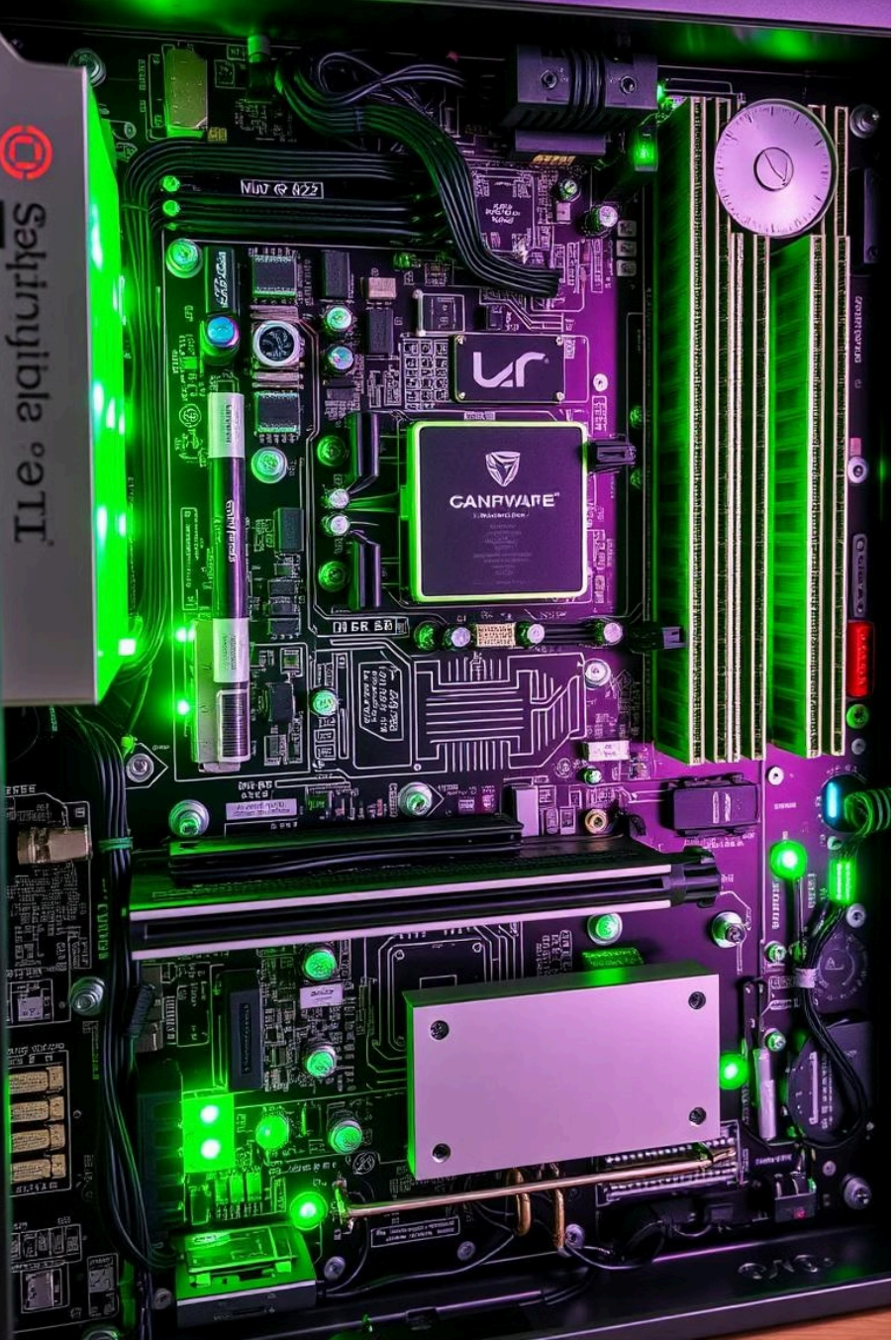


MIPS Implementations in Computer Organizations

This presentation explores the MIPS architecture, its instruction set, execution cycle, and its implementation in modern computer systems. We'll delve into pipelining techniques, hazards, and the advancements in MIPS microarchitecture.

