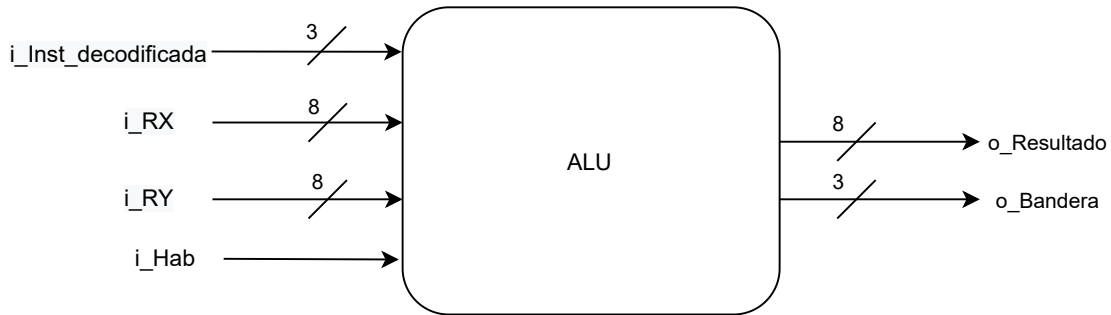


Unidad Lógica Aritmética (ALU)



Entradas y salidas

| Señal | Núm. de Bits | Descripción |
|---------------------|--------------|--|
| i_Inst_decodificada | 3 bits | Entrada que le indica a la unidad qué operación matemática o lógica realizará. |
| i_RX | 8 bits | Entrada desde el banco de registros, contiene el operando 1 que se procesará. |
| i_RY | 8 bits | Entrada desde el banco de registros, contiene el operando 2 que se procesará. |
| i_Hab | 1 bit | Habilita a la ALU para realizar una operación. |
| o_Resultados | 8 bits | Salida que contiene los datos post-procesamiento, a los cuales ya se les aplicó la operación deseada. |
| o_Bandera | 3 bits | Salida dirigida a la unidad de control para señalar el estado de la operación procesada en la ALU, para ejecutar una acción con base a ello. |

Descripción funcional

Se encarga de realizar las operaciones matemáticas y lógicas, se sirve del acumulador y del bloque de banderas Flags para enviar los datos de las operaciones a la unidad de control y posteriormente poder almacenar los resultados en los registros.

Se encarga de establecer banderas o avisos que le servirán a la unidad de control para evaluar el estado de la operación, las banderas que contiene pueden ser de acarreo, de cero, de desbordamiento y de signo. Recibe una copia del resultado de una operación desde la ALU y le asigna un estado o bandera.

°Pseudocódigo

Algoritmo ALU

//Entradas y salidas

Dimension i_Inst_decodificada[3]

Dimension i_RX[8]

Dimension i_RY[8]

Definir i_Hab como entero

Dimension o_Resultado[8]

Dimension o_Bandera[3]

//Señales internas

```

Dimension resultado[16]
Definir Z Como Entero
Definir C Como Entero
Definir N Como Entero

//Bloque always
Si i_Hab=1 Entonces
Segun i_Inst_decodificada Hacer
000:
resultado[]<-i_RX[]+i_RY[]
001:
resultado[]<-i_RX[]-i_RY[]
010:
resultado[]<-i_RX[]<<i_RY[]
011:
resultado[]<-i_RX[]>>i_RY[]
100:
resultado[]<-i_RX[]
101:
resultado[]<-i_RX[] Y i_RY[]
110:
resultado[]<-i_RX[] O i_RY[]
111:
resultado[]<-i_RX[]^i_RY[]
De Otro Modo:
resultado[]<-[00000000]
Fin Segun
Si resultado[]=0 Entonces
Z<-1
SiNo
Z<-0
Fin Si
Si ([11111111]-resultado[])<0 Entonces
C<-1
SiNo
C<-0
Fin Si
Si resultado[]<0 Entonces
N<-1
SiNo
N<-0
Fin Si
Fin Si
o_Resultado[]<-resultado[7:0]
o_Bandera[0]<-Z
o_Bandera[1]<-C
o_Bandera[2]<-N
FinAlgoritmo

```