

# Diseño y realización de un procesador avanzado

Arquitectura de computadores

Iván Martín y Asier Gutiérrez

# Índice de Contenidos

<b>Índice de Contenidos</b>	<b>1</b>
<b>Definición de las instrucciones</b>	<b>2</b>
<b>Distribución de las instrucciones</b>	<b>3</b>
<b>Diseño de la ALU</b>	<b>4</b>
<b>Diseño de la Unidad de Control</b>	<b>7</b>
<b>Diseño de la Unidad de Proceso</b>	<b>9</b>
<b>Bibliografía</b>	<b>10</b>

## Definición de las instrucciones

La tarea de un procesador es la de procesar instrucciones y dar resultados, así pues, el diseño de un procesador varía en cuanto a las instrucciones que tenga que poder ejecutar.

Esta tabla muestra las instrucciones que el procesador debe poder ejecutar además de su descripción y los campos los cuales actúan en ellas.

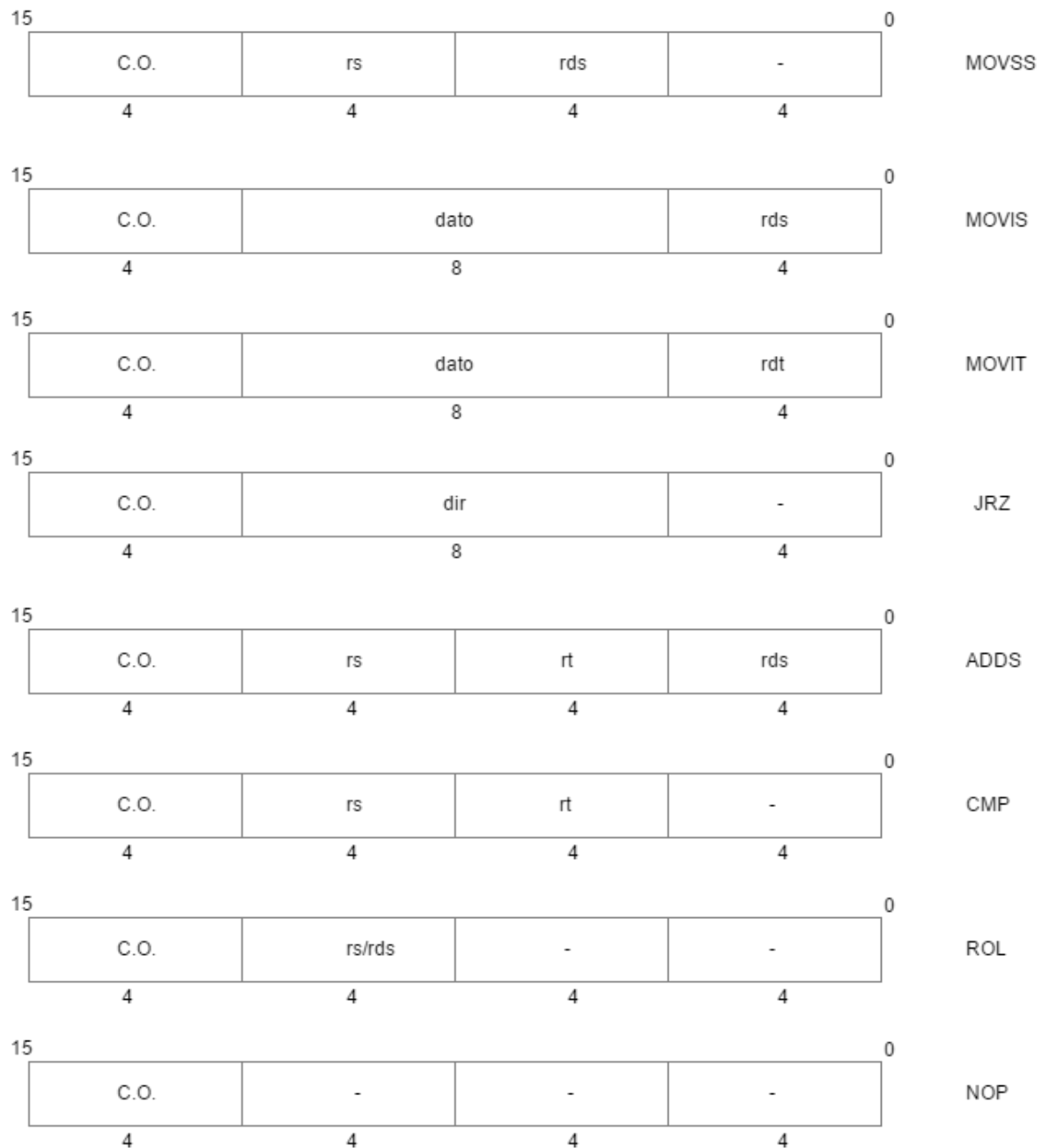
Número de instrucción	Instrucción	Campos	Descripción
1	MOVSS	rs, rds	$rs \rightarrow rds$
2	MOVIS	dato, rds	$dato \rightarrow rds$
3	MOVIT	dato, rdt	$dato \rightarrow rdt$
4	JRZ	dir	saltar a dir si $Z=1$
5	ADDS	rs, rt, rds	$rs + rt \rightarrow rds$ [Carry]
6	CMP	rs	$rs - rd \Rightarrow \text{Flag } Z$
7	ROL	rs	Rota una posición a la izquierda y deposita en rds
8	NOP	-	Depositar en rds el mismo valor de rs

