



# Trabajo de Fin de Grado

---

## Simulador didáctico de arquitectura de computadores

*Didactic simulator for Computer Architecture .*

Adrián Abreu González

---

La Laguna, 9 de mayo de 2017

D. **Iván Castilla Rodríguez**, con N.I.F. 78.565.451-G profesor Titular de Universidad adscrito al Departamento de Nombre del Departamento de la Universidad de La Laguna, como tutor

## **C E R T I F I C A**

Que la presente memoria titulada:

*“Simulador didáctico de arquitectura de computadores.”*

ha sido realizada bajo su dirección por D. **Adrián Abreu González**, con N.I.F. 54.111.250-R.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 9 de mayo de 2017

# Agradecimientos

XXX

XXX

XXX

XXX

# Licencia

\* Si quiere permitir que se compartan las adaptaciones de tu obra y quieres permitir usos comerciales de tu obra (licencia de Cultura Libre) indica:



© Esta obra está bajo una licencia de Creative Commons Reconocimiento 4.0 Internacional.

## Resumen

*El objetivo de este trabajo ha sido .... bla, bla, bla bla, bla, bla bla, bla, bla*

*La competencia [E6], que figura en la guía docente, indica que en la memoria del trabajo se ha de incluir: antecedentes, problemática o estado del arte, objetivos, fases y desarrollo del proyecto, conclusiones, y líneas futuras.*

*Se ha incluido el apartado de 'Licencia' con todas las posibles licencias abiertas (Creative Commons). En el caso en que se decida hacer público el contenido de la memoria, habrá que elegir una de ellas (y borrar las demás). La decisión de hacer pública o no la memoria se indica en el momento de subir la memoria a la Sede Electrónica de la ULL, paso necesario en el proceso de presentación del TFG.*

*El documento de memoria debe tener un máximo de 50 páginas.*

*No se deben dejar páginas en blanco al comenzar un capítulo, ya que el documento no está pensado para se impreso sino visionado con un lector de PDFs.*

*También es recomendable márgenes pequeños ya que, al firmar digitalmente por la Sede, se coloca un marco alrededor del texto original.*

*El tipo de letra base ha de ser de 14ptos.*

**Palabras clave:** Palabra reservada1, Palabra reservada2, ...

## Abstract

Here should be the abstract in a foreing language...

***Keywords:*** *Keyword1, Keyword2, Keyword3, ...*

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Sección Uno . . . . .	1
1.2. Sección Dos . . . . .	1
1.3. Sección Tres . . . . .	1
1.4. Sección Cuatro . . . . .	1
<b>2. Antecedentes</b>	<b>3</b>
2.1. Primera sección de otro capítulo . . . . .	3
<b>3. Estado del arte</b>	<b>4</b>
3.1. Primera sección de este capítulo . . . . .	4
3.2. Segunda sección de este capítulo . . . . .	4
3.3. Tercera sección de este capítulo . . . . .	4
<b>4. Objetivos y fases</b>	<b>5</b>
<b>5. Desarrollo del proyecto</b>	<b>6</b>
5.1. Migración del nucleo de la aplicación . . . . .	6
5.2. Migración de la máquina superescalar . . . . .	6
5.3. Desarrollo de la interfaz . . . . .	6
5.4. Integración interfaz - máquina superescalar . . . . .	6
<b>6. Ampliacion de funcionalidades</b>	<b>7</b>
<b>7. Conclusiones y líneas futuras</b>	<b>9</b>
<b>8. Summary and Conclusions</b>	<b>10</b>
8.1. First Section . . . . .	10
<b>9. Presupuesto</b>	<b>11</b>
9.1. Sección Uno . . . . .	11
<b>A. Título del Apéndice 1</b>	<b>12</b>
A.1. Algoritmo XXX . . . . .	12
A.2. Algoritmo YYY . . . . .	12

<b>B. Título del Apéndice 2</b>	<b>13</b>
B.1. Otro apéndice: Sección 1 . . . . .	13
B.2. Otro apéndice: Sección 2 . . . . .	13
<b>Bibliografía</b>	<b>13</b>



