

Departamento de
Arquitectura y
Tecnología de Computadores
UNIVERSIDAD DE SEVILLA

SIMULACIÓN DE UN ALGORITMO DE PLANIFICACIÓN DINÁMICA SIN RENOMBRAMIENTO DINÁMICO DE ETIQUETAS

Autora: Belén López
Salamanca
Dirigida por: Fernando Díaz
del Río

ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

Teoría de la Cronoplanificación

Arquitectura propuesta para la Cronoplanificación

Conclusiones y trabajo futuro



ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

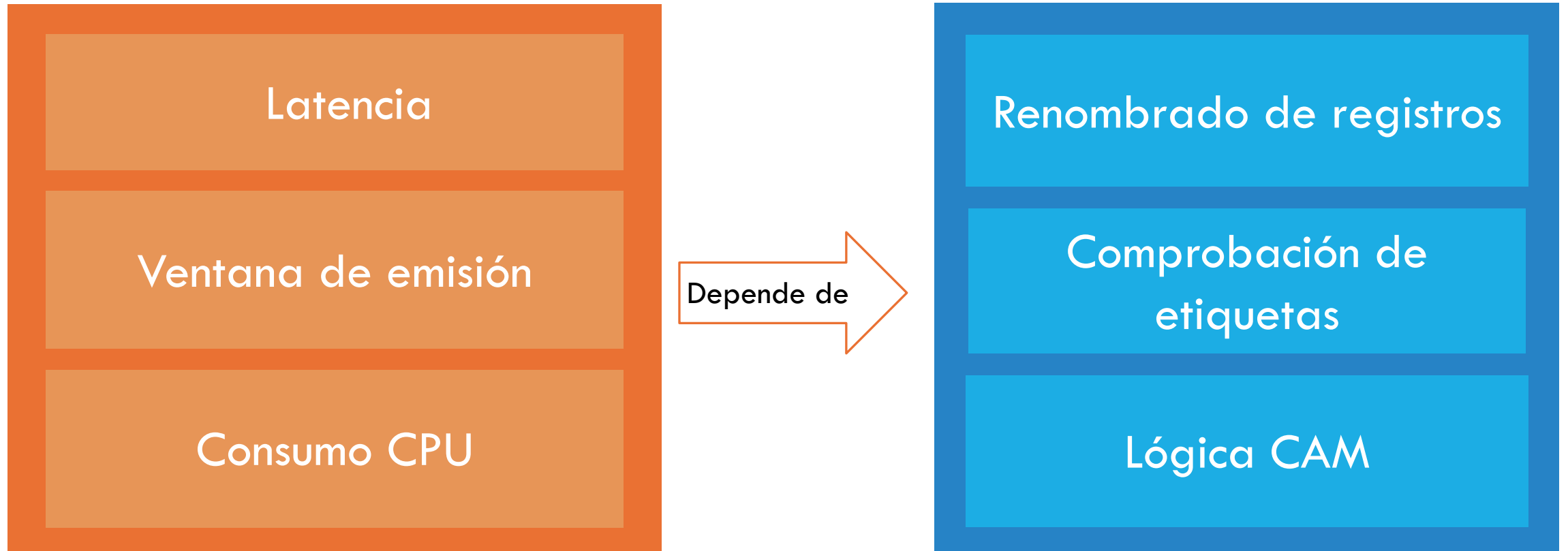
Teoría de la Cronoplanificación

Arquitectura propuesta para la Cronoplanificación

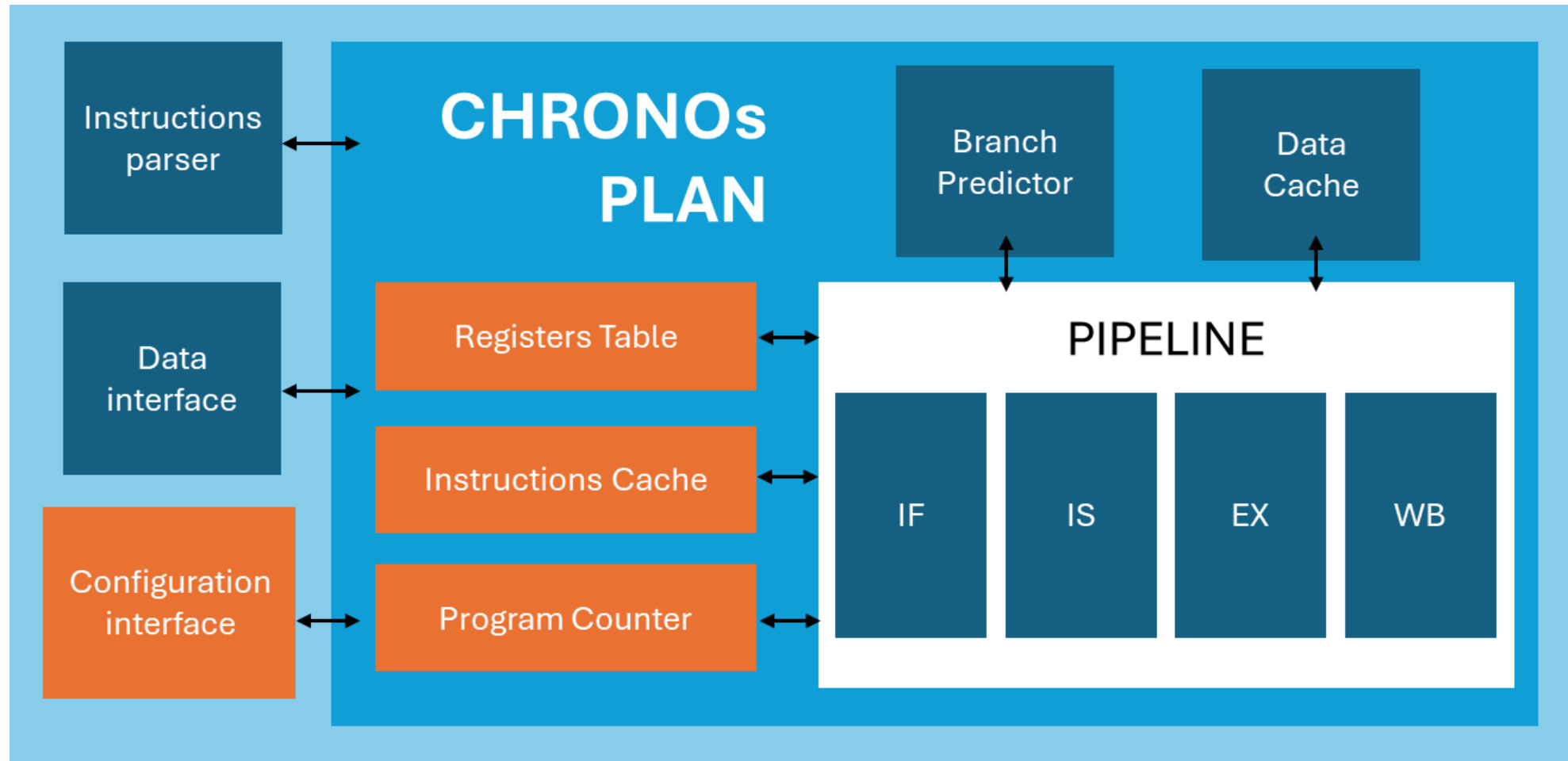
Conclusiones y trabajo futuro



MOTIVACIÓN



OBJETIVO



ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

Teoría de la Cronoplanificación

