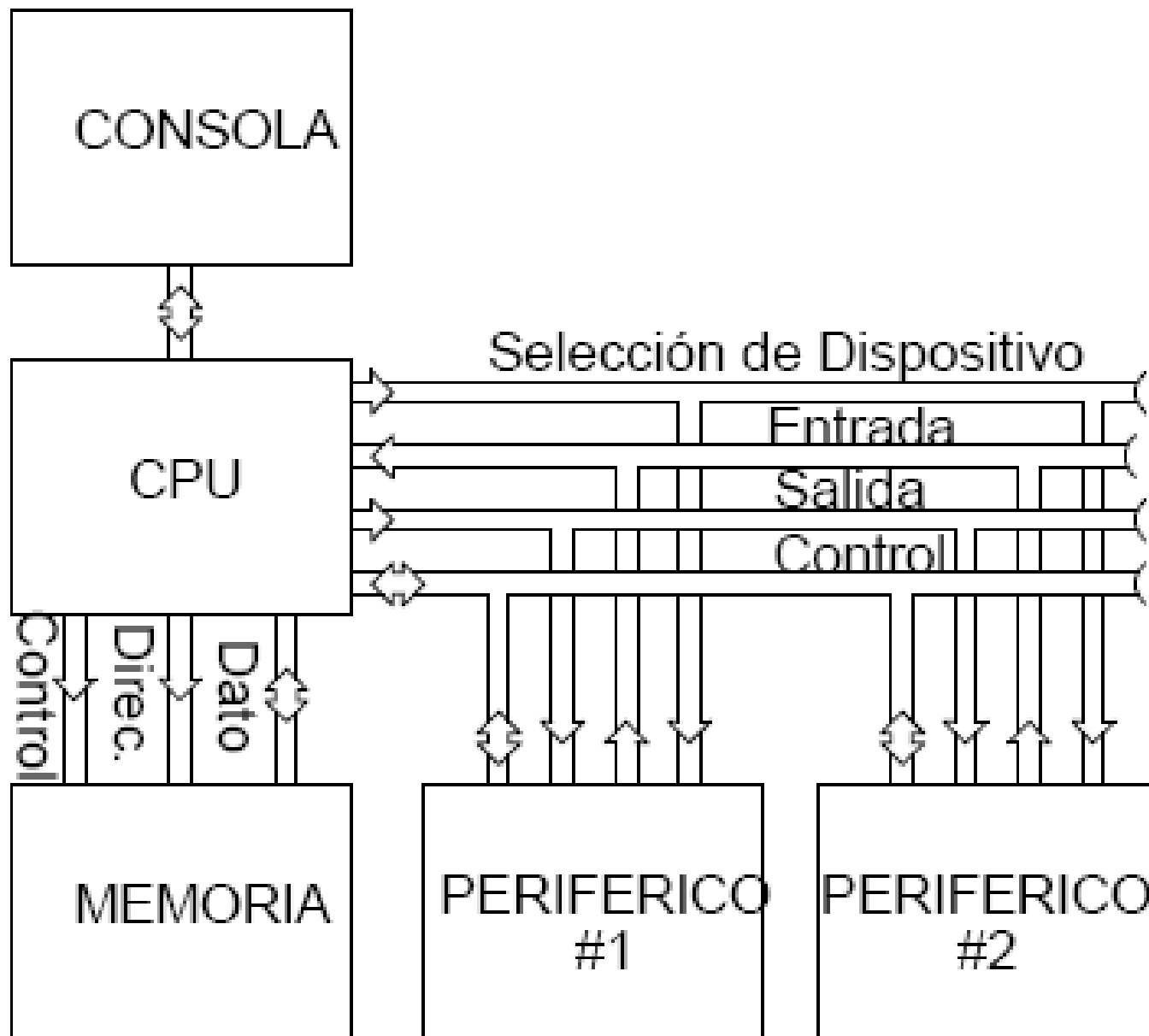


ARQUITECTURA DE COMPUTADORAS

UNIDAD TEMATICA N° 4

Blue Computer





Palabra de Datos



Palabra de Instrucción

Fig. 1: Formato de las palabras de datos e instrucciones.

