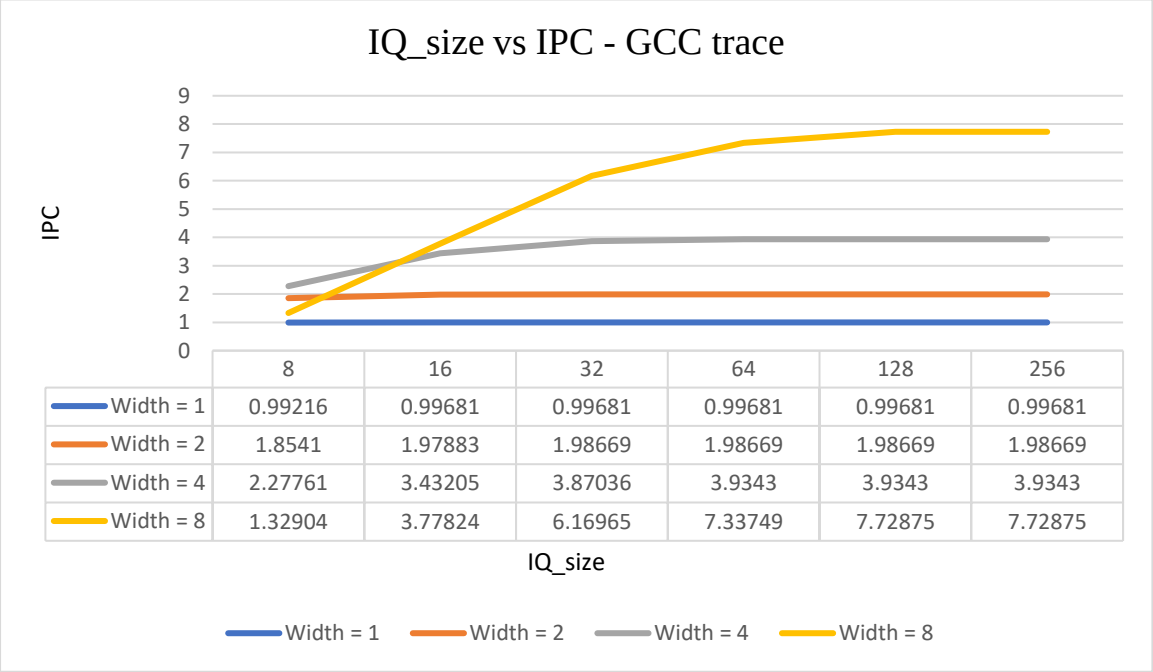
Dynamic Instruction Scheduling

Report

GRAPH 1:

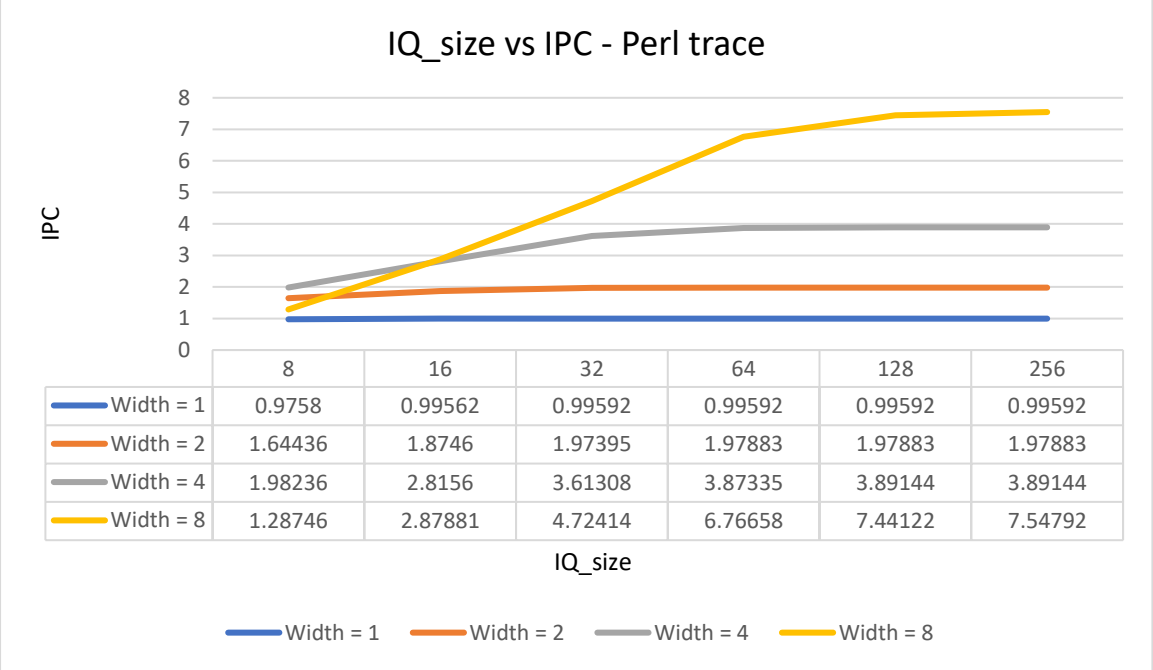
Large ROB, effect of IQ_size – IQ_size vs. IPC for various pipeline width

