

Conclusion

1 min

Congratulations on making it through the lesson on Instruction Set Architecture. We have covered a lot of ground so far so let's review some of the key concepts:

- The Instruction Set Architecture is the set of rules that define the communication of the software and the hardware.
- Computer hardware requires instructions that are presented in a specific way every time.
- The

Preview: Docs Loading link description

[CPU](#)

is composed of three primary components:

- Control Unit
- Arithmetic and Logic Unit
- Registers
- Two primary ISA designs have emerged in the computing world, CISC and RISC, each with its own advantages and disadvantages.
- ISAs are specific to each computer. An x86 Intel CISC machine cannot run the code that was produced for an ARM RISC machine.
- Working with computers at the

Preview: Docs Loading link description

[binary](#)

, or

Preview: Docs Loading link description

[machine code](#)

level requires extremely precise inputs, with the help of the ISA, compilers and higher-level programming languages abstract away much of the tedious work.

In our upcoming project, we will build on the knowledge of MIPS-specific instructions we learned here to replicate a simple calculator circuit in our hardware. We will create four basic arithmetic functions that will simulate the inputs and outputs of the Arithmetic and Logic Unit (ALU).

Instructions

Head over to the project where we will use MIPS to build a simple calculator!

