

## Introducción

MIPS32 es una arquitectura de alto desempeño estándar en la industria que provee un set de instrucciones robusto y bien definido, con soporte para una amplia gama de herramientas de desarrollo de hardware y software como compiladores, debuggers, emuladores en circuito, middleware, plataformas de aplicación y referencias de diseño.

La arquitectura MIPS se basa en un set de instrucciones que utiliza un modelo de carga y almacenamiento de datos, en el cual las operaciones se realizan sobre operandos en los registros del procesador. La memoria principal solo es accedida por las instrucciones de carga y almacenamiento.

Esta arquitectura tiene una ventaja substancial en cuanto a costo y rendimiento sobre otras implementaciones de microprocesadores que se basan en arquitecturas tradicionales.

El ISA (Instruction Set Architecture) de MIPS define una familia de instrucciones que operan con datos de 32 bits dentro de la estructura de la arquitectura MIPS. En el ISA se incluyen todas las instrucciones con y sin privilegios con las cuales el programador se comunica con el procesador.

El concepto de Endianness hace referencia al orden en que los bytes de la memoria de una computadora son leídos. La información puede ser representada de dos formas: Big-Endian y Little-Endian.

Big-Endian almacena primero el bit más significativo. Al leerse múltiples bytes, el primer byte o la dirección más baja de memoria será el más significativo. Little-Endian hace lo contrario, almacenando primero el bit menos significativo.

En el diseño de nuestro procesador y el decodificador de instrucciones se utilizará Big-Endian.

Set de instrucciones

Instrucción	Tipo	Sintaxis
<b>Add</b>	R	Add \$rd, \$rs, \$rt
<b>Sub</b>	R	Sub \$rd, \$rs, \$rt
<b>Mul</b>	R	Mul \$rd, \$rs, \$rt
<b>Div</b>	R	Div \$rs, \$rt
<b>Or</b>	R	Or \$rd, \$rs, \$rt
<b>And</b>	R	And \$rd, \$rs, \$rt
<b>Addi</b>	I	Addi \$rt, \$rs, immediate
<b>Subi</b>	I	Subi \$rt, \$rs, immediate
<b>Ori</b>	I	Ori \$rt, \$rs, immediate
<b>Andi</b>	I	And \$rt, \$rs, immediate
<b>Lw</b>	I	Lw \$rt, offset
<b>li</b>	I	li \$rd, value
<b>la</b>	I	la \$rd, exp
<b>Sw</b>	I	Sw \$rt, offset
<b>slt</b>	R	slt \$rd, \$rs, \$rt
<b>Slti</b>	I	Slti \$rt, \$rs, immediate
<b>beq</b>	I	beq \$rs, rt, offset
<b>bne</b>	I	bne \$rs, rt, offset
<b>j</b>	J	J target
<b>jal</b>	J	jal target
<b>jr</b>	J	jr \$rs
<b>nop</b>	R	nop
<b>bgtz</b>	I	bgtz \$rs, offset

### Ljubisa Bajic:

Ljubisa Bajic es un veterano de la industria de semiconductores, trabajando exhaustivamente en diseño y debug de VLSI, además de tener amplia experiencia en software y arquitectura de aceleración.

Ljubisa se desempeñó 10 años dentro de AMD como arquitecto de ASIC trabajando en manejo de energía y diseño DSP, antes de entrar a NVIDIA como arquitecto senior, para después regresar a AMD por algunos años.

Dejó AMD para iniciar Tenstorrent en 2016, una compañía que cofundó para traer al mercado soluciones en cómputo de IA con un nuevo acercamiento que responde al crecimiento exponencial de la complejidad de los modelos de IA.

### Jim Keller:

James B. Keller es un ingeniero de microprocesadores americano, con un trabajo pionero en la arquitectura de computadoras que abarca tres décadas. Su carrera en el desarrollo de CPUs inició en 1984, siendo parte del equipo que diseñó el chip Alpha de DEC, lanzado en 1992 y que corría a 500 MHz con una memoria caché de 1 GHz.

En 1998 se une a AMD con la idea de integrar el procesador con unidades separadas como la memoria y la transferencia de datos, ideas que fueron implementadas en los chips K8.

Keller también fue pionero en la idea de los diseños de doble núcleo. Además, trabajó en chips basados en PowerPC y en ARM para empresas como Apple y Samsung.

De vuelta en AMD en 2012, Keller sentó las bases para el diseño de la microarquitectura Zen. En 2015 se une a Tesla para el diseño de chips que serían implementados en los automóviles de la compañía. En 2018 se une a Intel como vicepresidente, sin embargo, abandonándolo en junio de 2020.

## Raja Koduri

Raja Koduri es un ingeniero y ejecutivo en hardware de gráficos de computadora. Koduri ha dirigido los esfuerzos en gráficos de AMD como vicepresidente y arquitecto en jefe para Radeon Technologies.

Antes de eso, se desempeñó como director en arquitectura de gráficos para Apple, donde ayudó a crear el subsistema de gráficos para las computadoras Macintosh de la compañía. Ha ocupado diversos puestos como líder en el sector de gráficos para computadora en AMD.

