## $\mathbf{\acute{I}ndex}$

1	Intr	Introducció							
	1.1	Descri	ipció del projecte	. 6					
		1.1.1	Risc-V (que és)	. 6					
		1.1.2	Estat del Art	. 6					
		1.1.3	Finalitat del projecte	. 6					
2	Cor	e Risc	:-V	6					
	2.1	Instru	action set	. 6					
		2.1.1	RV32I	. 6					
	2.2	Arquit	tectura	. 6					
		2.2.1	Diagrames	. 6					
	2.3	Arithr	meticologic	. 6					
		2.3.1	Funcionament	. 6					
		2.3.2	Implementació	. 6					
	2.4	Load		. 6					
		2.4.1	Funcionament	. 6					
		2.4.2	Implementació	. 6					
	2.5	Store		. 6					
		2.5.1	Funcionament	. 6					
		2.5.2	Implementació	. 6					
	2.6	Fetch	and Jumps	. 6					
		2.6.1	Funcionament	. 6					
		2.6.2	Implementació	. 6					
	2.7	CSR		. 6					
		271	Funcionament	6					

		2.7.2	Implementació	6
	2.8	Embol	lcall	6
		2.8.1	Arquitectura	6
		2.8.2	Memòria de Programa	6
		2.8.3	Memòria de Dades	6
3	Cor	e Risc	-V adaptat a memòries arbitrades	6
	3.1	Plataf	orma Pulpino	6
	3.2	Modifi	icacions necessàries	6
		3.2.1	Senyals	6
		3.2.2	Estats	6
		3.2.3	Stall Core	6
4	Ein	$_{ m es}$ ${ m Util}$	litzades	6
	4.1	Eines	de Programació	6
		4.1.1	Interpret de Verilog	6
		4.1.2	Visual Studio Code	6
		4.1.3	Docker	6
		4.1.4	RiscV Gnu Toolchain	6
		4.1.5	Git	6
		4.1.6	Continus integration	6
	4.2	Eines	de síntesi per a FPGA	6
		4.2.1	Quartus II	6
		4.2.2	Mega Wizard	6
	4.3	Eines	de validació	6
		4.3.1	Tests amb Verilog	6
ĸ	Dog	ultata	dela Teata	G

	5.1	Aritmeticològic	6							
	5.2	Branch	6							
	5.3	CSR	6							
	5.4	Load-Store	6							
6	6 Sintetitzat en FPGA									
	6.1	Embolcall FPGA	6							
		6.1.1 Memòries	6							
		6.1.2 PLL	6							
		6.1.3 Debounce	6							
		6.1.4 Display 7 segments	6							
	6.2	Programació FPGA	6							
		6.2.1 Síntesi	6							
		6.2.2 Flash	6							
	6.3	Funcionament	6							
7 Conclusions			6							
	7.1	Tests	6							
	7.2	Future Work	6							
Referències 6										