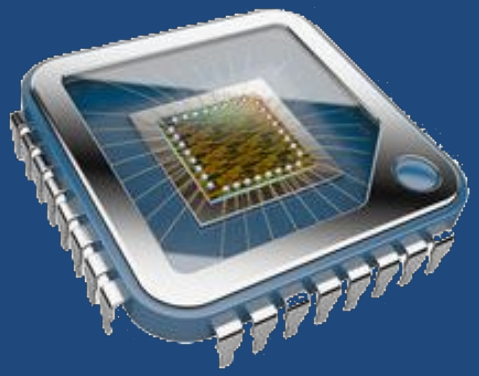


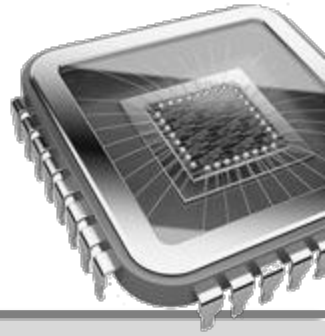
ORGANIZAÇÃO E ARQUITETURA DE COMPUTADORES
AULA 20

Prof. Max Santana Rolemberg Farias
max.santana@univasf.edu.br
Colegiado de Engenharia de Computação

GESE
Grupo de Engenharia de Sistemas Embarcados

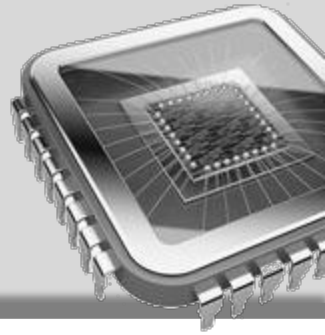
CEComp
Engenharia da Computação

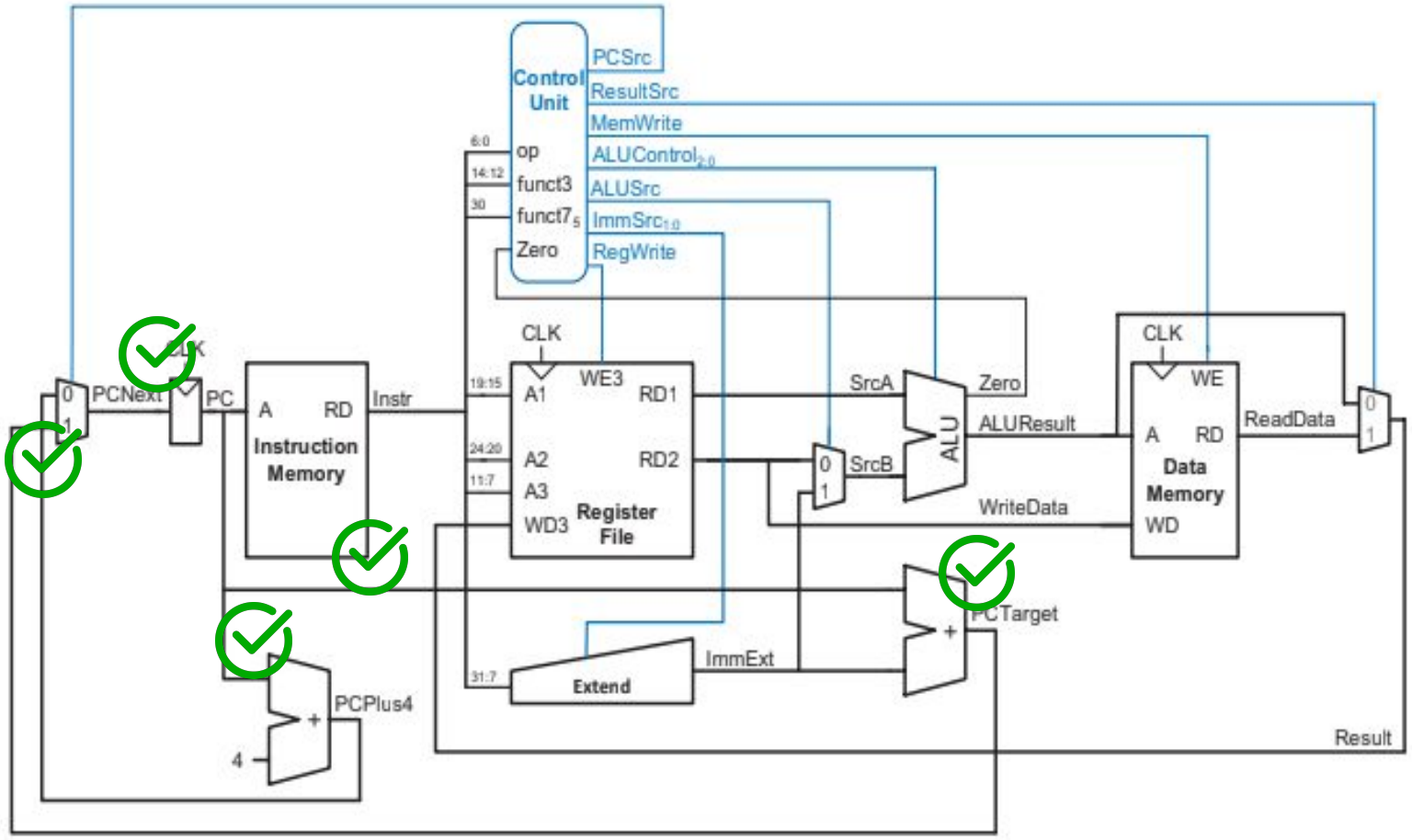
UNIVASF
UNIVERSIDADE FEDERAL DO VALE DO SÃO FRANCISCO



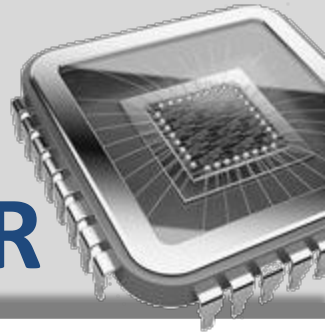
PROCESSADOR DIDÁTICO
myRISCVv1

PROCESSADOR myRISCVv1
DATAPATH





PROCESSADOR myRISCVv1
DATAPATH INSTRUCTION DECODER

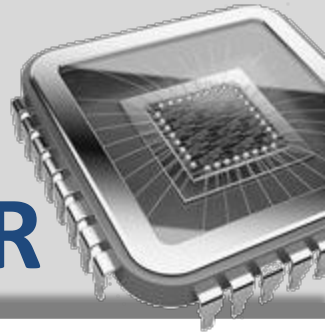


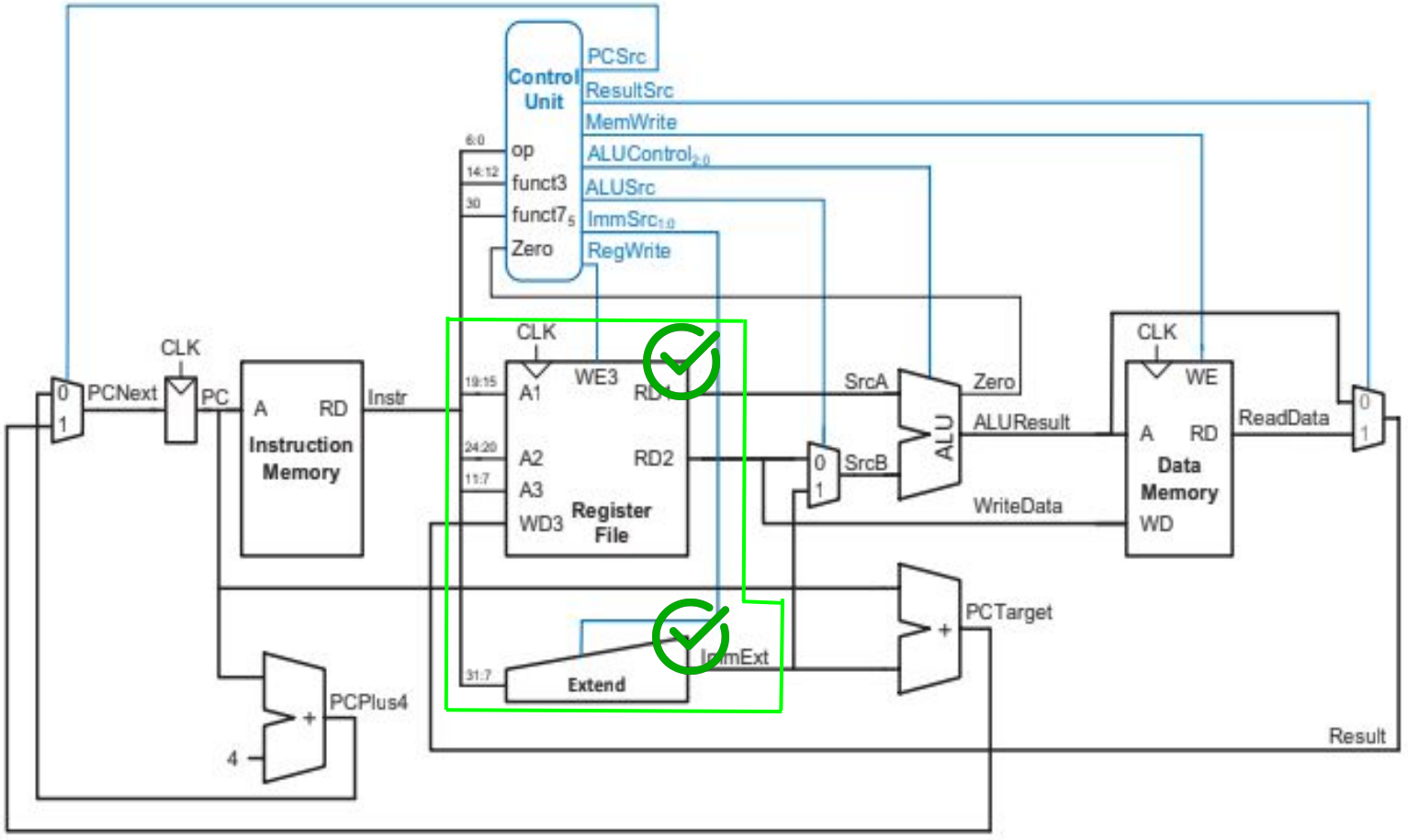
Componente fundamental que desempenha um papel crítico na interpretação das instruções.

- Traduz a representação binária das instruções em sinais de controle significativos e dados.