# Índice de figuras

1.1. Ejemplo . . . . .	2
------------------------	---

# Índice de tablas

9.1. Presupuesto . . . . .	11
----------------------------	----

# Capítulo 1

## Introducción

### 1.1. Sección Uno

- Item 1
- Item 2
- Item 3
- Item 4

### 1.2. Sección Dos

- Item 1
- Item 2
- Item 3

### 1.3. Sección Tres

Bla, bla, bla

### 1.4. Sección Cuatro

Bla, bla, bla

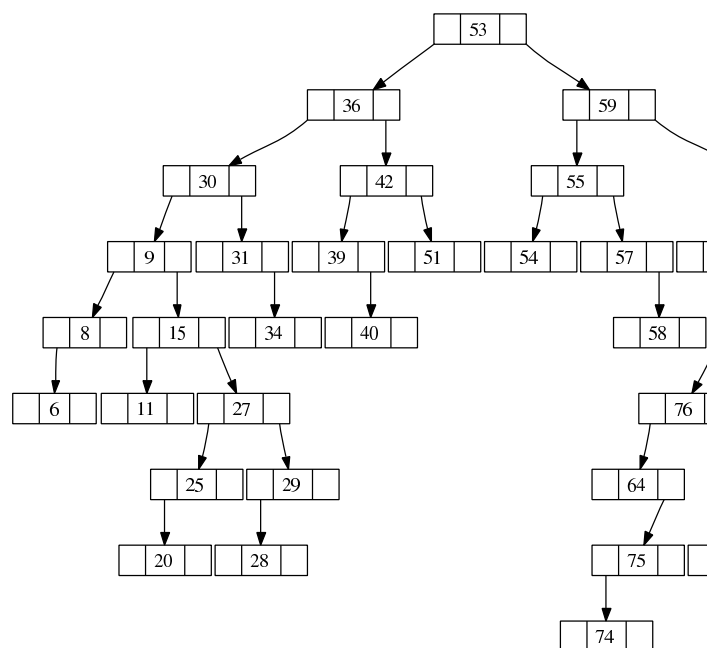


Figura 1.1: Ejemplo

# Capítulo 2

## Antecedentes

Los capítulos intermedios servían para cubrir los siguientes aspectos: antecedentes, problemática o estado del arte, objetivos, fases y desarrollo del proyecto.

En el capítulo anterior se ha introducido bla, bla, bla ....

### 2.1. Primera sección de otro capítulo

# Capítulo 3

## Estado del arte

Los capítulos intermedios servirán para cubrir los siguientes aspectos: antecedentes, problemática o estado del arte, objetivos, fases y desarrollo del proyecto.

Bla, Bla, Bla, .....

**3.1. Primera sección de este capítulo**

**3.2. Segunda sección de este capítulo**

**3.3. Tercera sección de este capítulo**

# Capítulo 4

## Objetivos y fases

El desarrollo del trabajo se puede dividir en cuatro fases principales que han permitido llevar a cabo la tarea de reescribir esta aplicación en web.

1. Migración del “núcleo de la aplicación”: Es decir las estructuras básicas comunes a las máquinas y el punto de entrada para la generación de las estructuras correspondientes, es decir, el analizador léxico de MIPS.
2. Migración de la máquina superescalar: El proceso completo de reescribir todas las estructuras de la máquina y su correcto funcionamiento en javascript.
3. Desarrollo de la interfaz: El desarrollo tanto de un diseño remodelado con las tecnologías web, como de componentes que gestionen correctamente su propio estado.
4. Integración interfaz-máquina: Realizar los enlaces necesarios entre el código desarrollado y la interfaz que permitan que .

# Capítulo 5

## Desarrollo del proyecto

### 5.1. Migración del nucleo de la aplicación

El núcleo de la aplicación se basa en el uso del generador de analizadores léxicos FLEX para parsear un conjunto de instrucciones similar al MIPS IV. Para realizar el proceso de migración he tenido que comprender primero el funcionamiento de los analizadores léxicos.

Aislando la implementación original y haciendo uso de una librería desarrollada para Javascript de Lex, se pudo conseguir en una escasa cantidad de tiempo tener en funcionamiento el código en Javascript.

Además, era imprescindible migrar las estructuras básicas que se crean cuando se carga el código, es decir: Las unidades funcionales y los bloques básicos y sucesores.