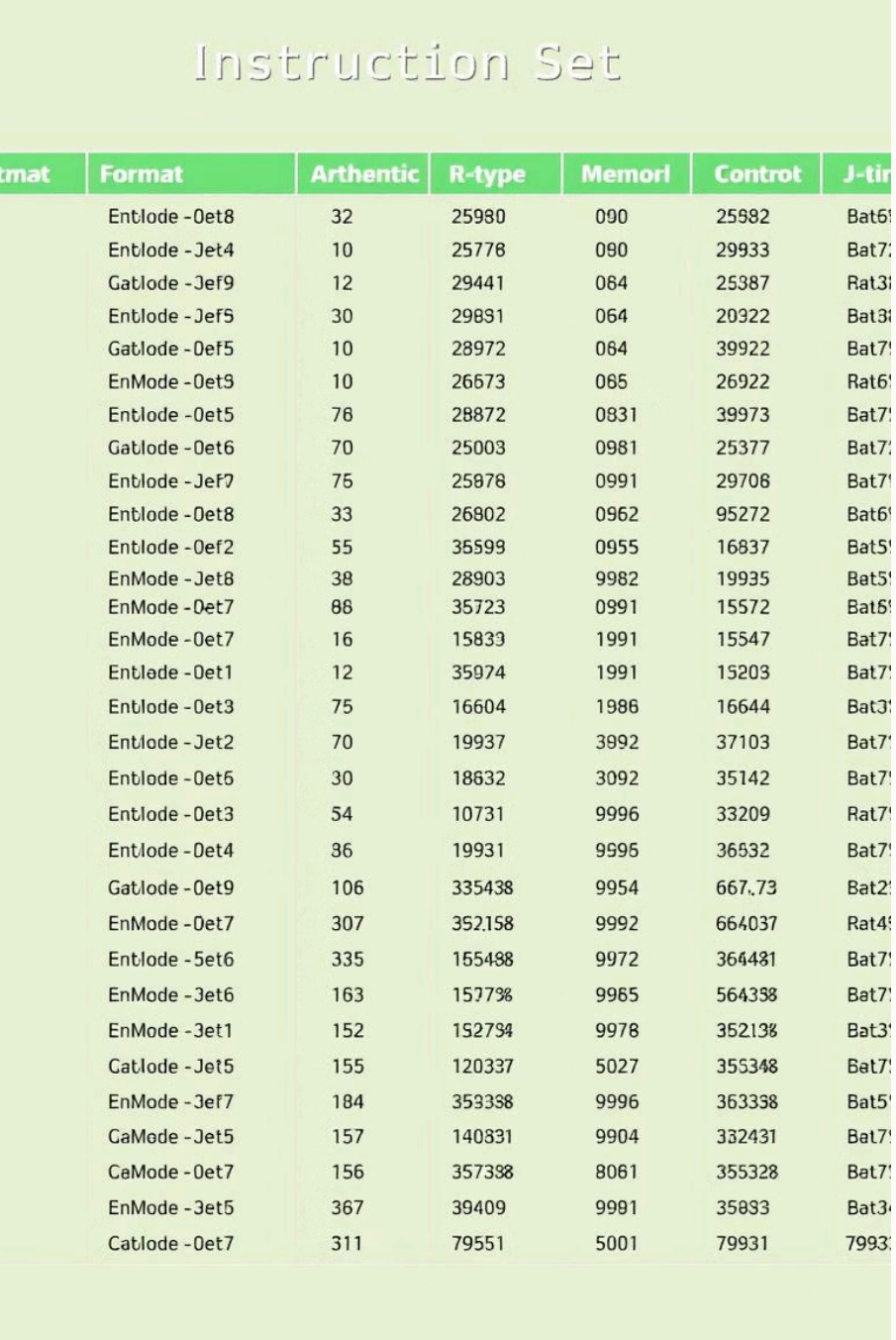
Introduction to MIPS Architecture

Reduced Instruction Set Computing (RISC)

MIPS is a RISC architecture that emphasizes simplicity and efficiency, focusing on a small set of instructions with fixed-length formats. It has been widely used in embedded systems, networking devices, and even supercomputers.

Load-Store Architecture

MIPS is a load-store architecture, where data can only be accessed through load and store instructions. This simplifies the instruction set and allows for faster execution.



