

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Milme Familia	1/10	microcontrolador	20/01/2025

Title: RISC-VISA

Keyword	Topic: Introducción al conjunto de instrucciones
microprocesador Hardware Procesador puerto VHDL	Notes: RISC-V empezó en el 2010 y actualmente hoy muchas universidades y empresas implementan. Una de sus ventajas es su diseño modular el objetivo es ayudar a desarrollar en microprocesador más sencillo y el enfoque es mayor a Hardware que Software. for diseño se puede realizar en lenguaje de hardware como VHDL la implementación final se realiza a través de sintetizadores que convierten el diseño en puertas lógicas. Los niveles de abstracción ayudan a reducir la complejidad del diseño e ingeniería y tiempos de la computación.
Questions	

Summary: Introduce al RISC-V como un conjunto de instrucciones abierto modular y eficiencia ideal propósito educativo e industrial destacando su simplicidad que facilita su desarrollo.

NAME

milne familia

PAGES

2/10

SPEAKER/CLASS

microcontrolador

DATE - TIME

21/01/2025

Title: RISC-VISA

Keyword

Ensamblador
Instrucciones
Tipo R

ISA

Topic: Instrucciones Tipo R

Notes: Desplaza a la izquierda el registro
RS1 un numero RS2 de bits y lo guarda en
RD

Desplazamiento a la izquierda.

 $Sll Rd, RS1, RS2 // Rd \leftarrow RS1 \ll RS2$

X20	00000000 ₂	
X21		
X22	00000111 ₂	
X23	00000110 ₂	

Questions

Cual es la dife-
rencia entre SLT
y SLTU?Que ventajas
tienen las
instrucciones
Tipo R sobre
otro tipos en
RISC-V?

- Permiten optimizar Calculos en Hardware, como multiplicaciones y divisiones rapidas.
- Son esenciales en algoritmos de busqueda encriptación y manipulación de datos.
- Se utilizan en estructuras de control para tomar decisiones en el código ensamblador.

Summary:

Hablamos de la instrucciones de tipo R en RISC-V enfocados en operaciones de composición y desplazamientos. Esta instrucción es fundamental para manipular datos, Calculos eficiente dentro del procesador etc.

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Milma familia	3/10	microcontrolador	21/01/2025

Title: RISC-VISA

Keyword	Topic: instrucciones tipo I																														
ADDI SLTI XORI ANDI ORI SLTI SLLI SRAI.	<p>Notes: Estas instrucciones usan un registro fuente (RS1), un registro destino (RD) y una constante inmediata (imm). La constante inmediata es de 12 bits y se representa en complemento a dos.</p> <p>Formato Tipo I:</p> <table border="1"> <tr> <td>31-20 (12 bits)</td> <td>19-15 (5 bits)</td> <td>14-12 (3 bits)</td> </tr> <tr> <td>inmediato (imm)</td> <td>RS1</td> <td>funcion</td> </tr> <tr> <td>11-7 (5 bits)</td> <td>6-0 (7 bits)</td> <td></td> </tr> <tr> <td>RD</td> <td>opcode</td> <td></td> </tr> </table> <p>Questions</p> <p>¿Qué hace la instrucción ADDX5, x6, -4 en un programa RISC-V?</p> <table border="1"> <tr> <td>ADDI</td> <td>Suma inmediata</td> <td>$rd = rs1 + imm$</td> </tr> <tr> <td>SLTI</td> <td>Set less than imm</td> <td>$rd = (rs1 < imm) ? 1 : 0$</td> </tr> <tr> <td>SLTIU</td> <td>Set less than imm (U)</td> <td>$rd = (rs1 < imm) ? 1 : 0$ unsigned</td> </tr> <tr> <td>XORI</td> <td>XOR con inmediata</td> <td>$rd = rs1 \wedge imm$</td> </tr> <tr> <td>ORI</td> <td>OR con inmediata</td> <td>$rd = rs1 \vee imm$</td> </tr> <tr> <td>ANDI</td> <td>AND con inmediata</td> <td>$rd = rs1 \& imm$</td> </tr> </table>	31-20 (12 bits)	19-15 (5 bits)	14-12 (3 bits)	inmediato (imm)	RS1	funcion	11-7 (5 bits)	6-0 (7 bits)		RD	opcode		ADDI	Suma inmediata	$rd = rs1 + imm$	SLTI	Set less than imm	$rd = (rs1 < imm) ? 1 : 0$	SLTIU	Set less than imm (U)	$rd = (rs1 < imm) ? 1 : 0$ unsigned	XORI	XOR con inmediata	$rd = rs1 \wedge imm$	ORI	OR con inmediata	$rd = rs1 \vee imm$	ANDI	AND con inmediata	$rd = rs1 \& imm$
31-20 (12 bits)	19-15 (5 bits)	14-12 (3 bits)																													
inmediato (imm)	RS1	funcion																													
11-7 (5 bits)	6-0 (7 bits)																														
RD	opcode																														
ADDI	Suma inmediata	$rd = rs1 + imm$																													
SLTI	Set less than imm	$rd = (rs1 < imm) ? 1 : 0$																													
SLTIU	Set less than imm (U)	$rd = (rs1 < imm) ? 1 : 0$ unsigned																													
XORI	XOR con inmediata	$rd = rs1 \wedge imm$																													
ORI	OR con inmediata	$rd = rs1 \vee imm$																													
ANDI	AND con inmediata	$rd = rs1 \& imm$																													