GCC trace:



	8	16	32	64	128	256
Width = 1	0.99216	0.99681	0.99681	0.99681	0.99681	0.99681
Width = 2	1.8541	1.97883	1.98669	1.98669	1.98669	1.98669
Width = 4	2.27761	3.43205	3.87036	3.9343	3.9343	3.9343
Width = 8	1.32904	3.77824	6.16965	7.33749	7.72875	7.72875

Perl trace:



	8	16	32	64	128	256
Width = 1	0.9758	0.99562	0.99592	0.99592	0.99592	0.99592
Width = 2	1.64436	1.8746	1.97395	1.97883	1.97883	1.97883
Width = 4	1.98236	2.8156	3.61308	3.87335	3.89144	3.89144
Width = 8	1.28746	2.87881	4.72414	6.76658	7.44122	7.54792

Graph Analysis:

The 5% values of IPC to that of largest IQ size are:

	GCC trace	Perl trace
Width = 1	0.9469695	0.946124
Width = 2	1.8873555	1.8798885
Width = 4	3.737585	3.696868
Width = 8	7.3423125	7.170524

Thus, the corresponding IQ size are:

	"Optimized IQ_size per width" Minimum IQ_size that still achieves within 5% of the IPC of the largest IQ_size	
	Benchmark 1	**Benchmark 2**
Width = 1	8	8
Width = 2	16	32
Width = 4	32	64
Width = 8	128	128

Discussion:

- From the graphs it is observed that, for increasing Issue Queue size, the Instruction count Per Cycle (IPC) increases until it reaches a certain point beyond which it remains constant. It is also found that for increasing width, the IPC achieved increases and reaches a value almost equal to the value of the pipeline width. Similar performance is observed for all values of width.

$$\text{Width} \propto \text{IPC}$$

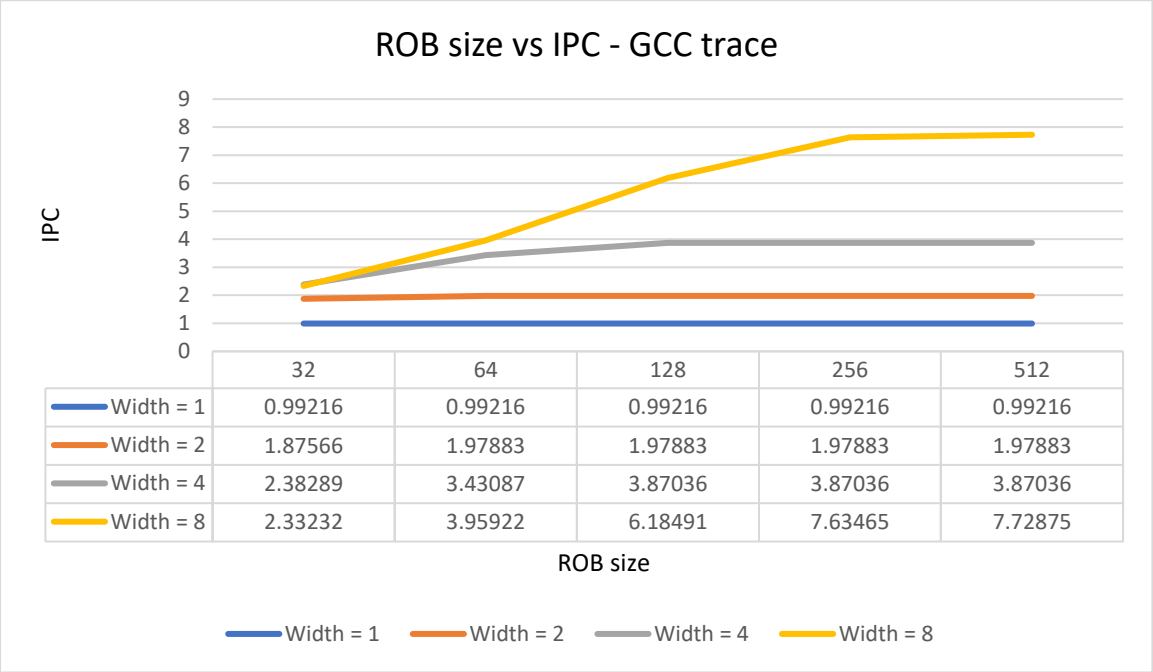
- From the graph analysis, it is found that GCC trace produces optimal IPC characteristics for lower IQ_size on average and thus performs better than Perl trace. This may be due to less stalling of instructions at the issue queue and the number of independent instructions being higher than that in Perl trace. Higher number of dependent instructions being closer to each other leads to stalling of dependent instructions and the stall time may increase in case of higher latency of execution.