MIPS Instruction Set and Addressing Modes

Instruction Types

MIPS instructions are categorized into arithmetic, logical, data transfer, control flow, and system instructions. These categories simplify the design and implementation of the processor.

Addressing Modes

MIPS supports various addressing modes, including register direct, immediate, register indirect, and base-offset addressing. These modes offer flexibility in accessing data in memory.

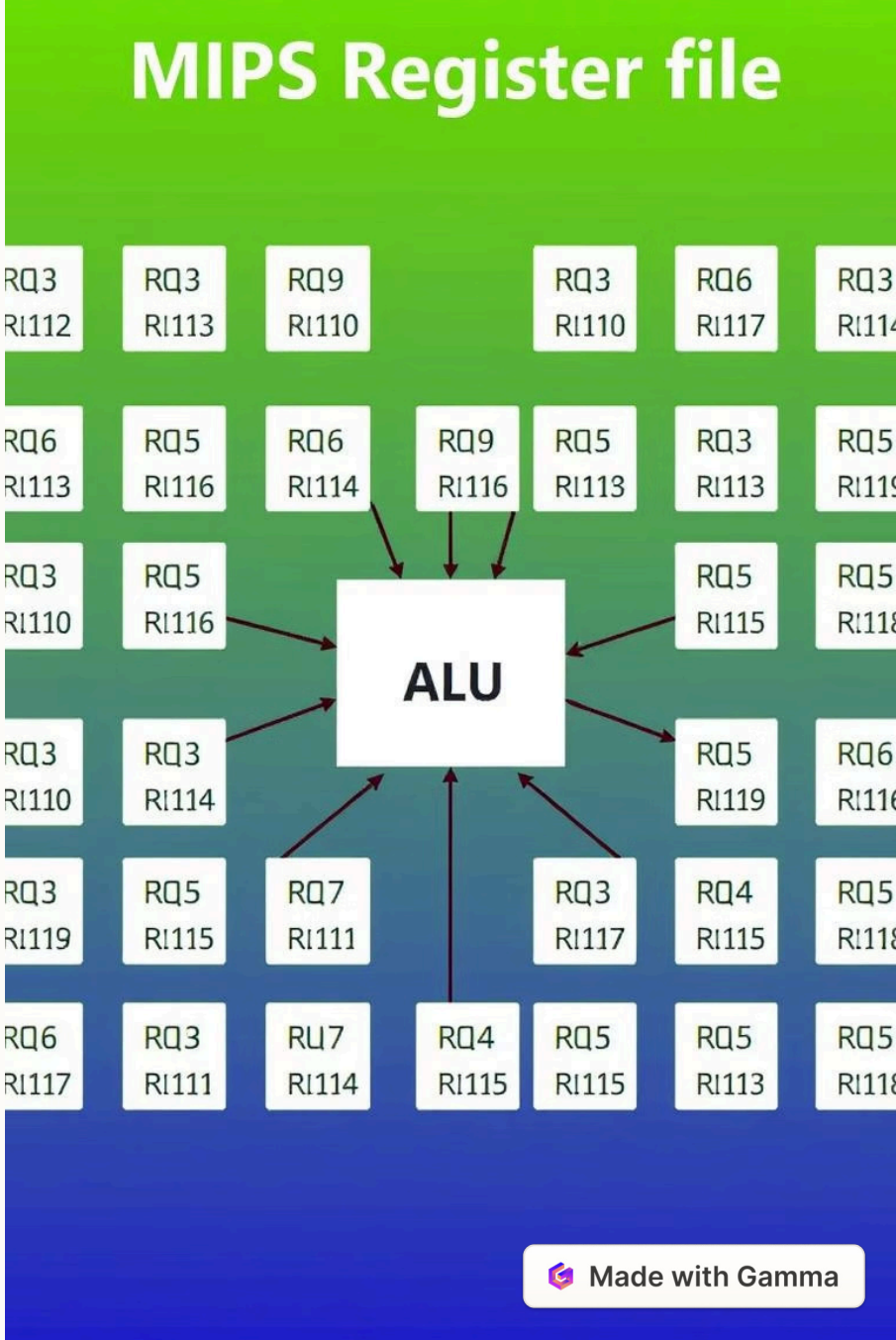
MIPS Register File and Memory Organization