Summary: Las instrucciones Tipo I en RISC-V permiten operaciones entre un registro y una constante inmediata de 12 bits. Incluyen suma (ADDI), comparación (SLTI, SLTIU), operaciones lógicas (XORI, ANDI, ORI) y desplazamientos (SLLI, SLLI, SRAI).

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Milne Jomila	4/10	microcontrolador	21/01/2025

Title: RISC-V ISA

Keyword	Topic: instrucciones de carga y almacenamiento.
Load store instruccion I instruccion R LW (load word) SW (store word)	Notes: el conjunto de instrucciones RISC-V incluye operaciones específicas para mover datos entre registros y memoria, conocidas como instrucciones de carga y almacenamiento. RISC-V sigue una arquitectura load-store, en la que sólo se accede a memoria mediante estas instrucciones, sin realizar operaciones directamente sobre los datos almacenados. Las instrucciones de carga (tipo I) incluyen LW (load word), LH/LHU (load halfword con/sin signo), y LB/LBU (load byte con/sin signo). Las de almacenamiento (tipo S) incluyen SW (store word), SH (store halfword) y SB (store byte).
Questions ¿Qué diferencia hay entre LB y LBU?	<ol style="list-style-type: none"> Diagrama de LW: dirección = Base + offset (datos en memoria se guardan en registro) Diagrama de SW: dirección = Base + offset (datos del registro se guardan en memoria) Comparativa de instrucciones LB vs LBU, mostrando extensión de signo.

Summary: Las instrucciones de carga y almacenamiento en RISC-V son esenciales para mover datos entre registros y memoria. Su diseño claro y eficiente permite un control preciso del acceso a memoria. Además, el soporte para tamaño de variables y accesos desalineados los hace versátiles.

NAME

Wilma familia

PAGES

5/10

SPEAKER/CLASS

microcontroladores

DATE - TIME

21/01/2025

Title: RISC-V ISA

Keyword

Topic: instrucciones de control y programa almacenado en memoria.

Notes: e

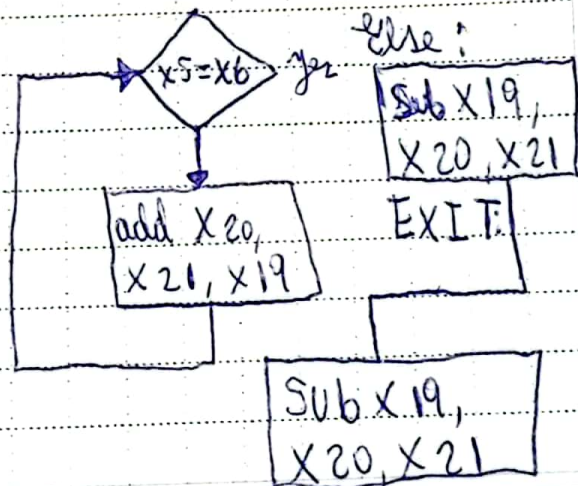
Portabilidad
Salto condicional
flujo condicional

