library ieee;

use ieee.std\_logic\_1164.all;

entity ULA1 is

port (

a, b, less, binvert, carryin, operation: in std\_logic;

resultado, set, overflow: out std\_logic;

);

end ULA1;

architecture ULA1\_arch of ULA1 is

component somador is

port (

a, b, c : in std\_logic;

soma, carry: out std\_logic;

);

end somador;

component mult2x1 is

port (

val, binvert : in std\_logic;

s : out std\_logic

);

end mult2x1;

component mult4x1 is

port (

r0, r1, r2, r3, op: in std\_logic;

s: out std\_logic

);

end mult2x1;

signal resultSomador: std\_logic;

signal resultAnd: std\_logic;

signal resultOr: std\_logic;

signal resultBInvert: std\_logic;

begin

resultAnd <= a and b;

resultOr <= a or b;

mux2: mult2x1

port map(

val => b;

binvert => binvert

s => resultBInvert

);

somador1b: somador

port map (

a => a;

b => resultBInvert;

c => carryin;

soma => resultSomador;

carry => cout

);

mux4: mult4x1

port map (

r0 => resultAnd,

r1 => resultOr,

r2 => resultSomador,

r3 => less,

op => operation;

s => resultado;

);

end ULA1\_arch;