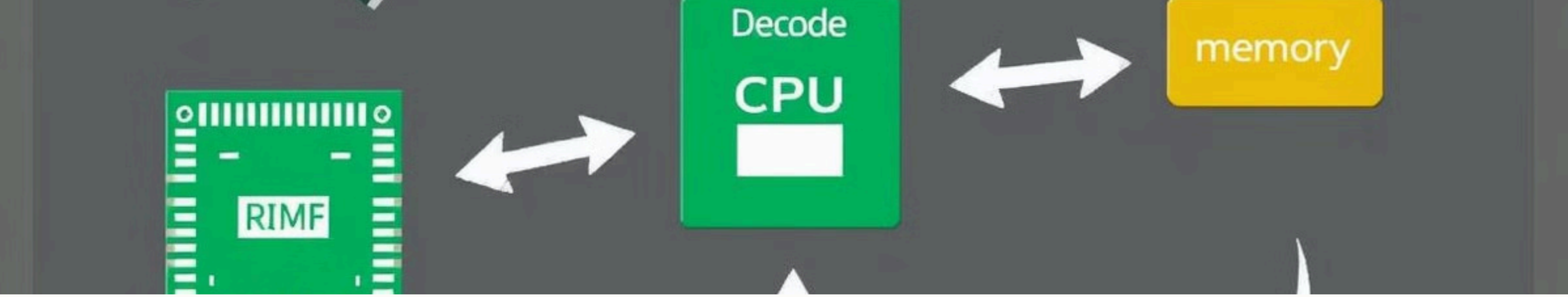
Register File

MIPS has 32 general-purpose registers, providing fast access to frequently used data. These registers are used to store temporary values and program variables.

Memory Organization

MIPS memory is organized into a linear address space, with each byte having a unique address. This structure facilitates efficient memory management and data access.





MIPS Instruction Execution Cycle

1

Fetch: The instruction is fetched from memory and placed into the instruction register.