Estas instrucciones permiten a los computadores con el propósito general, ejecutando secuencialmente programas cargados sin modificar el hardware. Se introducen instrucciones de salto condicional (branch), como beq (salto si son iguales) y bne (salto si no son iguales), que alteran el flujo del programa dependiendo de ciertas condiciones. Además, el Contador de programa se utiliza para acceder y ejecutar instrucciones almacenadas en la memoria de manera eficiente.

Questions

¿De qué manera el contador de programa optimiza la ejecución secuencial de instrucciones en los sistemas basados en RISC-V?

```
beq x5, x6, Else
add x20, x21, x19
EXIT
```



Summary:

Las instrucciones de control en RISC-V permiten la ejecución eficiente de programas, facilitando el flujo condicional y la programación secuencial, optimizando la flexibilidad y portabilidad del hardware.

By Carlos Pichardo Vinque

NAME

Milma familia

PAGES

6/10

SPEAKER/CLASS

microcontroladores

DATE - TIME

21/01/2025

Title: RISC-V ISA

Keyword

ALU

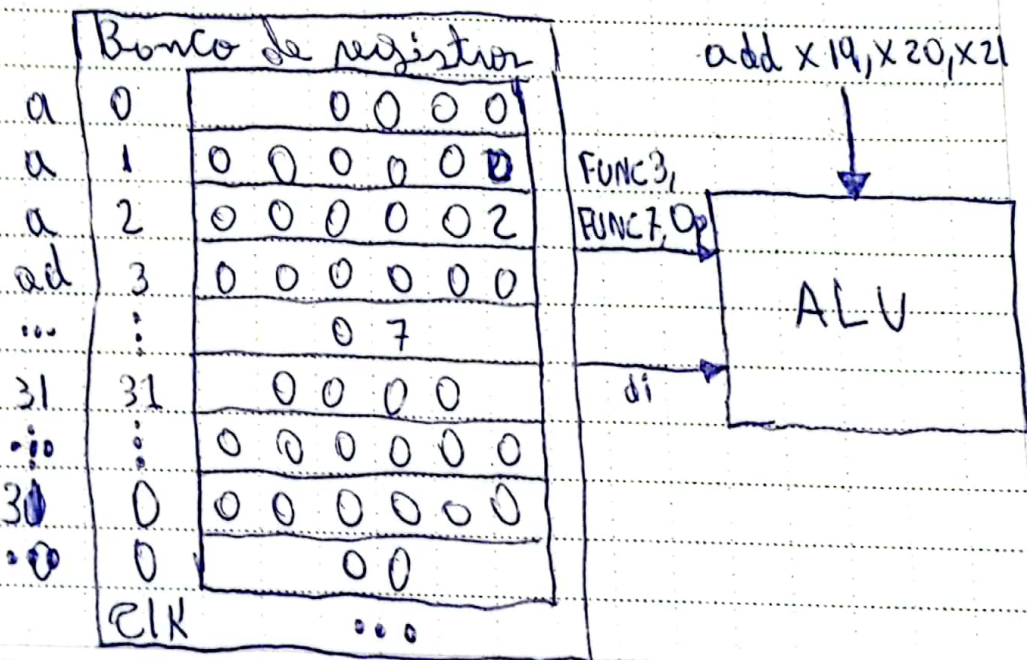
Registro

banco de
registrooperaciones
aritméticas

Topic: Operaciones entre registros

Notes: Una introducción al diseño de un procesador RISC-V simplificado, enfocándose en las operaciones entre registros. Se cubren los 10 tipos de instrucciones tipo R del RISC-V, como suma, resta y operaciones a nivel de bits, explicando cómo se implementan en hardware mediante un banco de registros y cómo se sincronizan las lecturas y escrituras con el reloj.