Arquitectura propuesta para la cronoplanificación

Bloqueos comparados con Tomasulo

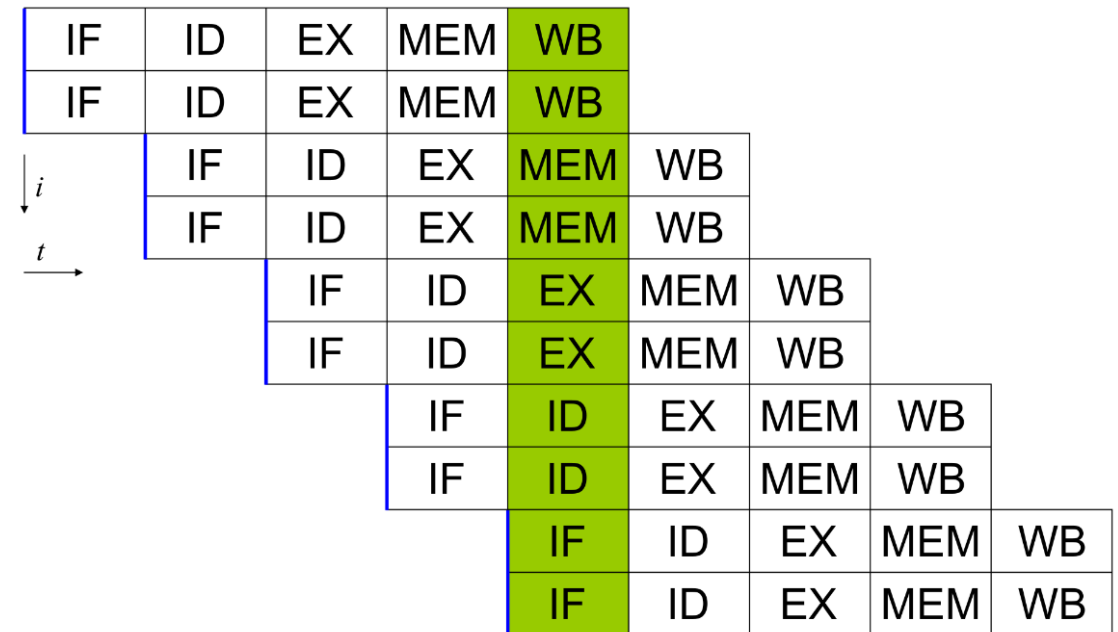
Conclusiones y trabajo futuro



PARALELIZACIÓN A NIVEL DE INSTRUCCIÓN

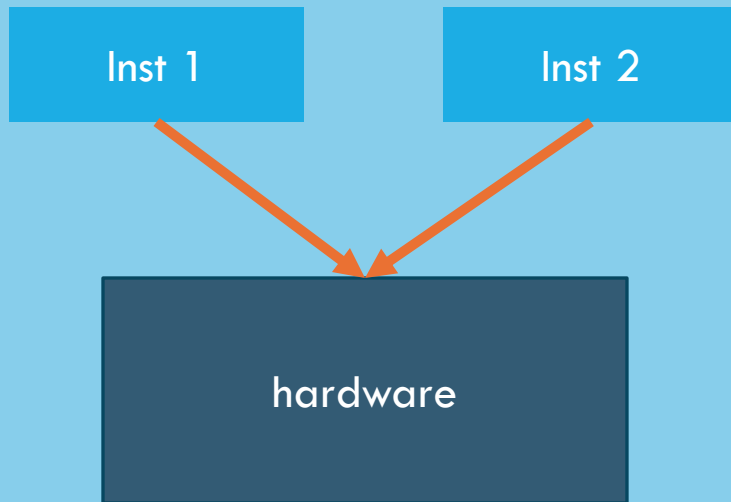
DEFINICIÓN

Habilidad de un procesador ejecutar múltiples instrucciones simultáneamente

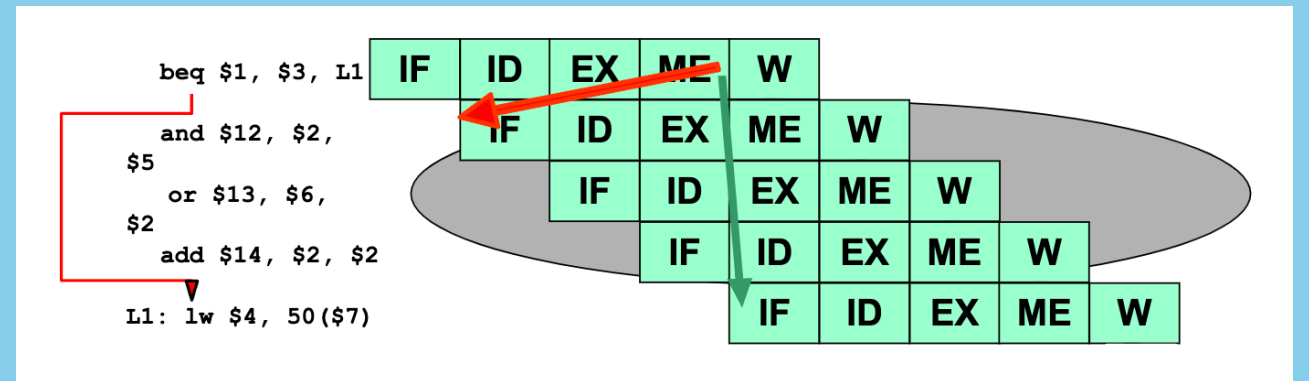


DEPENDENCIAS DE INSTRUCCIONES

Dependencias estructurales



Dependencias de control



DEPENDENCIAS DE INSTRUCCIÓN

Dependencias de datos

RAW:

lectura después de escritura

```
i1. R2 ← R5 + R8  
i2. R4 ← R2 + R8
```

WAR:

escritura después de lectura

```
i1. R4 ← R1 + R5  
i2. R5 ← R1 + R2
```

WAW:

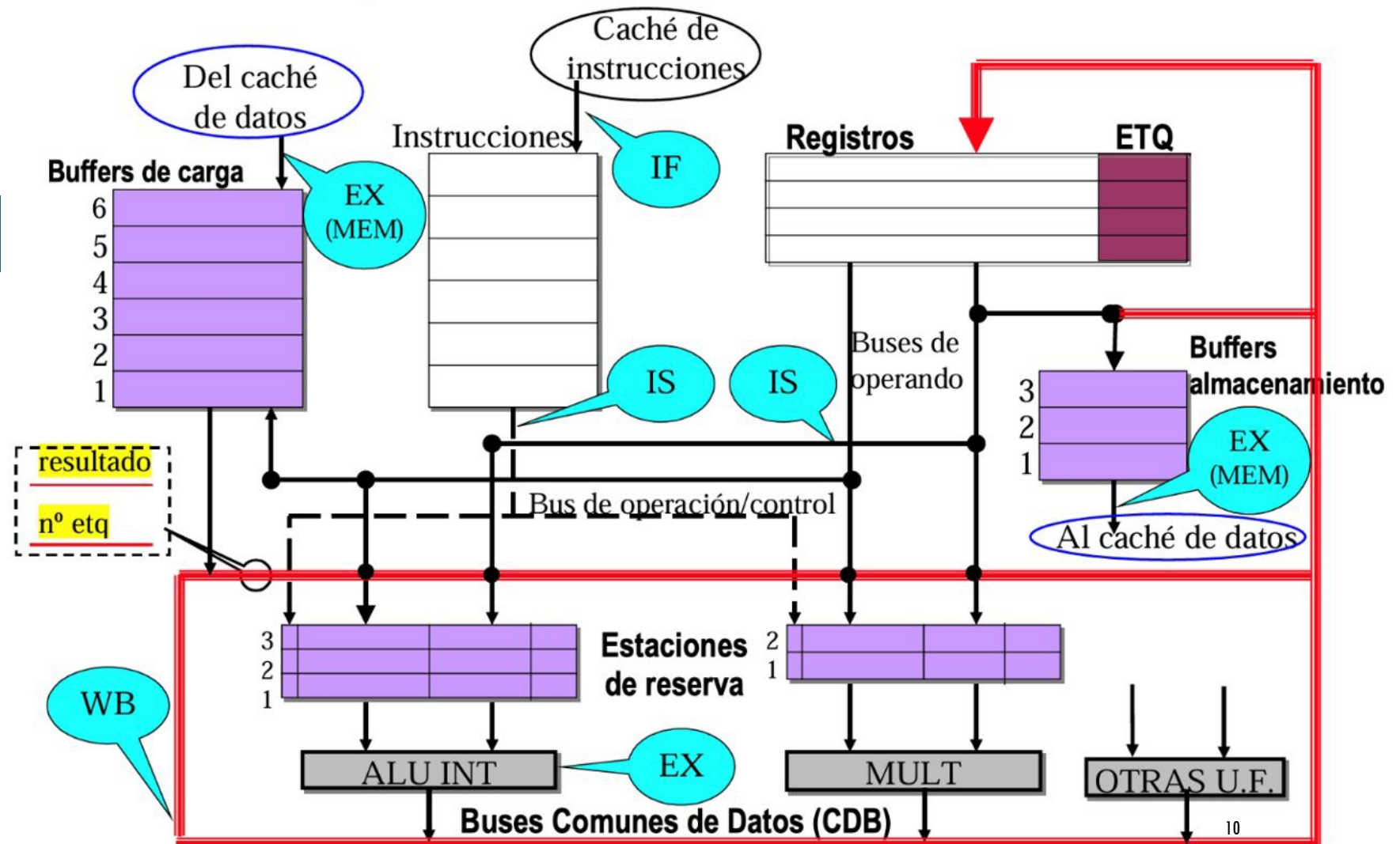
escritura después de escritura

```
i1. R5 ← R4 + R7  
i2. R5 ← R1 + R3
```