## Distribución de las instrucciones

Una vez se conocen las instrucciones y lo que necesitan para que se puedan ejecutar, es momento de organizarlas en memoria. Esta organización determina cómo se almacenarán las instrucciones y como se cargarán de dicha memoria.

La clave de esta organización es reducir el número de multiplexores necesarios. Así pues, es por eso que el código de operación se posiciona en el mismo lugar en todas las instrucciones.

Este es el mapa de distribución de instrucciones que hemos elaborado. Los números que aparecen aquí están reflejados en bits.

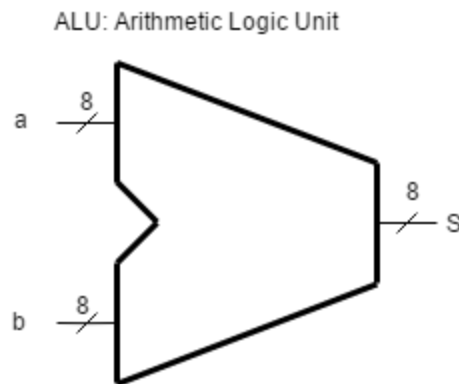


## Diseño de la ALU

La ALU es el componente el cual procesa todas las operaciones lógico-aritméticas. Al igual que el procesador, la ALU también varía en función a las operaciones derivadas de las instrucciones que debe ejecutar.

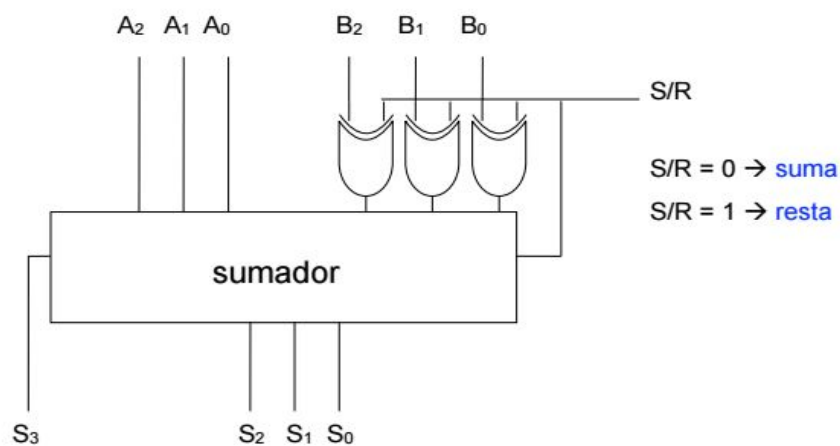
Las operaciones de ALU serán las siguientes: suma, resta y rotación. Esto es debido a que la instrucción ADDS necesita de una suma, la instrucción CMP necesita de una resta y la operación ROL necesita de una rotación.

