

Carrera de Especialización en Sistemas Embebidos Trabajo práctico final Microarquitecturas y Softcores

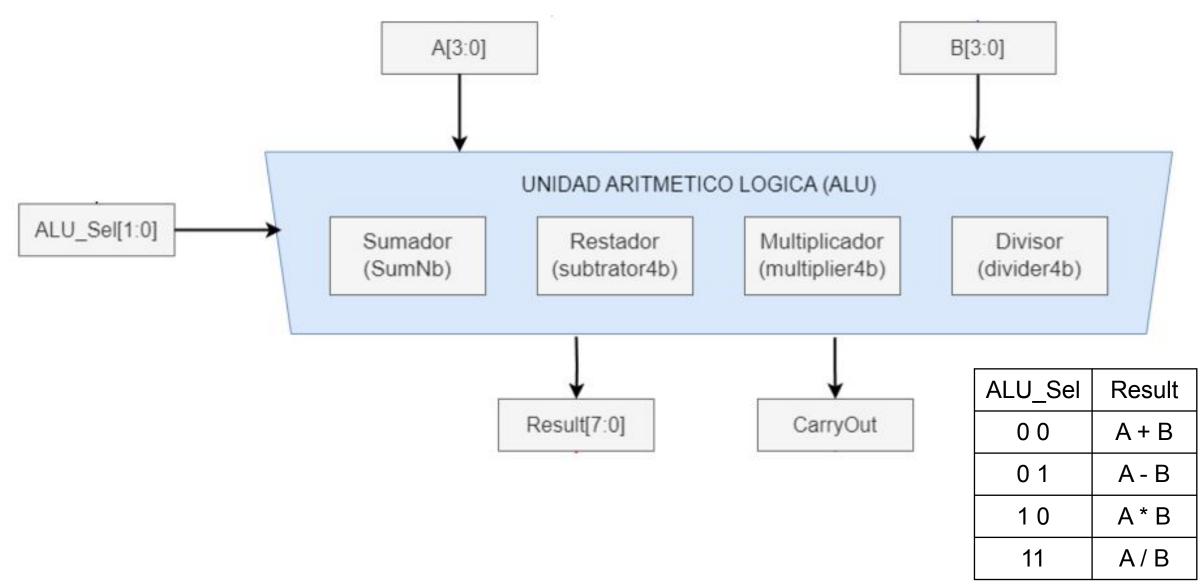
#### IP core ALU

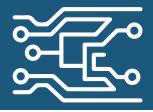
Autor: Ing. Jorge Fernando Vásquez González Profesor: Ing. Nicolás Álvarez (FIUBA)