Koduri se unió a Intel en 2017 como arquitecto en jefe, donde se concentró en continuar el crecimiento de los esfuerzos de la compañía en materia de gráficos.

## **Objetivo**

### Objetivo general

- Implementar el single datapath de un microprocesador con arquitectura MIPS de 32 bits.

### Objetivos particulares

- Ser capaz de ejecutar instrucciones de tipo R.
- Definir el set de instrucciones que se ejecutarán en el diseño del procesador.

## **Desarrollo**

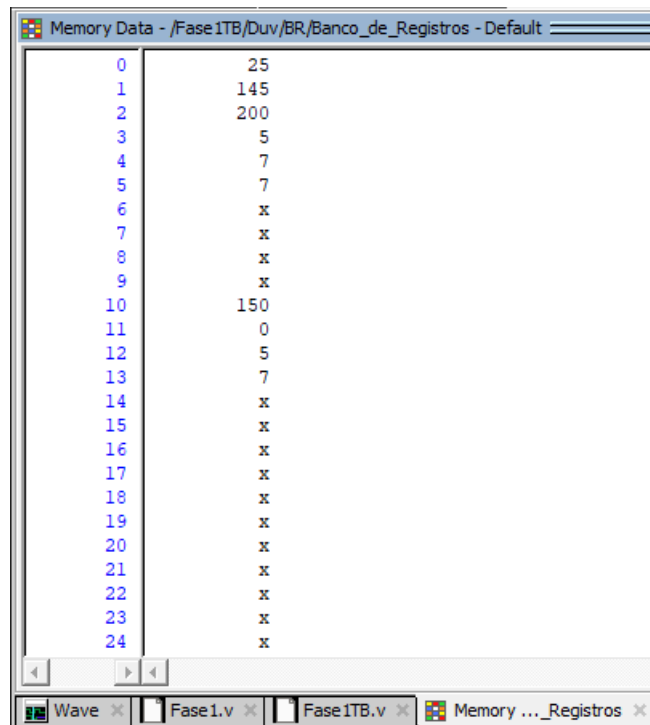
En esta fase se implementó el single datapath de nuestro procesador MIPS de 32 bits. Este cuenta con los siguientes módulos:

- PC o Program Counter, el cual consta de un registro que se encarga de llevar seguimiento de la secuencia del programa.
- La memoria de instrucciones, la que almacena la instrucción actual.
- El adder, que incrementa el PC una cantidad predeterminada cada ciclo de reloj.
- La unidad de control, que envía señales a las otras unidades del datapath para indicarles la respuesta adecuada a cada instrucción.

- El banco de registros, el cual contiene los registros para almacenar información. Tiene puertos de lectura y escritura.
- La ALU, que ejecuta una operación aritmeticológica sobre dos operandos.

## Conclusiones

Con la fase 1 implementada, ahora debería de poder ejecutar instrucciones de tipo R. Para evaluar el funcionamiento del sistema se precargaron instrucciones en código binario a la memoria de instrucciones.



Register	Value
0	25
1	145
2	200
3	5
4	7
5	7
6	x
7	x
8	x
9	x
10	150
11	0
12	5
13	7
14	x
15	x
16	x
17	x
18	x
19	x
20	x
21	x
22	x
23	x
24	x

## Referencias

- MIPS (2014). MIPS Architecture For Programmers Volume I-A: Introduction to the MIPS32 Architecture. Obtenido de <https://s3-eu-west-1.amazonaws.com/downloads-mips/documents/MD00082-2B-MIPS32INT-AFP-06.01.pdf>
- MIPS (2016). MIPS Architecture For Programmers Volume II-A: The MIPS32 Instruction Set Manual. Obtenido de <https://s3-eu-west-1.amazonaws.com/downloads-mips/documents/MD00082-2B-MIPS32INT-AFP-06.01.pdf>

[1.amazonaws.com/downloads-mips/documents/MD00086-2B-MIPS32BIS-AFP-6.06.pdf](https://1.amazonaws.com/downloads-mips/documents/MD00086-2B-MIPS32BIS-AFP-6.06.pdf)

- Ahmad, M. (2021). The story of Jim Keller and his pioneering work on chip design and architecture. Obtenido de <https://www.edn.com/the-story-of-jim-keller-and-his-pioneering-work-on-chip-design-and-architecture/>
- Cutress, I. (2021). An Interview with Tenstorrent: CEO Ljubisa Bajic and CTO Jim Keller. Obtenido de <https://www.anandtech.com/show/16709/an-interview-with-tenstorrent-ceo-ljubisa-bajic-and-cto-jim-keller>
- Martin, D. (2023). Raja Koduri, Main Driver Of Intel's GPU Efforts, To Leave Company. Obtenido de <https://www.crn.com/news/components-peripherals/raja-koduri-main-driver-of-intel-s-gpu-efforts-to-leave-company>
- Parr, K. (2021). What is Endianness? Big-Endian vs Little-Endian Explained with Examples. Obtenido de <https://www.freecodecamp.org/news/what-is-endianness-big-endian-vs-little-endian/>