

October 27th, 2019.

MIPS REPORT

1. Explanation of the MIPS ISA.

The ISA of MIPS includes multiple instruction divided in three groups: R-Type, I-Type and J-Type. Every instruction starts with a 6 bits operation code (op) which determines what instruction need to be done by the architecture. These instructions are 32 bits long.

The R-Type instruction are the ones that only need the Register File to perform. Besides this op-code, this operations has 6 bits extra at the end to determine the function to perform. They also specified 3 registers of 5 bits each where 2 are for operations and one to stored. The las 5 bits before the function code, are reserved for shift.

The I-Type instructions can use also the Random Access Memory in order to get values for operations. These instructions have also its 6 first bit to specify which instruction is about to perform. The last 16 bits are reserved to add an immediate value whether to the RAM or the RF. It also has an space to specify 2 registers.

The J-Type instructions are used to change de Program Counter inside the MIPS. These instructions also has an opcode to determine each

instruction. In this project, I only used 16 bits for the address where the PC use.

2. A table showing your instruction encoding. This must include ALL your supported instructions.

General Table.

	Instruction	op	func
R-Type	ADD	000_000	000_000
	SUB	000_000	000_001
	AND	000_000	000_010
	OR	000_000	000_011
	XOR	000_000	000_100
	SLL	000_000	000_101
	SRL	000_000	000_110
	SLA	000_000	000_111
	SRA	000_000	001_000
	ADDI	010_000	ZZZ_ZZZ
I-Type	SUBI	010_001	ZZZ_ZZZ
	ANDI	010_010	ZZZ_ZZZ
	ORI	010_011	ZZZ_ZZZ
	XORI	010_100	ZZZ_ZZZ
	LUI	001_110	ZZZ_ZZZ
	LLI	001_111	ZZZ_ZZZ
	LWR	100_000	ZZZ_ZZZ
	SWR	101_000	ZZZ_ZZZ
	LWI	100_001	ZZZ_ZZZ
	SWI	101_001	ZZZ_ZZZ
	BEQ	110_000	ZZZ_ZZZ
	BNE	110_001	ZZZ_ZZZ
J-Type	JUMP	111_000	ZZZ_ZZZ
	JAL	111_001	ZZZ_ZZZ
	RET	111_010	ZZZ_ZZZ

(A more specified table is an Excel named: MIPS which is attached)

3. An explanation of your design choices and design flow.

This design is based on the one that we received for the R-Types only, but now is adapted to perform all the instructions mentioned in General Table. I decided to add multiple multiplexors in order to perform all the instructions needed. The RAM and ROM (Instruction Memory) are not inside the MIPS.

4. A computer-drawn schematic diagram showing your final Microarchitecture (μA) design for your MIPS core.

The only extra thing that I add to this draw was a Flip Flop at the end of the module of Next PC and the next_pc is linked to PC inside MIPS, which means MIPS does not have an input PC.