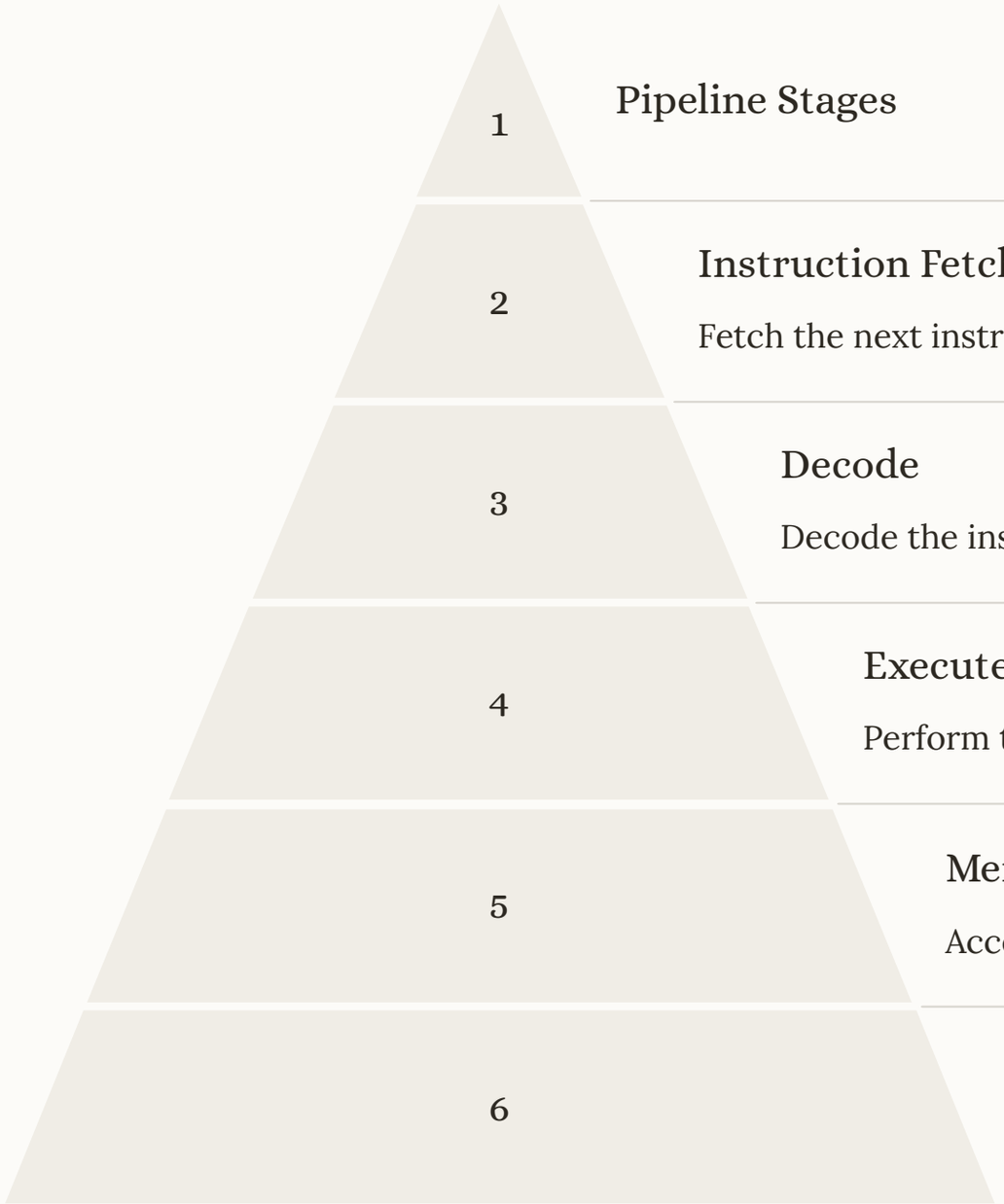
2

Decode: The instruction is decoded, identifying the operation and operands.

3

Execute: The operation is performed, with the result being stored in the register file or memory.

Pipelining in MIPS Processors



Hazards and Stalls in MIPS Pipelines

1

Data Hazards

Occur when an instruction depends on the result of a previous instruction that hasn't completed yet.