Questions



Summary: El diseño de un procesador RISC-V simplificado destaca la importancia de las operaciones entre registros, el banco de registros y la ALU, fundamentales para su funcionamiento eficiente.

NAME

Wilma familia

PAGES

7/10

SPEAKER/CLASS

Microcontrolador

DATE - TIME

21/01/2025

Title: RISC-V ISA

Keyword

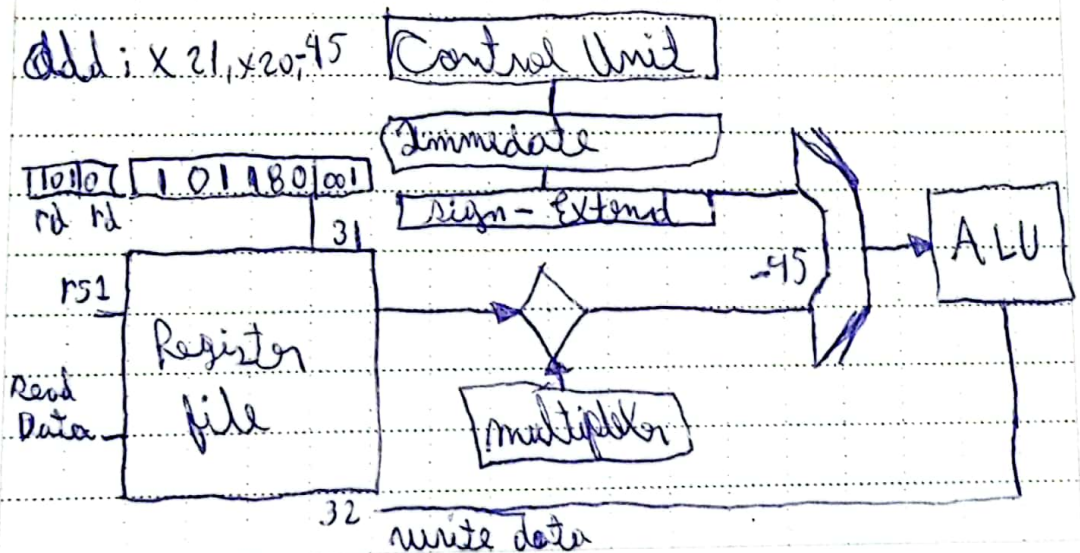
Complemento a 2
 Extension de signo
 multiplexor
 operaciones
 constantes

Topic: Operaciones con Constantes

Notes:

Las instrucciones tipo I incluyen un registro fuente y un inmediato de 12 bits codificado en Complemento a 2. Se detalla cómo se extrae el inmediato, se extiende su signo, y se realiza la operación. Como una suma, entre el registro fuente y el inmediato. El resultado se guarda en el registro destino. Además, se utiliza un multiplexor para seleccionar entre el inmediato y el registro fuente. La unidad de control gestiona la operación.

Questions



Summary: Las operaciones con constantes utilizan un inmediato de 12 bits, extendido, y gestionado por la unidad de control y multiplexores para eficiencia.

By Carlos Pickardo Uinque

NAME	PAGES	SPEAKER/CLASS	DATE - TIME
Milma familia	8/10	microcontrolador	21/01/2025

Title: RISC-V ISA

Keyword

Registro fuente
direccionamiento
procesador
monociclo.

Topic: Instrucción de carga desde memoria

Notes: La operación "load word" consiste en sumar el valor de un registro fuente (RS1) con un inmediato, lo que da como resultado una dirección de memoria. La memoria se de ~~occeso~~ instantáneo, y el dato de esa dirección se carga en el registro de destino.

```
int Load_Word (int* base_addr, int offset) {
    return * (base_addr + offset);
    // carga la palabra de memoria
```

Questions

¿Cómo se direcciona la lectura de memoria en un procesador monociclo?