PLANIFICADORES DINÁMICOS

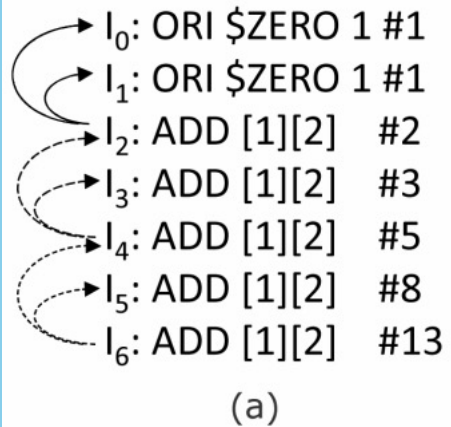
Ejecución fuera de orden

Arquitectura Tomasulo



NUEVO ISA VS PLANIFICADORES DINÁMICOS

STRAIGHT



Cronoplanificación Dinámica

	T1	T2	T3	T4	T5	T6	T7	T8	T9
MULT R3,R2,R1							IF IS: R2,R1 available. <u>R3 in MAX(+0,+0)+4=+4 (T7)</u>		
ADD R4,R3,R1							IF IS: R1 avail.; R3 in +4(-1)=+3. <u>R4 in MAX(+0,+3)+2=+5 (T9)</u>		
SUB R5,R4,R3							IF IS: R4 in +5(-1)=+4; R3 in +4(-2)=+2. <u>R5 in MAX(+4,+2)+2=+6 (T11)</u>		
XOR R4,R6,R1							IF IS: R1,R6 avail.; <u>R4 in MAX(+0,+0)+2=+2 (T8)</u>		
SHR R3,R4,R2							IF IS: R2 avail.; R4 in +2(-1)=+1, <u>R3 in MAX(0,+1)+2=+3 (T10)</u>		

ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

Teoría de la Cronoplanificación

Arquitectura propuesta para la cronoplanificación

Conclusiones y trabajo futuro



DEFINICIONES

Latencia de una Unidad Funcional L_{UF}

Número de ciclos para etapas EX y WB

Tiempo relativo: **RP** (Relative Period)

Número de ciclos para que un registro tenga su valor disponible.

$$\begin{aligned} & Inst\ rd, rs1, rs2 \\ RP1 &= \min(rs1, rs2) \rightarrow \text{se almacena} \\ RP2 &= \max(rs1, rs2) \rightarrow \text{posición} \end{aligned}$$

$$RP = RP_2 + L_{UF}$$

TEORÍA DE LA CRONOPLANIFICACIÓN

Latencia: Constante y conocida a priori



RP conocido en etapa IS



Nuevas estaciones de reserva

Prioridad a instrucciones a corto plazo (SS)

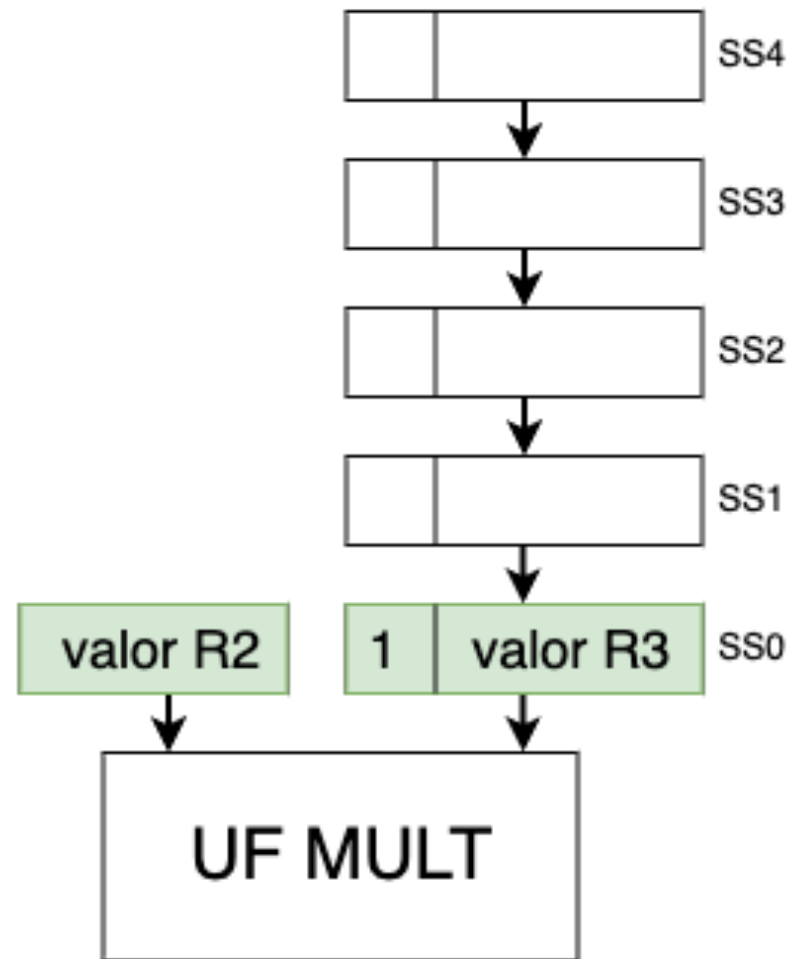
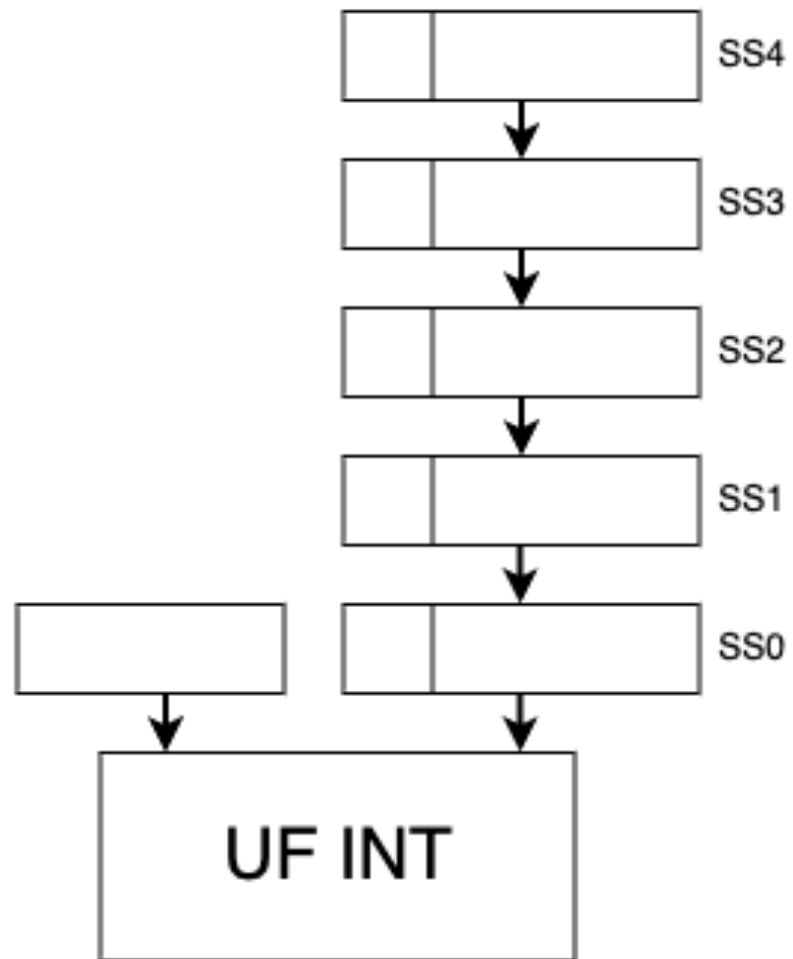
Estaciones para instrucciones a largo plazo (HS)

TEORÍA DE LA CRONOPLANIFICACIÓN

	T0	T1	T2	T3	T4	T5	T6	T7	T8	T9
mult R1, R2, R3	IF	IS	EX	EX	WB					
mult R2, R3, R3		IF	IS			EX	EX	WB		
add R3, R2, R1			IF	IS					EX	WB
add R3, R0, R0				IF	IS	EX	WB			