2

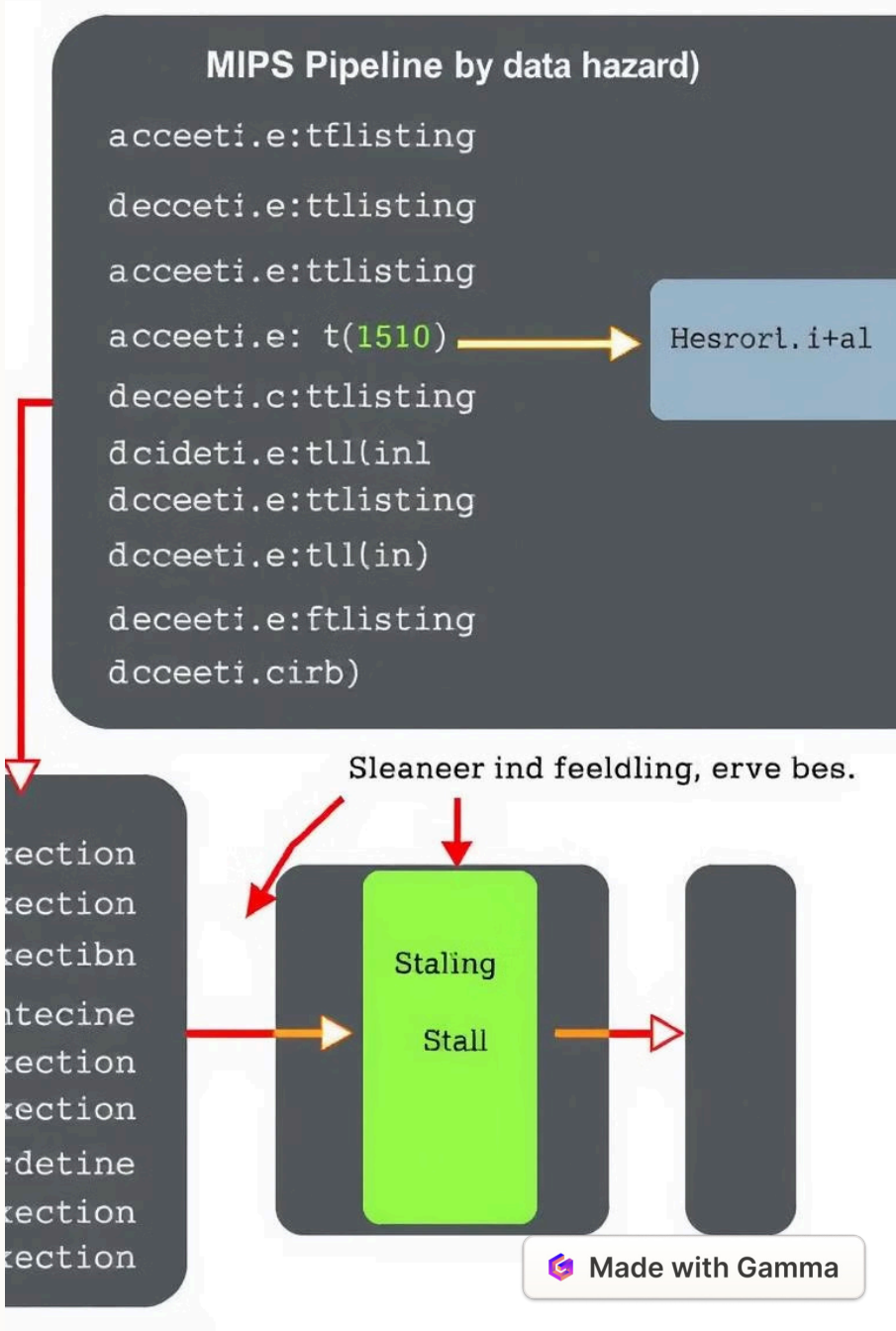
Control Hazards

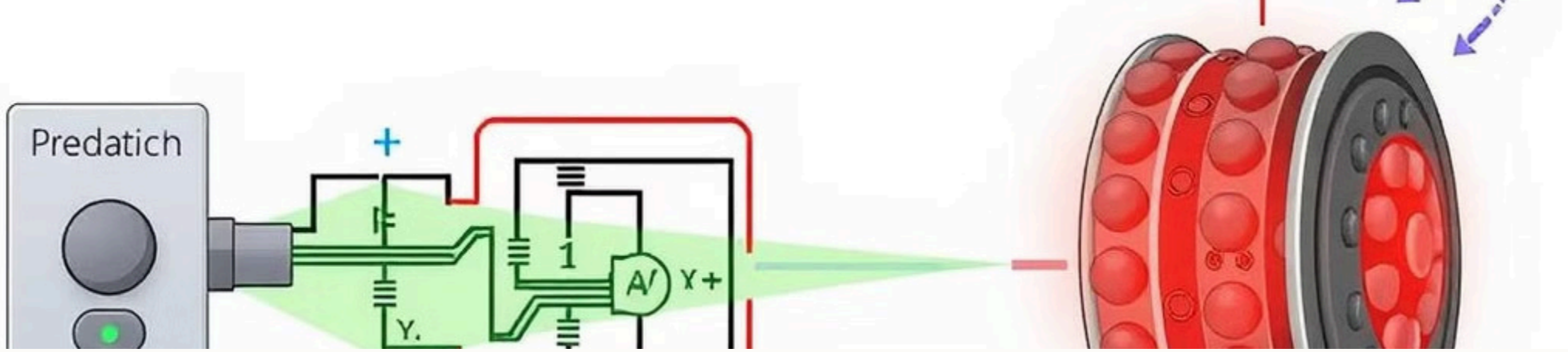
Occur when a branch instruction changes the flow of control, disrupting the pipeline's sequential execution.

3

Stalls

Pipeline stalls introduce delays to resolve hazards, ensuring the correct sequence of instructions.





Branch Prediction Techniques in MIPS

1

Static Prediction

Always predicts a branch to be taken or not taken, based on a predefined rule.

2

Dynamic Prediction

Uses the execution history of previous branches to predict the outcome of current branches.

3

Branch History Table

Stores recent branch outcomes, allowing for more accurate predictions based on historical patterns.

Exceptions and Interrupts in MIPS

1

Exceptions

Synchronous events that occur during program execution, such as an arithmetic overflow.

2

Interrupts

Asynchronous events that occur external to the CPU, such as a network packet arrival.

3

Exception Handling

A mechanism to detect, handle, and resume program execution after an exception or interrupt.

Advances in MIPS Microarchitecture