En este punto ya se tomó una de las decisiones más importantes para el desarrollo. Se decidió utilizar Typescript.

### 5.2. Migración de la máquina superescalar

### 5.3. Desarrollo de la interfaz

### 5.4. Integración interfaz - máquina superescalar



# Capítulo 6

## Ampliacion de funcionalidades

Existen cuatro extras principales que se han desarrollado en este trabajo de fin de grado.

### 1. Documentación

El primero de ellos es la recuperación de la documentación. En su proyecto original el profesor Iván Castilla elaboró una extensa documentación sobre SIMDE que contenía consejos sobre el desarrollo de aplicaciones para las distintas máquinas y no solo eso, sino que además explicaba el funcionamiento de las máquinas. Por desgracia, esta documentación fue realizada en formato .HLP, un formato abandonado en Windows Vista. Es por eso, que utilizando herramientas de extracción, se pudieron recuperar los textos originales de parte del profesor. Ahora, para integrar esta documentación en la aplicación web lo mejor era hacer un sistema web también. Por ello se ha migrado la documentación a un formato muchísimo más mantenible utilizando un generador de contenido estático basado en NodeJS (Hexo) y un tema apropiado, se ha conseguido que la documentación esté integrada de nuevo.

IMAGEN DOCUMENTACIÓN

### 2. Parseador El siguiente extra es la mejora del parseador. Una de las cosas más molestas de la versión original de SIMDE es que ante un error de código sólo muestra el mensaje “error”. Ahora, tras una serie de modificaciones en el parseados de código, se muestran los siguientes errores: Operando erróneo Opcode desconocido Etiqueta repetida

IMAGENES ERROR

Además, se muestra la línea del error. Esto era muy importante ya que en varias ocasiones la frustración de buscar en un código largo los errores

veces distraía a los usuarios de la finalidad del SIMDE; que no es otra que comprender y visualizar el funcionamiento de la máquina Superescalar.

3. Batch Mode
4. Histórico

## Capítulo 7

# Conclusiones y líneas futuras

Este capítulo es obligatorio. Toda memoria de Trabajo de Fin de Grado debe incluir unas conclusiones y unas líneas de trabajo futuro

# Capítulo 8

## Summary and Conclusions

This chapter is compulsory. The memory should include an extended summary and conclusions in english.

### 8.1. First Section

# Capítulo 9

## Presupuesto

### 9.1. Sección Uno

Descripción	Coste
200 Horas de trabajo	8000 €
Ordenador para desarrollo	1300€
Total	9300 €

Tabla 9.1: Presupuesto

# Apéndice A

## Título del Apéndice 1

### A.1. Algoritmo XXX

```
*****
*
* Fichero .h
*
*****
*
* AUTORES
*
*
* FECHA
*
*
* DESCRIPCION
*
*
*****/
```

### A.2. Algoritmo YYY

```
/******
*
* Fichero .h
*
*****
*
* AUTORES
*
* FECHA
*
* DESCRIPCION
*
*
*****/
```

# Apéndice B

## Título del Apéndice 2

### B.1. Otro apéndice: Sección 1

Texto

### B.2. Otro apéndice: Sección 2

Texto

# Bibliografía

- [1] ACM LaTeX Style. [http://www.acm.org/publications/latex\\_style/](http://www.acm.org/publications/latex_style/).
- [2] FACOM OS IV SSL II USER'S GUIDE, 99SP0050E5. Technical report, 1990.
- [3] D. H. Bailey and P. Swarztrauber. The fractional Fourier transform and applications. *SIAM Rev.*, 33(3):389–404, 1991.
- [4] A. Bayliss, C. I. Goldstein, and E. Turkel. An iterative method for the Helmholtz equation. *J. Comp. Phys.*, 49:443–457, 1983.
- [5] C. Darwin. *The Origin Of Species*. November 1859.
- [6] C. Goldstein. Multigrid methods for elliptic problems in unbounded domains. *SIAM J. Numer. Anal.*, 30:159–183, 1993.
- [7] P. Swarztrauber. *Vectorizing the FFTs*. Academic Press, New York, 1982.
- [8] S. Taásan. *Multigrid Methods for Highly Oscillatory Problems*. PhD thesis, Weizmann Institute of Science, Rehovot, Israel, 1984.