Ciclo T1: mult R1, R2, R3

$$RP = \max(0,0) + 3$$



Registros

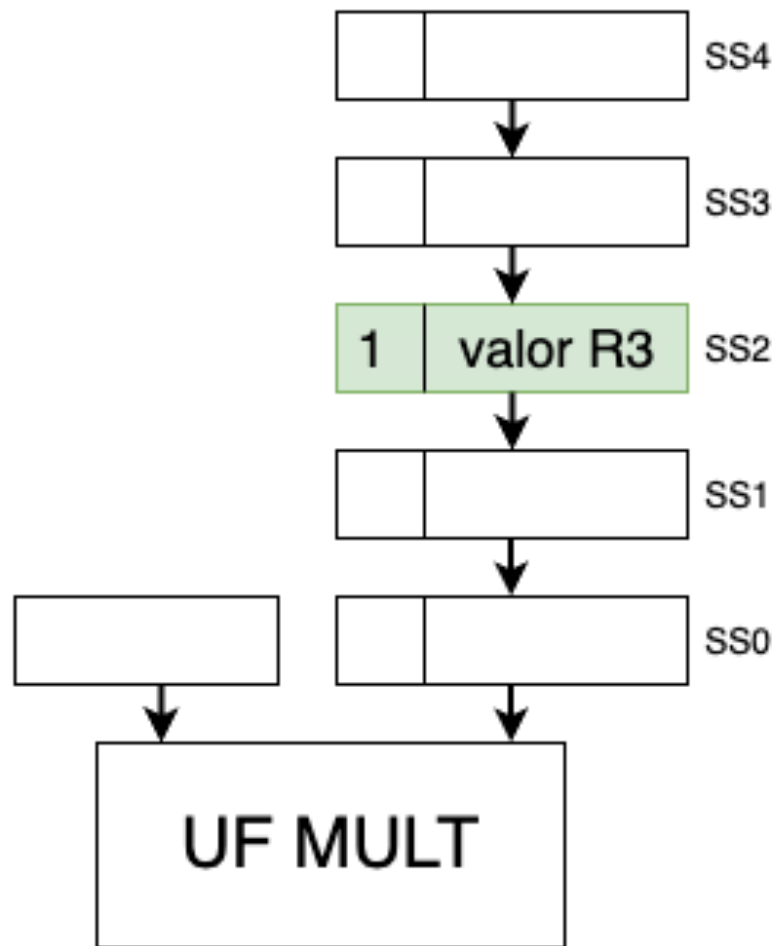
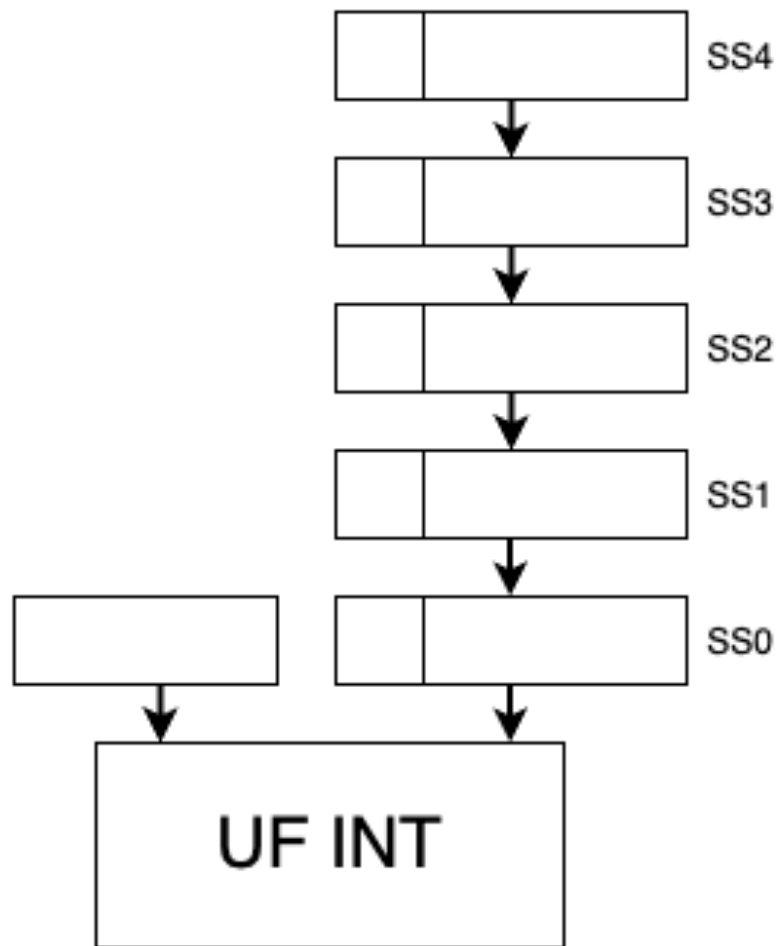
R0	valor
R1	+3 (mult)
R2	valor
R3	valor

RAW

Ciclo T1: mult R1, R2, R3

Ciclo T2: mult R2, R1, R3

$$RP = \max(2,0) + 3$$



Registros

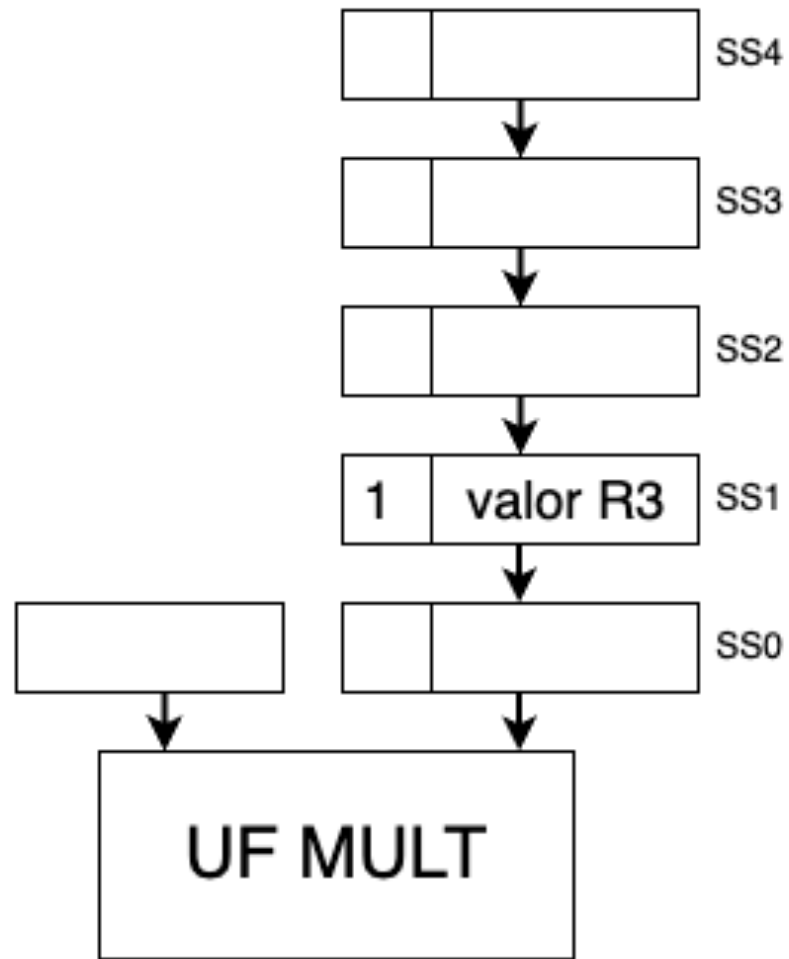
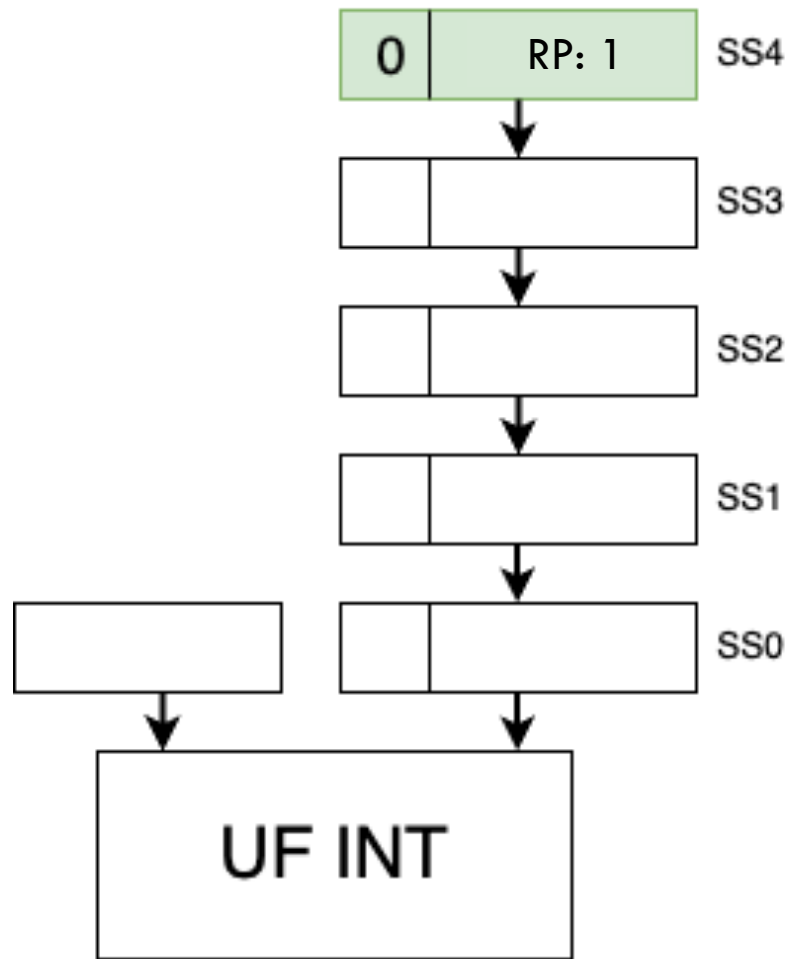
R0	valor
R1	+2 (mult)
R2	+5 (mult)
R3	valor

