

# An Implementation Study of Register Renaming Logic

## Group Members:

Anupam Singh  
Harshit Goenka  
Jitendra Patel  
Prathamesh  
Sanskar Agarwal  
Shivam Yadav  
Utkarsh

## Abstract:

Most modern computer applications require that the computer system meets minimum requirements in order to run. One of those requirements is processor's performance. Computer architects are trying to improve a processor's performance in every way possible. A processor's performance can be enhanced in two ways:

- By improving the design of the processor (Hardware)
- By changing the ways in which the processor works (Software)

One of the ways of improving processor's performance is **register renaming**. A [register](#) is a small unit of fast storage quickly accessible to a computer's CPU. Every logical register has a set of physical registers associated with it. While a programmer in assembly language refers,

for instance to a logical register rax, the processor transposes this name to one specific physical register. The physical registers are opaque and cannot be referenced directly but only via the canonical names. This property of CPU registers can be exploited to improve CPU performance. If in an assembly language, multiple instructions refer to the same registers (i.e. having same canonical names), but these instructions are independent of each other, we can simply change the canonical name of some of these registers. This will make these instructions eligible for parallel execution, thus reducing number of CPU cycles per instruction. This helps to reveal more [instruction level parallelism](#) in an instruction stream. There are three types of data dependencies in CPU instructions:

- Read after Write (True data dependency)
- Write after Read (False data dependency)
- Write after Write (False data dependency)

Register renaming helps to remove these false data dependencies. This project will help us understand how this elegant technique is implemented in modern processors and how it affects the CPU performance. This project will cover some register renaming algorithms (implementation and analysis).