La ALU, de manera externa, se simbolizará de la siguiente manera:

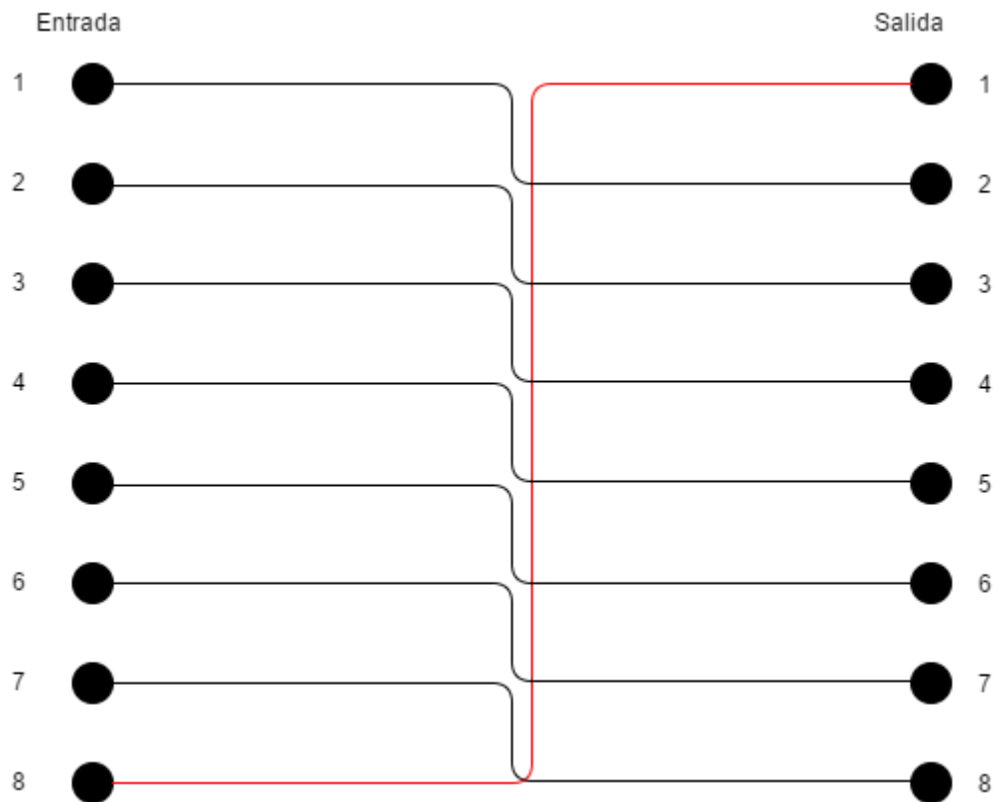


Para la suma se utilizará un sumador, para la resta se utilizará el mismo sumador con un ajuste para hacerlo complemento a dos y para la rotación se seguirá un esquema de cableado que se muestra a continuación puesto que se ha desestimado el uso de un componente.