WAR

Ciclo T2: mult R2, R1, R3

Ciclo T3: add R3, R2, R1

$$RP = \max(4,1) + 2$$



Registros

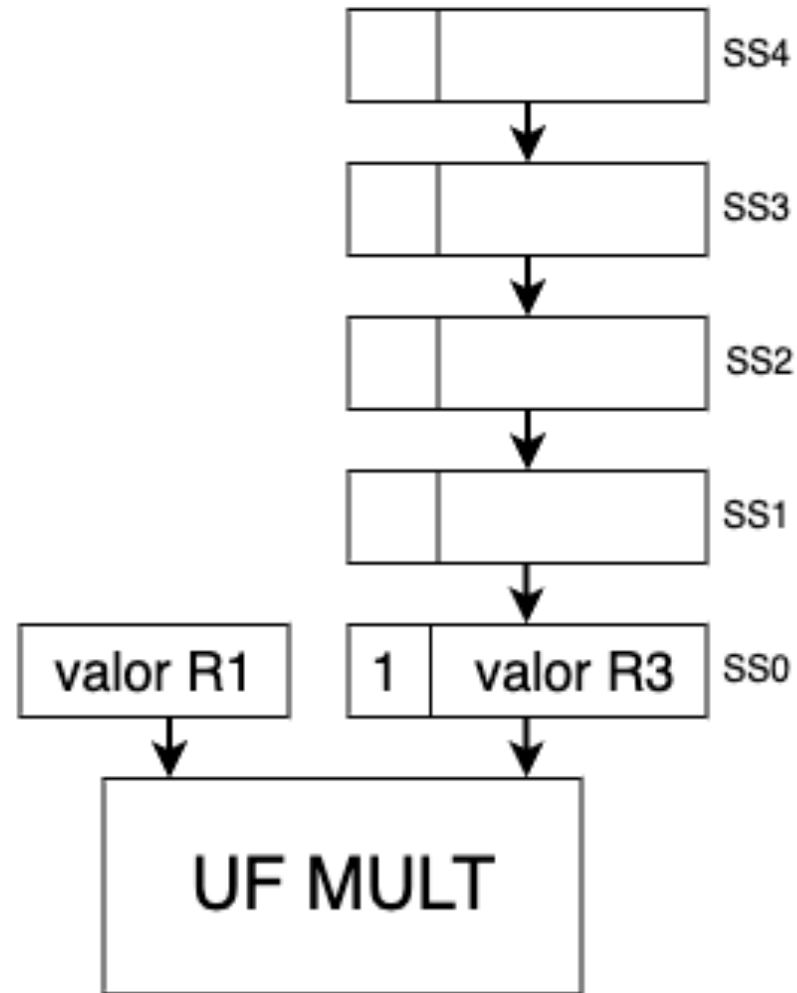
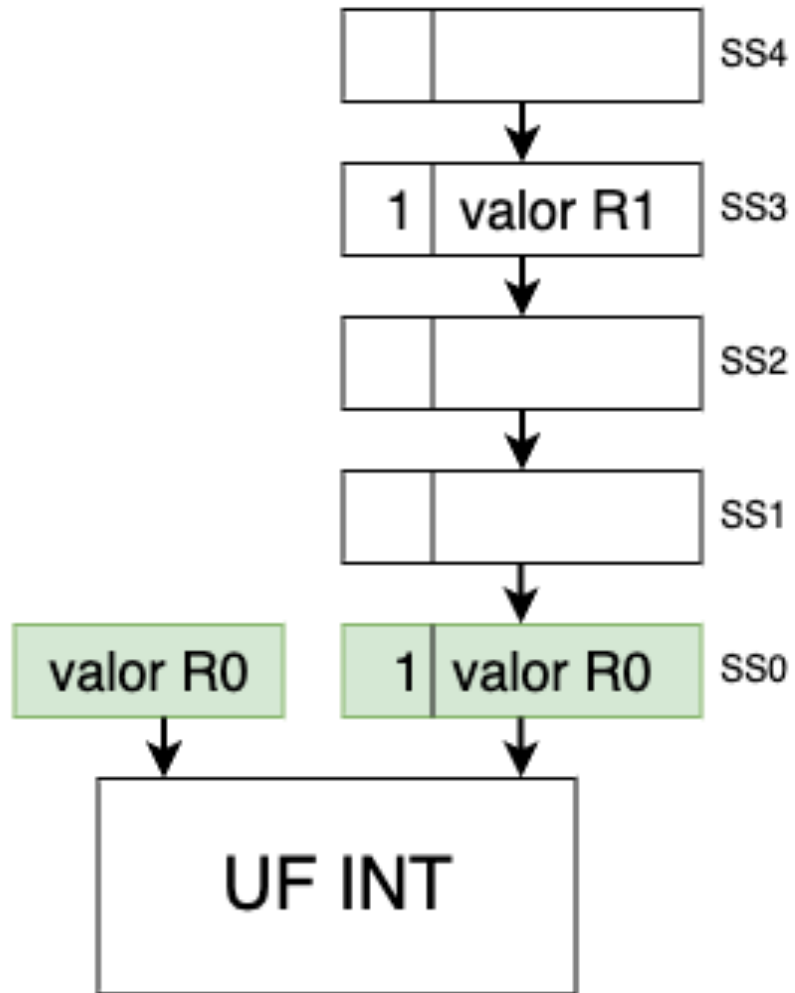
R0	valor
R1	+1 (mult)
R2	+4 (mult)
R3	+6 (add)

WAW

Ciclo T3: add R3, R2, R1

Ciclo T4: add R3, R0, R0

$$RP = \max(0,0) + 3$$



Registros	
R0	+2 (add)
R1	valor
R2	+3 (mult)
R3	+2 (add)

ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

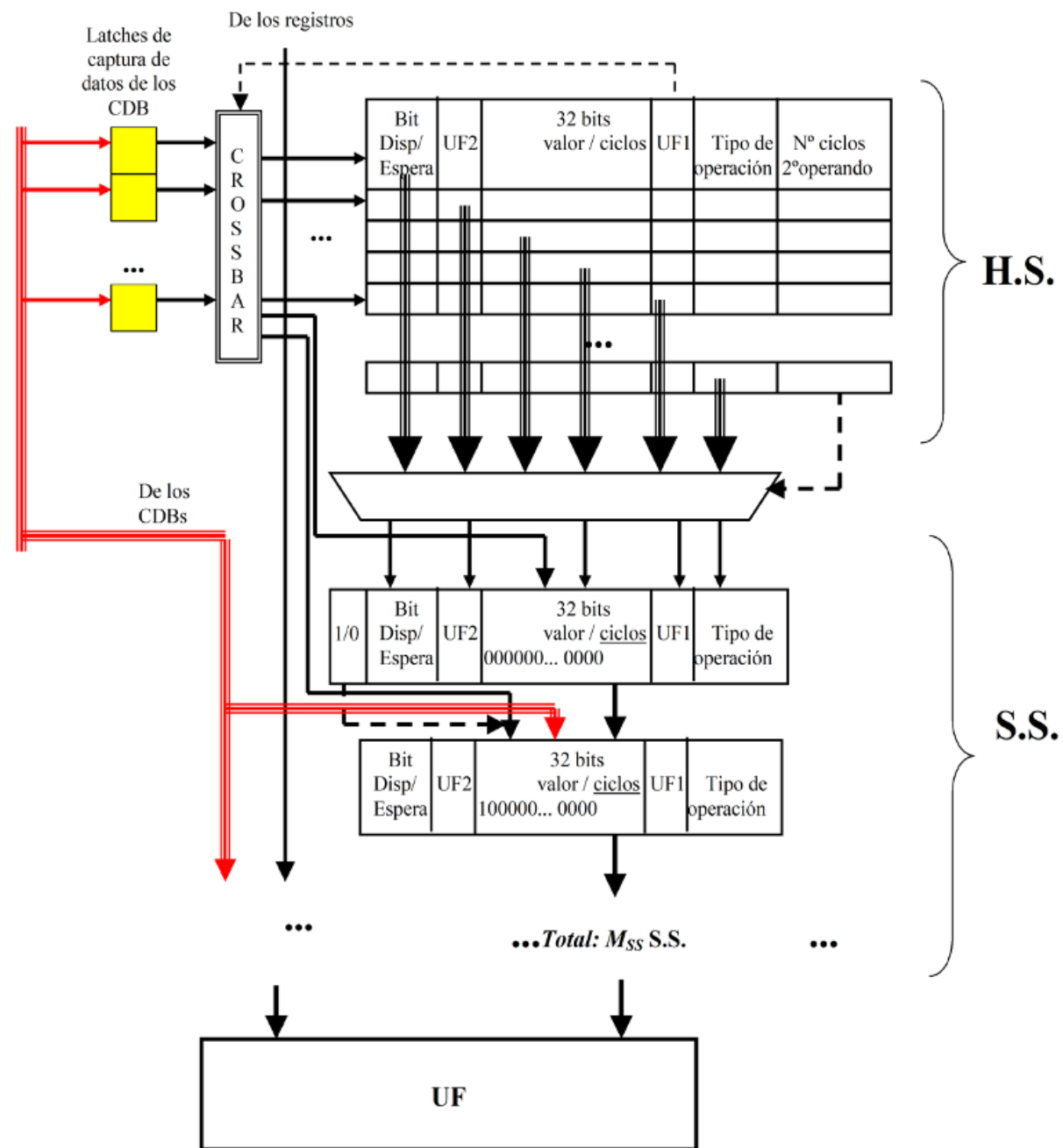
Teoría de la Cronoplanificación

Arquitectura propuesta para la cronoplanificación

Conclusiones y trabajo futuro

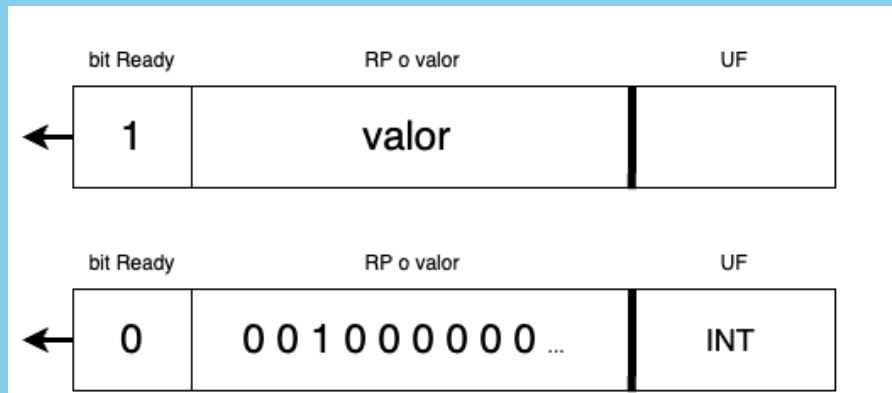


ARQUITECTURA PROPUESTA



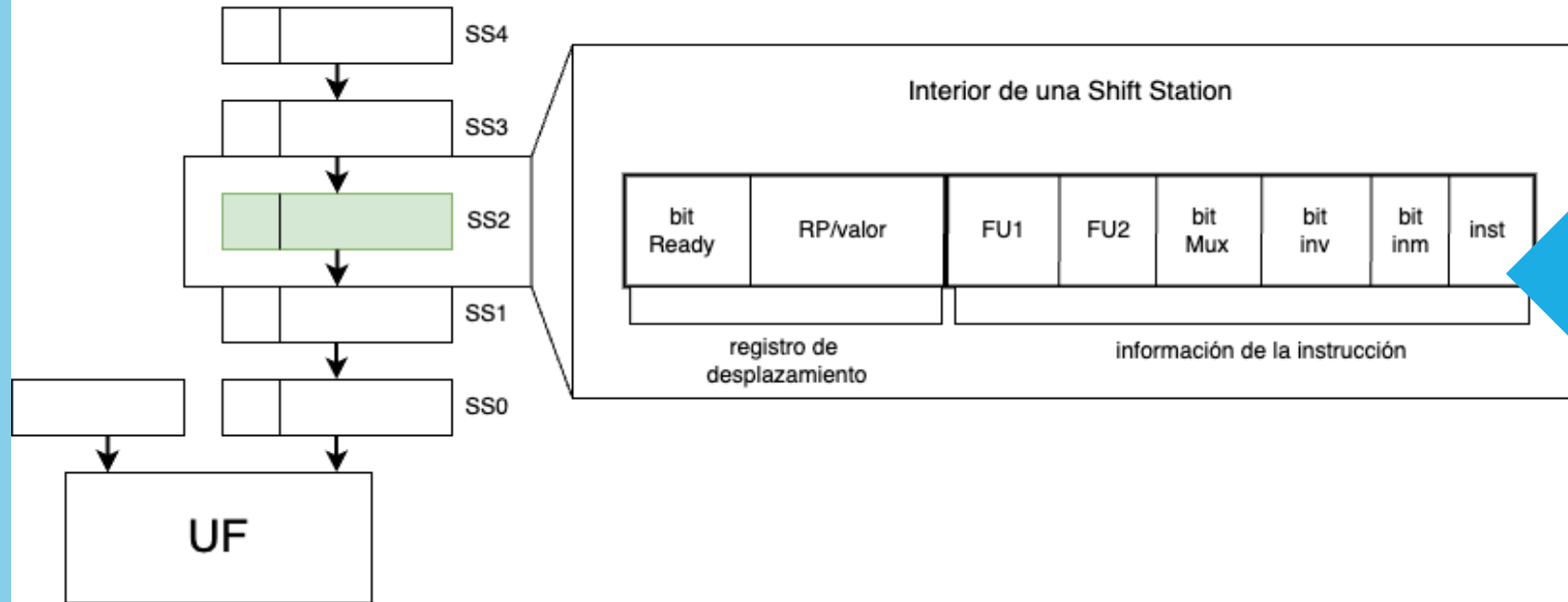
ARQUITECTURA PROPUESTA

Registros



- Encargados de almacenar datos temporales
- Bit Ready: si se tiene valor o RP almacenado

ARQUITECTURA PROPUESTA



Shift Stations (SS)

- Cola de instrucciones
- Almacena el valor del primer registro disponible

Hold Stations (HS)

- Común para todas las unidades funcionales
- Manejo de instrucciones a largo plazo
- Hardware más lento

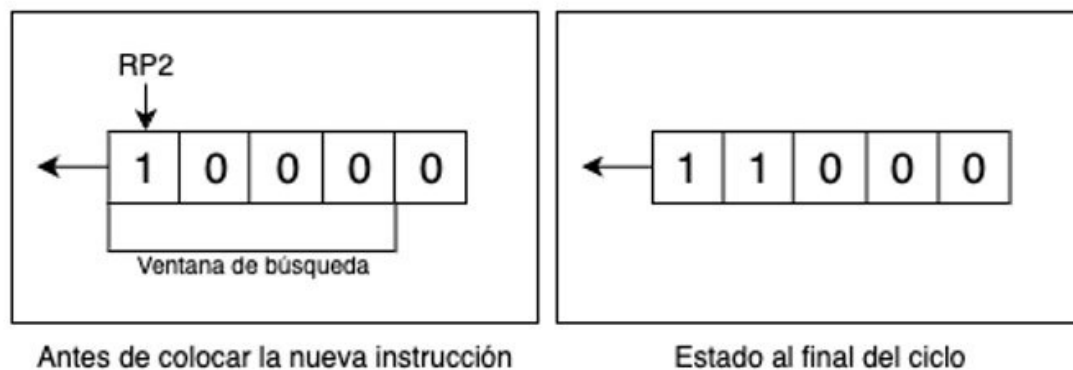
Hold Stations

occupied	HS	case QSD?	bitMux	RP1	RP2	position	destination	FU1	FU2	value1	value2
1	HS0	False	6		18	18	alu_1	inm	alu_1	1	
1	HS1	False	6		10	10	alu_0	inm	alu_1	1	
1	HS2	False	6		14	14	alu_0	inm	alu_1	1	
1	HS3	False	6	6	8	8	alu_1	alu_1	alu_2		
1	HS4	False	6	14	16	16	alu_1	alu_1	alu_2		
1	HS5	False	6	16	18	18	alu_0	alu_2	alu_1		
1	HS6	False	6	18	20	20	alu_2	alu_1	alu_0		
1	HS7	False	6	8	10	10	alu_2	alu_2	alu_1		
1	HS8	False	6	10	12	12	alu_1	alu_1	alu_2		
1	HS9	False	6	12	14	14	alu_2	alu_2	alu_1		