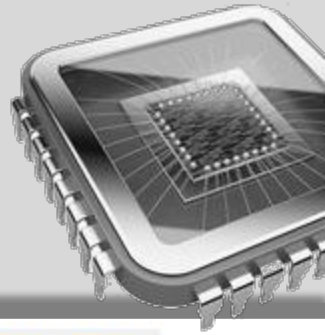
- Garante que o processador entenda qual operação executar, quais operandos usar e como manipular dados com base no formato de instrução

PROCESSADOR myRISCVv1
DATAPATH INSTRUCTION DECODER





PROCESSADOR myRISCVv1
DATAPATH IDECODER



```
----------------------------------------------------------------------------------
-- Instruction decode                                                           --
-- myRISCVv1                                                                    --
--                                                                              --
-- Prof. Max Santana  (2025)                                                    --
-- CEComp/Univasf                                                               --
----------------------------------------------------------------------------------

library ieee;
use ieee.std_logic_1164.all;

entity idecoder is
  port(
    clk   : in std_logic;
    rst   : in std_logic;
    instr : in std_logic_vector(31 downto 0);  -- instruction
    we    : in std_logic;                      -- control signal (regwrite)
    immSrc: in std_logic_vector(1 downto 0);   -- control signal (immSrcs)
    wd    : in std_logic_vector(31 downto 0);  -- write data
    op    : out std_logic_vector(6 downto 0);  -- opcode
    funct3: out std_logic_vector(2 downto 0);   -- funct3
    funct7: out std_logic_vector(6 downto 0);   -- funct7
    rd1   : out std_logic_vector(31 downto 0); -- Read data 1
    rd2   : out std_logic_vector(31 downto 0); -- Read data 2
    immExt: out std_logic_vector(31 downto 0); -- Immediate extend
  );
end idecoder;
```