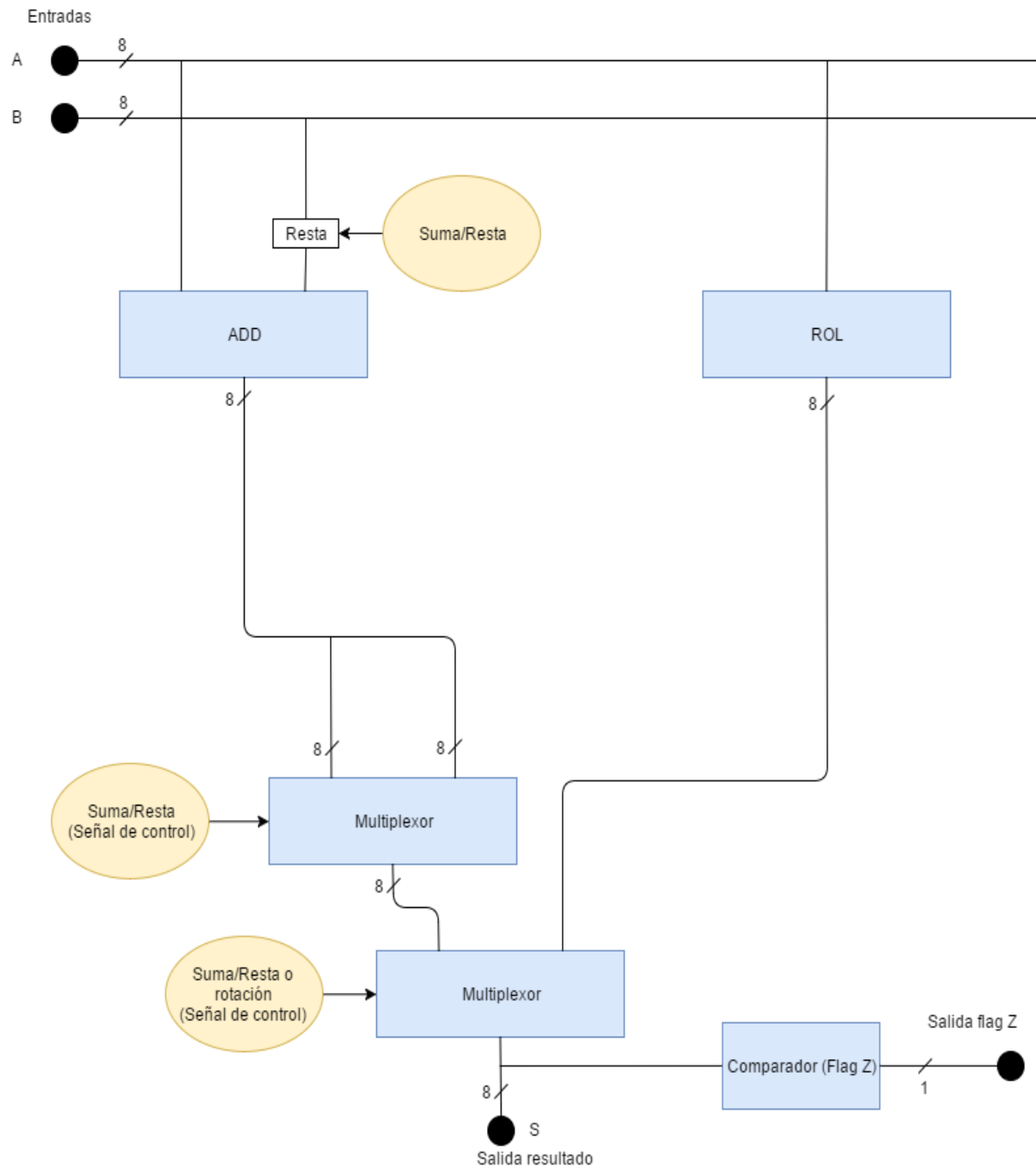
La idea del sumador y restador complemento a dos es igual que este solo que contiene más entradas y salidas (dos entradas de ocho bits y una salida de ocho bits) y solo una señal (no llega a entrar en el sumador):



El siguiente esquema de cableado muestra la implementación de la instrucción que supone una rotación:



La implementación de estas operaciones es la siguiente:





## Diseño de la Unidad de Control

Cuando varios datos deben entrar a los registros u a otro componente, se debe multiplexar. El código de operación es en la gran mayoría de veces quién dictamina qué dato debería entrar a dicho componente. Por ello, la unidad de control es la encargada de generar las señales dado un determinado código de operación.

