

# Tabla de Contenido

1	Introducción . . . . .	1
1.1	Motivación . . . . .	1
1.2	Objetivos del trabajo . . . . .	2
2	Marco teórico y estado del arte . . . . .	4
2.1	Conceptos elementales . . . . .	4
2.1.1	Sistemas digitales . . . . .	4
2.1.2	Arquitectura de computadores . . . . .	5
2.1.3	Representación binaria en punto flotante . . . . .	10
2.2	RISC-V . . . . .	11
2.2.1	Visión general al <i>ISA</i> de <i>RISC-V</i> . . . . .	12
2.2.2	La Memoria en <i>RISC-V</i> . . . . .	13
2.2.3	Codificación del largo de instrucciones . . . . .	14
2.2.4	Excepciones, trampas e interrupciones . . . . .	15
2.2.5	El <i>ISA</i> base y extensiones implementadas, <i>RV32IMF</i> . . . . .	16
2.2.5.1	Instrucciones enteras, <i>RV32IM</i> . . . . .	19
2.2.5.2	Unidad de punto flotante ( <i>FPU</i> ), <i>RV32F</i> . . . . .	22
2.3	Revisión del estado del arte . . . . .	25
2.3.1	<i>A RISC-V Instruction Set Processor-Micro- architecture Design and Analysis</i> . . . . .	25
2.3.2	<i>Open-Source RISC-V Processor IP Cores for FPGAs – Overview and Evaluation</i> . . . . .	29
2.3.3	<i>RISC-V based implementation of Programmable Logic Controller on FPGA for Industry 4.0</i> . . . . .	30
2.3.4	<i>Design and Verification Environment for RISC-V Processor Cores</i> . . . . .	31
2.3.5	<i>Co-verification design flow for HDL Languages</i> . . . . .	31
2.3.6	Ejemplo de una implementación real: <i>HiFive Rev B</i> . . . . .	32
2.4	Tarjeta <i>FPGA</i> y <i>software</i> utilizado . . . . .	34
2.4.1	<i>Nexys 4</i> . . . . .	34
2.4.2	<i>Vivado Design Suite</i> de <i>Xilinx</i> . . . . .	35
2.4.3	<i>RARS 1.5</i> . . . . .	36
3	Metodología de trabajo . . . . .	37
3.1	Metodología de diseño <i>Top-Down</i> . . . . .	37
3.2	Flujo de trabajo . . . . .	38
3.3	Metodología de verificación . . . . .	39
4	Diseño propuesto . . . . .	41
4.1	<i>Instruction Decoder</i> . . . . .	43
4.2	<i>Register Files</i> . . . . .	45

4.3	Unidades de memoria . . . . .	46
4.3.1	<i>Cache Controller</i> . . . . .	47
4.4	<i>Arithmetic Logic Unit (ALU)</i> . . . . .	50
4.5	<i>Floating Point Unit (FPU)</i> . . . . .	51
4.5.1	<i>FP Arithmetic Unit</i> . . . . .	55
4.5.1.1	<i>FP Arithmetic Unit: ADDER</i> . . . . .	63
4.5.1.2	<i>FP Arithmetic Unit: MULTIPLIER</i> . . . . .	64
4.5.1.3	<i>FP Arithmetic Unit: DIVISOR</i> . . . . .	65
4.5.1.4	<i>FP Arithmetic Unit: significands_divisor</i> . . . . .	66
4.5.1.5	<i>FP Arithmetic Unit: SQRT</i> . . . . .	71
4.5.1.6	<i>FP Arithmetic Unit: significand_sqrt</i> . . . . .	72
4.5.1.7	<i>FP Arithmetic Unit: Exceptions and trivial cases checker</i> . . . . .	76
4.5.1.8	<i>FP Arithmetic Unit: NORMALIZER</i> . . . . .	78
4.5.1.9	<i>FP Arithmetic Unit: ROUNDER</i> . . . . .	81
4.5.2	<i>FP Converter Unit</i> . . . . .	81
4.5.2.1	<i>FP Converter Unit: exceptions_checker</i> . . . . .	85
4.5.2.2	<i>FP Converter Unit: fp_to_integer_unit</i> . . . . .	87
4.5.2.3	<i>FP Converter Unit: Integer rounder</i> . . . . .	87
4.5.2.4	<i>FP Converter Unit: integer_to_fp_unit</i> . . . . .	88
4.5.2.5	<i>FP Converter Unit: NORMALIZER</i> . . . . .	89
4.5.2.6	<i>FP Converter Unit: ROUNDER</i> . . . . .	90
4.6	<i>Core Complex</i> . . . . .	90
4.6.1	<i>Single-Cycle Version</i> . . . . .	93
4.6.2	<i>Pipelined Version</i> . . . . .	100
4.7	<i>System on a Chip</i> . . . . .	109
4.7.1	<i>debounce module</i> . . . . .	115
5	Verificación, pruebas e implementación final . . . . .	116
5.1	Etapas de verificación . . . . .	116
5.1.1	Testeo de: <i>memory.sv</i> . . . . .	116
5.1.2	Testeo de: <i>ALU.sv</i> . . . . .	118
5.1.3	Testeo de: <i>fp_converter.sv</i> . . . . .	120
5.1.4	Testeo de: <i>fp_arithmetic_unit.sv</i> . . . . .	124
5.1.5	Testeo de: <i>FPU.sv</i> . . . . .	128
5.1.6	Testeo de: <i>riscv32imf_singlecycle.sv</i> y <i>riscv32imf_pipeline.sv</i> . . . . .	128
5.1.7	Testeo de: <i>TOP.sv</i> . . . . .	132
5.2	Pruebas en <i>hardware</i> . . . . .	133
5.2.1	<i>fibonacci_inputs.s</i> . . . . .	133
5.2.2	<i>operating_fp.s</i> . . . . .	135
5.3	Detalles de la implementación . . . . .	138
6	Conclusiones . . . . .	141

**Bibliografía** **143**

**Anexos** **146**

1	Código fuente del <i>SoC</i> . . . . .	146
2	<i>Test Benches</i> . . . . .	228
3	Códigos generadores de <i>Test Benches</i> . . . . .	242

4	Códigos <i>Assembler</i> . . . . .	261
4.1	<i>Script</i> conversor de palabras . . . . .	267
5	Manual de uso del <i>SoC</i> . . . . .	269
5.1	Configurando el proyecto en <i>Vivado</i> . . . . .	269
5.2	Compilación de programas para el <i>SoC</i> . . . . .	278