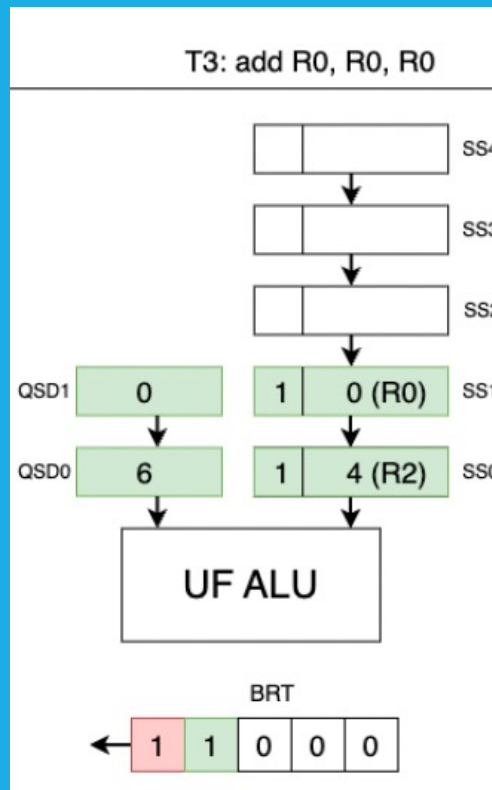
ARQUITECTURA PROPUESTA

Binary Reservation Table (BRT):

- Marca que SS está ocupada
- Solucionar conflictos estructurales
- n: número de desplazamiento



Queue of Structural Dependency (QSD)



Dar prioridad a instrucciones a corto plazo a pesar de tener dependencias estructurales

Almacenan RP2 y UF2

ARQUITECTURA PROPUESTA

Program Counter (PC) y ventana de emisión

- Manejo de instrucciones con dos listas
 - Instrucciones a procesar en el ciclo
 - Instrucciones bloqueadas

REGISTERS

Register	RP	FU	value	locked
R0	+0	alu_0	0	False
R1	+0		2	False
R2	+0		4	False
R3	+2		None	False
R4	+0		8	False

Instrucciones: [0,1,2,3,4]
Inst. Bloqueadas: [1,3]
Siguiete Inst: 5

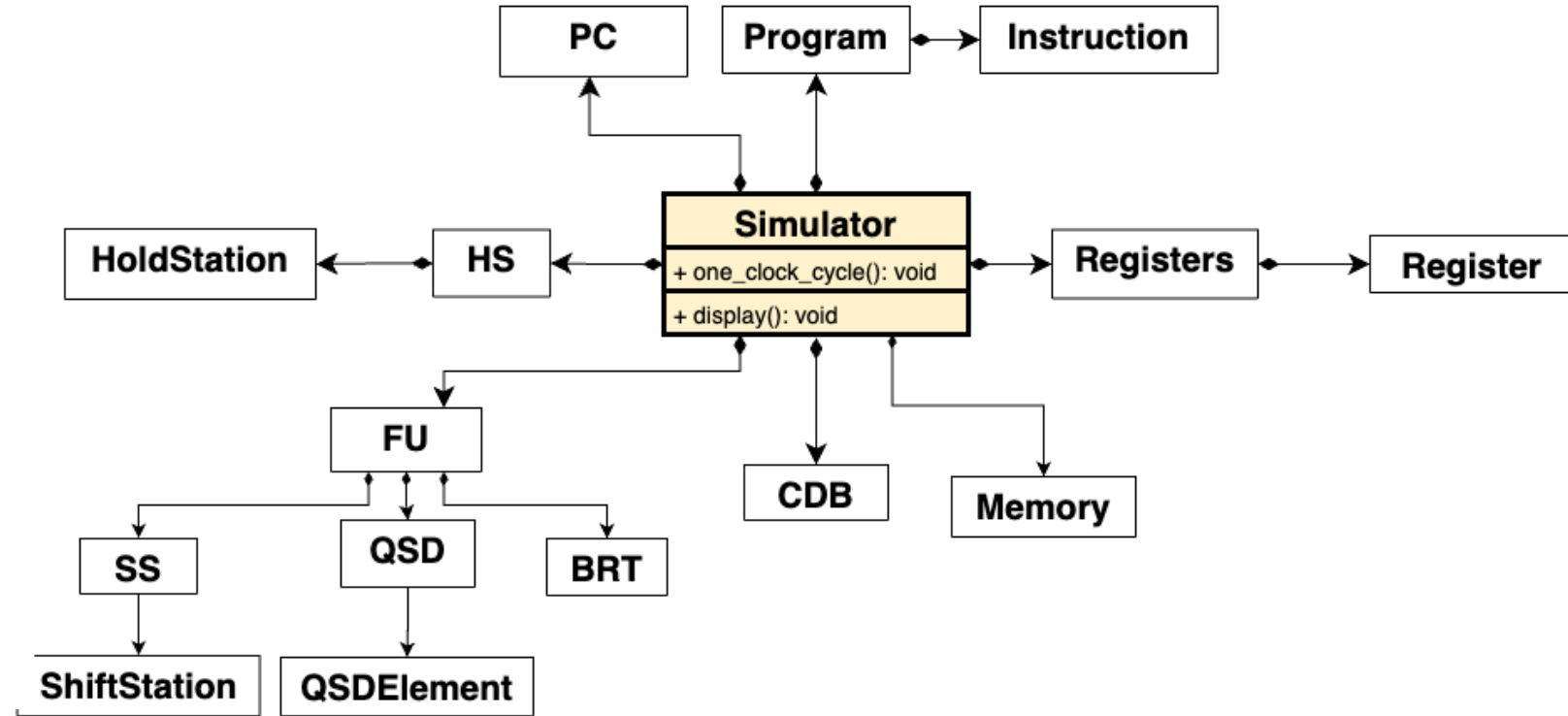
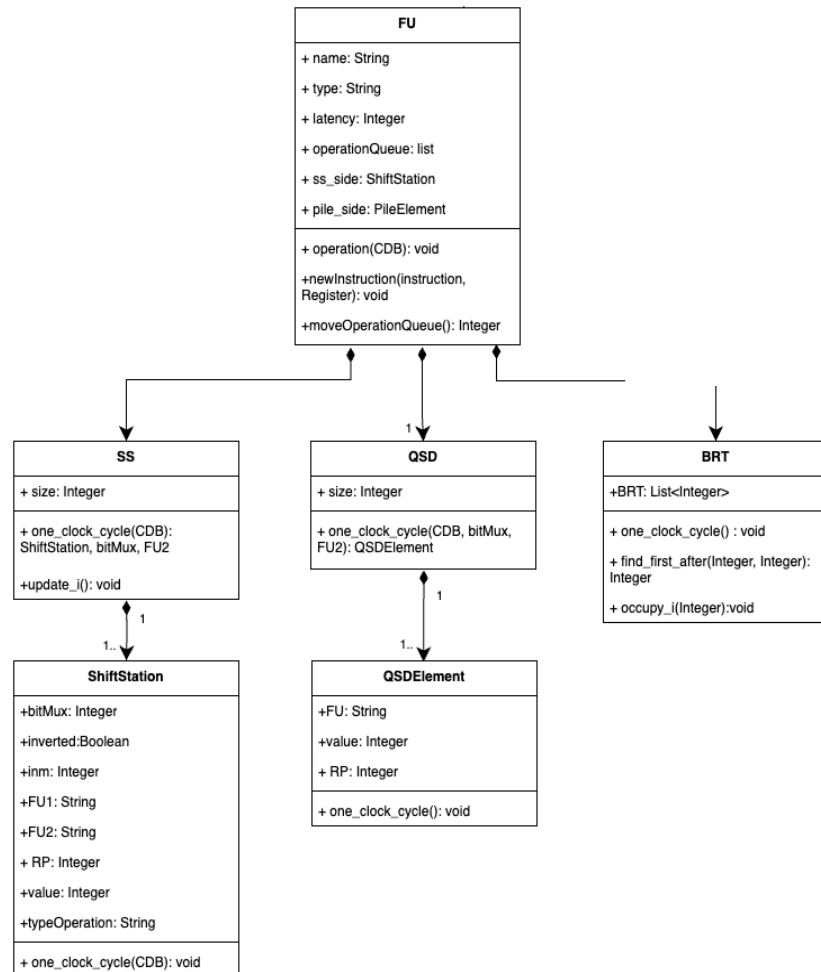


Instrucciones: [1,3,5,6,7]
Inst. Bloqueadas: [1,7]
Siguiete Inst: 8

ARQUITECTURA PROPUESTA

Bit MUX	Explicación
type_0	$RP1 = RP2 = 0, n = 0$ La instrucción se ejecutará en el siguiente ciclo
type_1	$RP1 = RP2 = 0, n = 0$. La instrucción se ejecutará en n ciclos Se necesita QSD
type_2	$RP2 > 0, n = 0$. La instrucción se ejecutará en $RP2$ ciclos
type_3	$RP2 > 0, n > 0$. La instrucción se ejecutará en $RP2 + n$ ciclos Se necesita QSD
type_4	BLOQUEO por agotamiento de BRT. Tanto ventana como RP máximo
type_5	BLOQUEO por agotamiento de HS
type_6	$RP2 > n^{\circ} SS$ y $n = 0$ Se necesita HS
type_7	$RP2 + n > n^{\circ} QSD$ y $n > 0$ Se necesita HS Se necesita QSD
type_8	BLOQUEO por registros bloqueados