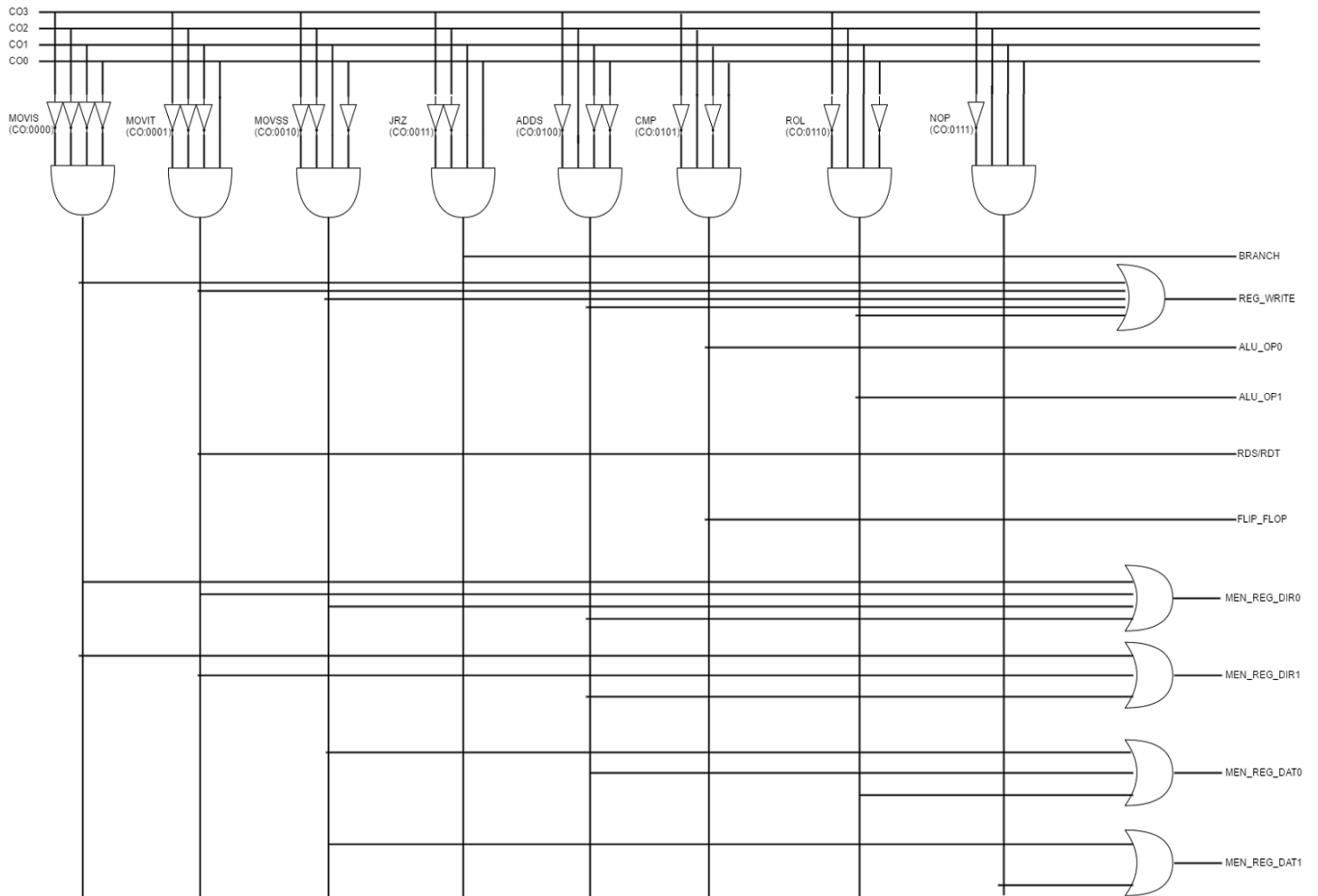
Las señales de control, dependen del conjunto global de las instrucciones, pues hay instrucciones, por ejemplo cuyo dato entra de un distinto sitio a un mismo registro. Se deben por tanto hallar dichas señales.

No obstante, hay señales que también surgen de un componente. Es el ejemplo de los registros, depende de si se quiere leer o escribir se necesitará un cero o un uno.

Estas son las señales de control necesarias en este caso:

INST	CO	BRANCH	REG_WRITE	ALU_OP	RDS/RDT	MEM_REG_DIR	MEM_REG_DATA	FLIP_FLOP
MOVIS	0000	0	1	XX	0	11	00	0
MOVIT	0001	0	1	XX	1	11	00	0
MOVSS	0010	0	1	XX	0	01	11	0
JRZ	0011	1	0	XX	X	XX	XX	0
ADDS	0100	0	1	00	0	11	01	0
CMP	0101	0	0	01	X	XX	XX	1
ROL	0110	0	1	10	0	00	01	0
NOP	0111	0	X	XX	0	00	10	0