La memoria tiene acceso instantáneo

```
int main() {
    int memory[10] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9};
    int base_addr = 3;
    int offset = 4;
    int result = load_word (memory, base_addr + offset);
    printf("Valor cargado: %d\n", result);
    return 0;
}
```

Summary: La instrucción de carga permite acceder a la memoria, sumar un inmediato al registro base y cargar el valor resultante en el registro destino.

NAME

Wilma familia

PAGES

9/10

SPEAKER/CLASS

microcontrolador

DATE - TIME

21/01/2025

Title:

RISC-V ISA

Keyword

Ciclo de reloj

Señales de control

Unidad control

monociclo.

Topic: memoria de programa y contador de programa

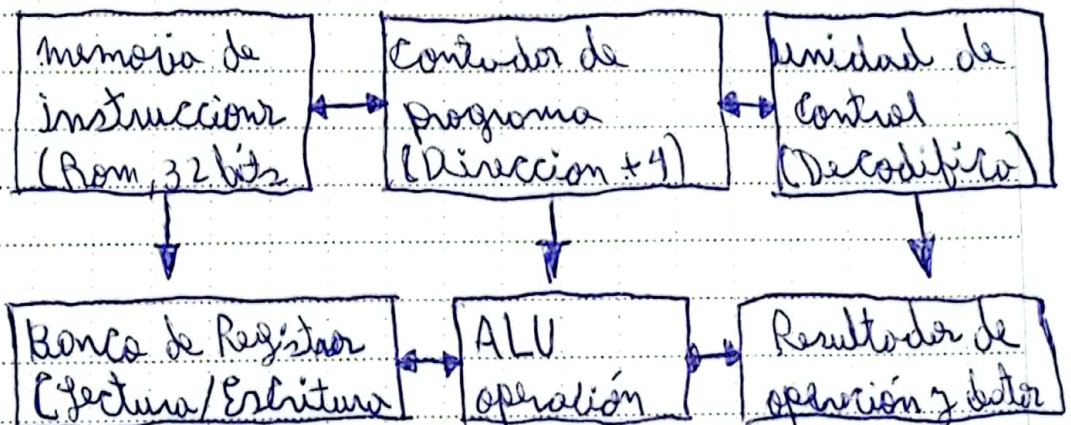
Notes:

El diseño de un procesador RISC-V monociclo, se exploran los conceptos de memoria de programa, contador de programa y ejecución secuencial de instrucciones. El procesador toma instrucciones de memoria, las decodifica y ejecuta utilizando registros y la unidad de control. Cada instrucción se encuentra en formato binario de 32 bits y se accede en bloques de 4 bytes.

Questions

¿Que es el "Signo extendido" y como se aplica en las instrucciones?

Se utiliza para ampliar un valor inmediato.



Summary:

El procesador RISC-V monociclo permite la ejecución secuencial eficiente de instrucciones, utilizando memoria de programa, contador de programa y registros, facilitando la ejecución.

NAME: Mihne familia PAGES: 10/10 SPEAKER/CLASS: microcontrolada DATE - TIME: 21/01/2025

Title: RISC-V ISA

Keyword

Topic: Instrucciones de Control (branch)

puerto AND
salto condicio-
nal.

operaciones
lógicas.

Questions

Notes: Estas instrucciones permiten cambiar la dirección de ejecución dependiendo de una comparación entre dos registros, como bne (branch if not equal) o blt (branch if less than). Para realizar esta comparación, se utilizan operaciones lógicas dentro de la ALU (Unidad Aritmética Lógica), como set less than (set LT) o set less than unsigned (set LT unsigned). El proceso implica seleccionar los registros fuente, obtener el valor inmediato (si es necesario) y realizar la comparación a través de la ALU. El resultado de la comparación, normalmente el bit menos significativo, se utiliza para decidir si se debe realizar el salto o continuar con la siguiente instrucción secuencial. Si la condición es verdadera, se ejecuta el contador del programa (PC) con el valor de desplazamiento a la siguiente instrucción.

Summary: Las instrucciones de salto condicional permiten un control eficiente del flujo del programa. Utilizando operaciones lógicas en la ALU y mecanismos como multiplexores y puertos AND, se asegura la correcta ejecución de instrucciones sin alterar el contador de programa.