DIAGRAMA DE CLASES



Tecnologías usadas

- **Simulador:** Python 3.8.0
- **Aplicación:** Tkinter, Rich text, Pandas, matplotlib

CHRONOs PLAN

Load Files

Import csv No Instructions selected

Import csv Selected file: config_sim.csv

Configuration

n° of SS: 8

n° of registers: 5

QSD size: 3

Memory size: 32

Multiplicity: 5

HS?: ☐ False ☒ True

N° of HS: 10

N° of future cycles: 30

FU	n°	lat
ALU	3	2
MULT	3	3
DIV	3	3
STORE	3	2
LOAD	3	2
TRANS	3	5

SAVE config

Display Settings

Display ALU: FU ☐ BRT ☐

Display MULT: FU ☐ BRT ☐

Display DIV: FU ☐ BRT ☐

Display TRANS: FU ☐ BRT ☐

Display STORE: FU ☐ BRT ☐

Display LOAD: FU ☐ BRT ☐

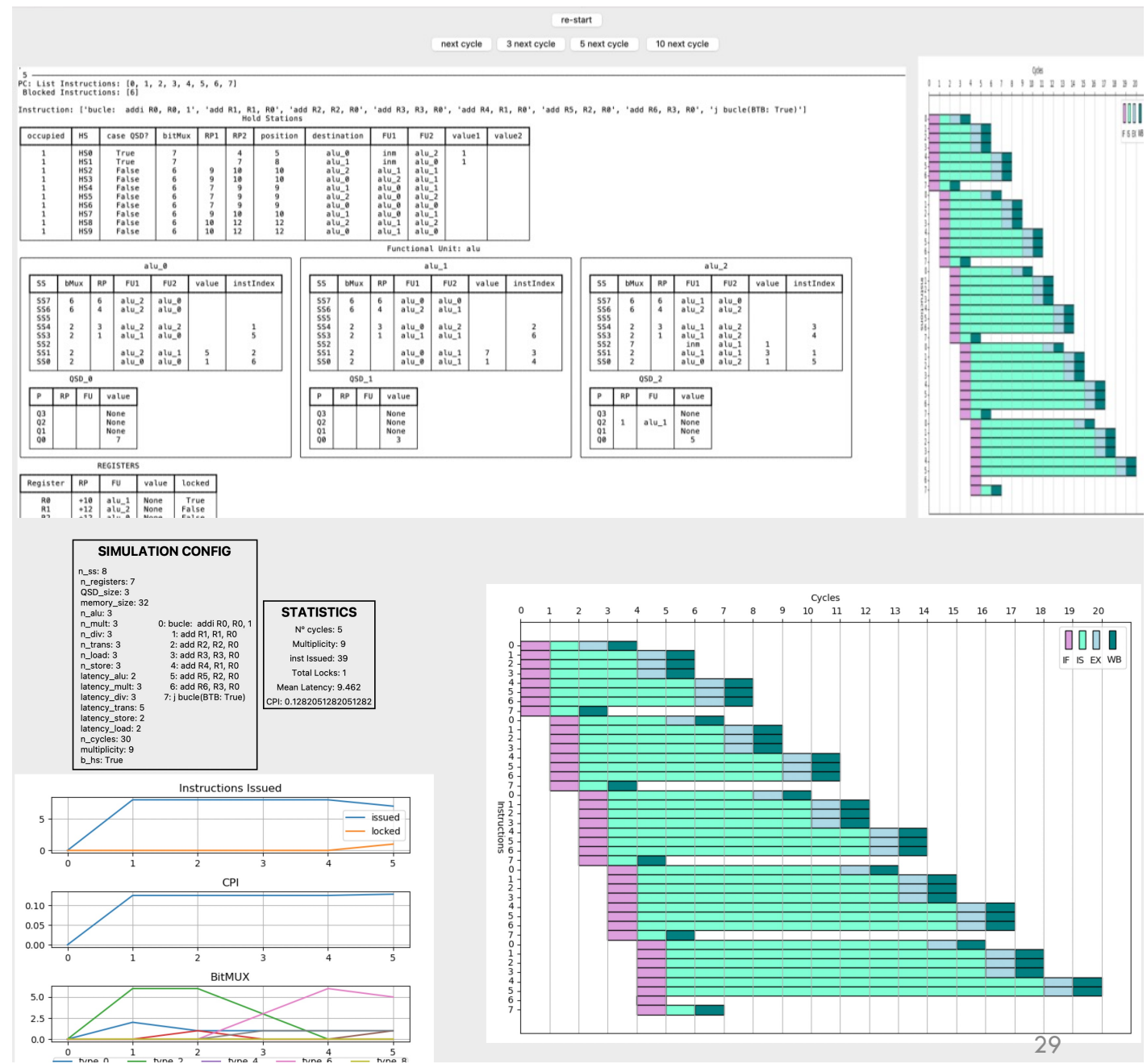
Display JUMP: BRT ☐

Display memory ☐

Display CDB ☐

Display HS ☐

START



ÍNDICE

Motivación y Objetivo

Problemática y Estado del arte

Teoría de la Cronoplanificación

Arquitectura propuesta para la Cronoplanificación

Conclusiones y trabajo futuro



BLOQUEOS

Tomasulo

Agotamiento Estaciones de Reserva

Agotamiento de etiquetas

Cronoplanificación

Agotamiento de las Hold Stations
Agotamiento de las Shift Stations

RP2 mayor que el n° de ciclos futuro

RESULTADOS Y CONCLUSIONES

Estaciones de
Reserva (SS) simples

No existe necesidad
de modificar ISA

Bloqueos similares a
Tomasulo

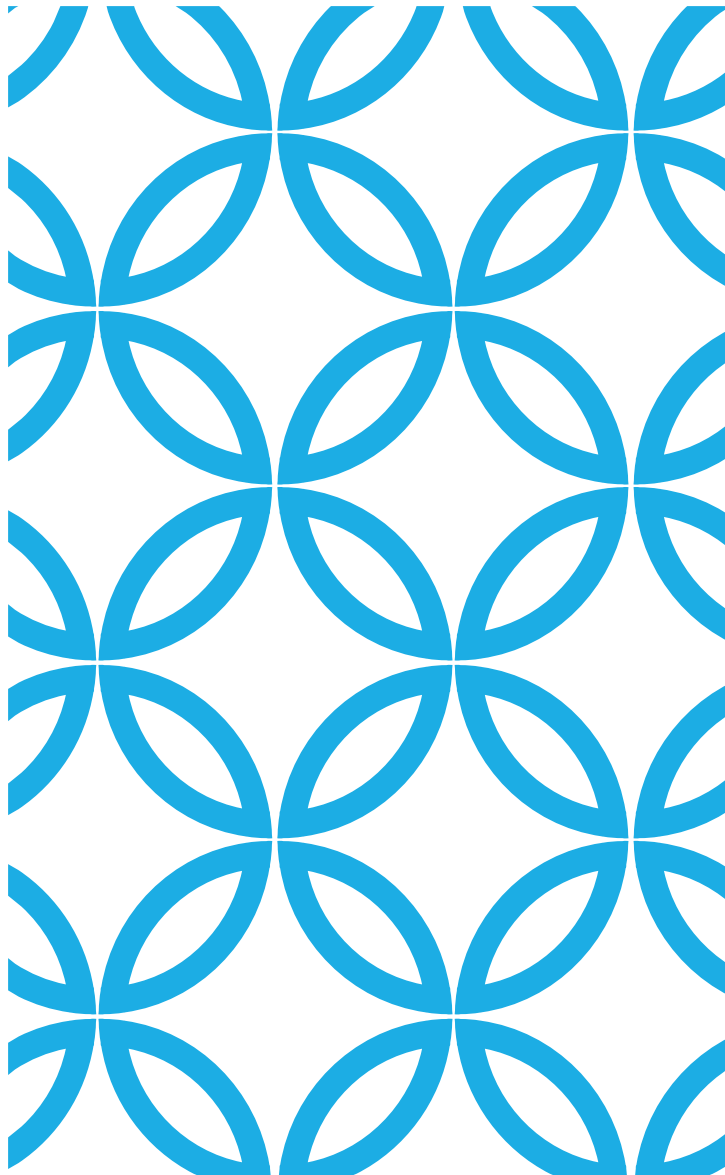
Bloqueo de instrucciones

TRABAJO FUTURO

Implementación
en Hardware

Manejo de ramificaciones
en ejecución
Mecanismo similar al ROB

Manejo de instrucciones
con latencia no constante



MUCHAS GRACIAS
DEMOSTRACIÓN DEL SIMULADOR