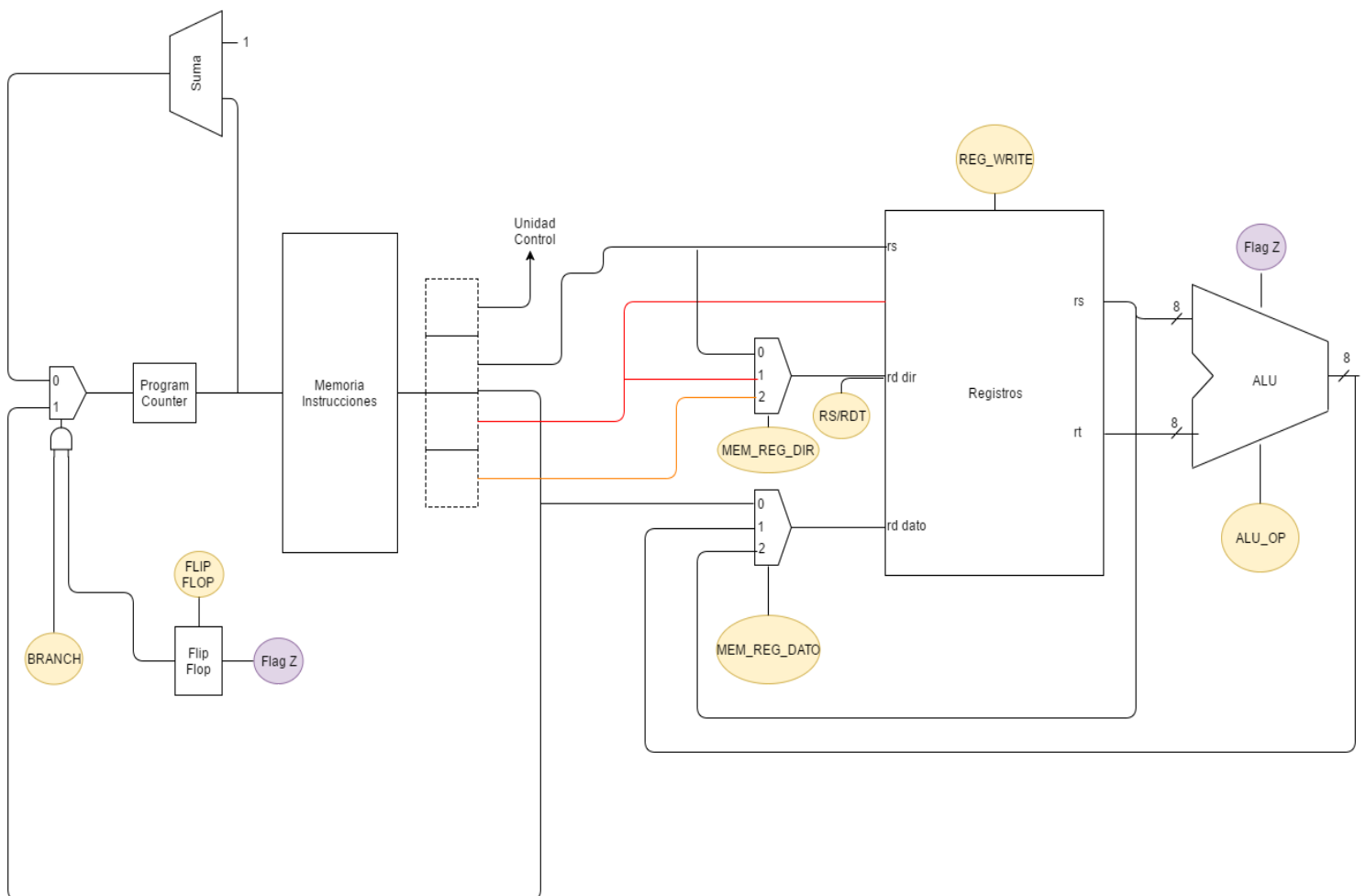
La implementación de esta tabla es un circuito combinacional sencillo. Este es el circuito resultante:



## Diseño de la Unidad de Proceso

La unidad de proceso es la conjunción de todo lo diseñado previamente junto con unos componentes que por defecto se deben añadir, como son el contador de programa, la suma al contador del programa y la memoria de instrucciones. Los registros, también se da por hecho que se deben añadir debido al set de instrucciones.

Diseño del procesador:



## Bibliografía

- Fundamentos y Estructura de Computadores (2003) - José María Angulo Usategui, Javier García Zubia e Ignacio Angulo Martinez.
- <http://www.esi.uclm.es/www/isanchez/teco0910/sr.pdf> Sumador restador complemento a dos.