### ALU de 4 bits

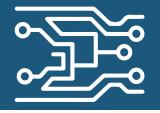
646



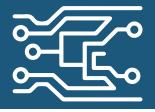




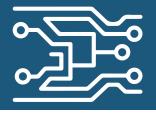
## Test bench ALU de 4 bits

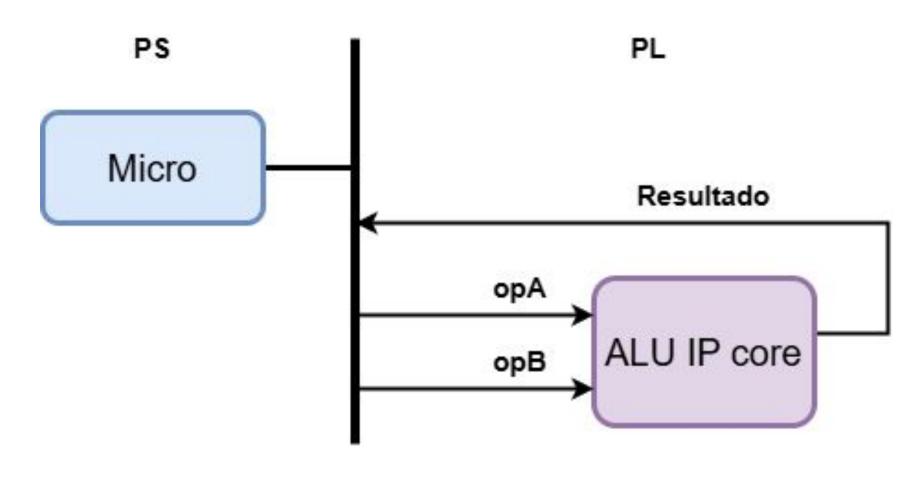






## Ip core ALU

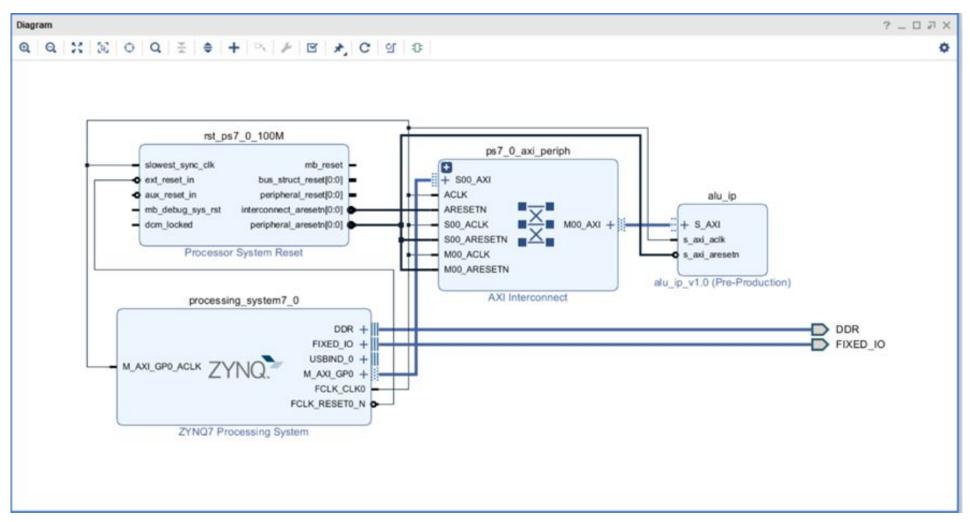




PS y PL del sistema a implementar.







Sistema que incluye el ip core personalizado.





```
Sources
     ₹ | ♦ | + | ? | ● 0
Design Sources (1)

✓ ●∴ proyecto_ALU_wrapper(STRUCTURE) (proyecto_ALU_wrapper.vhd) (1)

       proyecto_ALU_i : proyecto_ALU (proyecto_ALU.bd) (1)

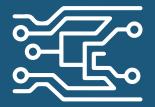
✓ ● proyecto_ALU(STRUCTURE) (proyecto_ALU.vhd) (5)

¬ ♀ ■ alu ip: proyecto ALU alu ip 0 0 (proyecto ALU alu ip 0 0.xci) (1)

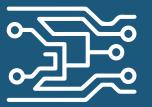
✓ ■ proyecto ALU alu ip 0 0(proyecto ALU alu ip 0 0 arch) (proyecto ALU alu ip 0 0.vhd) (1)

∨ ● U0: alu_ip_v1_0(arch_imp) (alu_ip_v1_0.vhd) (1)
                        alu_ip_v1_0_S_AXI_inst: alu_ip_v1_0_S_AXI(arch_imp) (alu_ip_v1_0_S_AXI.vhd) (1)
                            alu_inst: ALU(ALU_arq) (ALU.vhd) (4)
                                 sumador_inst: sumNb(sumNb_arg) (SumNb.vhd) (4)
                                    subtractor_inst: subtractor4b(subtractor4b_arq) (subtractor4b.vhd)
                                   multiplier_inst: multiplier4b(multiplier4b_arg) (multiplier4b.vhd)
                                  divider_inst: divider4b(divider4b_arq) (divider4b.vhd)
              > Pm processing_system7_0: proyecto_ALU_processing_system7_0_0 (proyecto_ALU_processing_system7_0_0.xci)
              ps7_0_axi_periph: proyecto_ALU_ps7_0_axi_periph_0(STRUCTURE) (proyecto_ALU.vhd) (1)
                ps7_0_axi_periph: proyecto_ALU_ps7_0_axi_periph_0
              > Per rst_ps7_0_100M: proyecto_ALU_rst_ps7_0_100M_0 (proyecto_ALU_rst_ps7_0_100M_0.xci)
```

Archivo Wrapper HDL generado y agregado al proyecto.



# DEMOSTRACIÓN



```
artyz7-user00@lse-server-pc:~
Welcome to minicom 2.8
OPTIONS: I18n
Compiled on Jul 20 2023, 00:00:00.
Port /dev/ArtyZ7-Board01-003017A4C8AB, 01:40:32
Press CTRL-A Z for help on special keys
 -- Este programa valida el funcionamiento de una ALU de 4 bits con un IP cores propios --
-- suma --
Operacion: 6 + 2 = 8
-- resta --
Operacion: 6 - 2 = 4
-- multiplicacion --
Operacion: 6 * 2 = 12
 - division --
Operacion: 6 / 2 = 3
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.8 | VT102 | Offline | ArtyZ7-Board01-003017A4C8AB
```