GRAPH 2:

Effect of ROB_size – ROB_size vs. IPC for various pipeline width

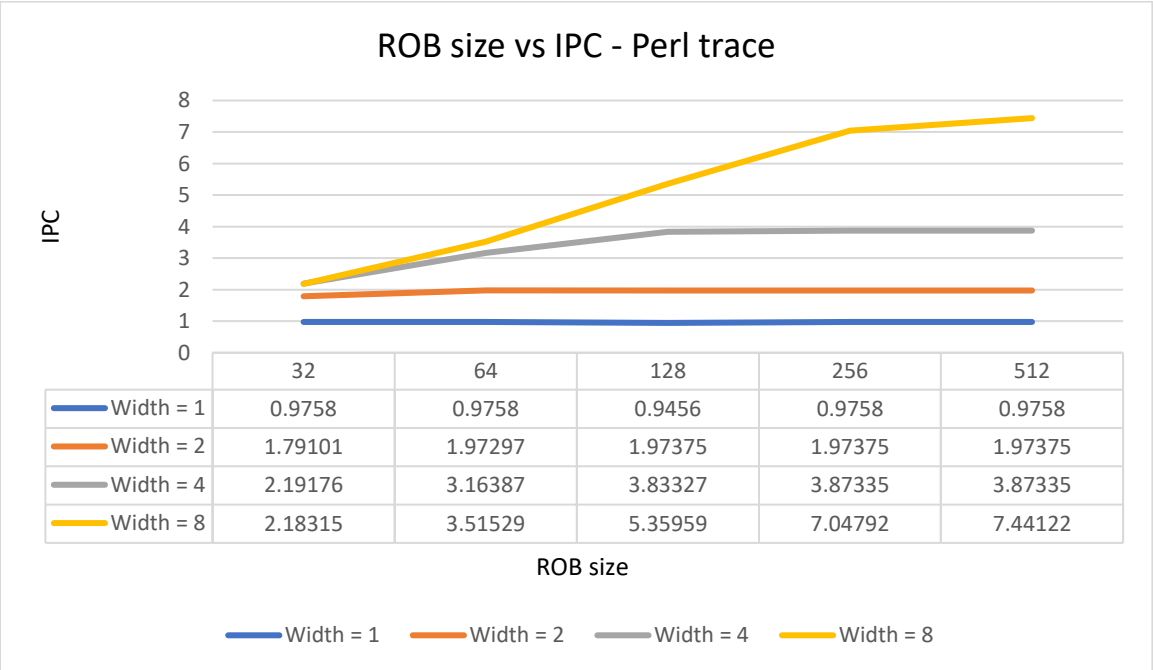
IQ size is selected from the analysis of previous graph (Optimized IQ size).

GCC trace:



	32	64	128	256	512
Width = 1	0.99216	0.99216	0.99216	0.99216	0.99216
Width = 2	1.87566	1.97883	1.97883	1.97883	1.97883
Width = 4	2.38289	3.43087	3.87036	3.87036	3.87036
Width = 8	2.33232	3.95922	6.18491	7.63465	7.72875

Perl trace:



	32	64	128	256	512
Width = 1	0.9758	0.9758	0.9456	0.9758	0.9758
Width = 2	1.79101	1.97297	1.97375	1.97375	1.97375
Width = 4	2.19176	3.16387	3.83327	3.87335	3.87335
Width = 8	2.18315	3.51529	5.35959	7.04792	7.44122

Discussion:

From the graphs, it can be observed that for increasing ROB size, IPC increases and reaches a near equal value of pipeline width. This may be due to fact as width increases, the number of instructions in a pipeline stage increases and thus a larger Re-order buffer is required accommodate incoming instructions in-order to reduce the stall time. Similar performance is observed for various width simulated.

Conclusion:

Thus the performance of the simulator is observed for varying IQ size and ROB size by measuring IPC and the graphs are analysed for optimal performance.