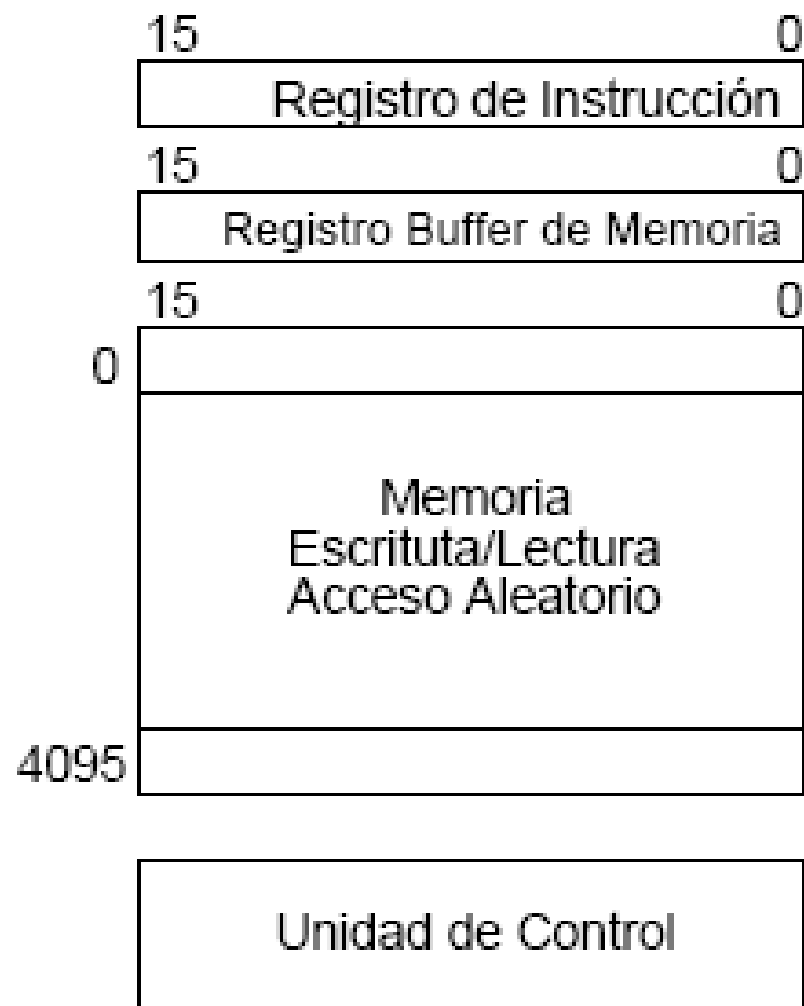


Fig. 2: Una vista general de la BLUE.

Código Máquina	Mnemónico	Operando	Descripción	
00	HLT	XXXX	Esta instrucción hace que la computadora se detenga. Apretar el botón START en la consola hará que la máquina arranque de nuevo; comenzando con la instrucción siguiente. El campo del dirección XXXX es ignorado.	
01	ADD	XXXX	Los contenidos de XXXX se agregan a los contenidos del acumulador, y la suma se introduce en el acumulador. Si la suma resultante es mayor que $2^{15}-1$ o menor que -2^{15} la máquina se detiene. El contenido de XXXX no se cambia ni se altera.	SUMA ARITMÉTICA $AC \leftarrow (XXXX) + (AC)$
02	XOR	XXXX	El contenido de la dirección XXXX, efectúa la máquina una OR exclusiva bit a bit entre el contenido de XXXX y el contenido del acumulador, el resultado de la operación reemplaza a los contenidos anteriores del acumulador. El contenido de XXXX no se altera	O EXCLUSIVA LÓGICA $AC \leftarrow (XXXX) \oplus (AC)$
03	AND	XXXX	El computador realiza una operación AND bit a bit entre los contenidos de la dirección XXXX y los contenidos del acumulador. El resultado de la operación es puesto en el acumulador y los contenidos de la dirección XXXX no son alterados.	Y LÓGICA $AC \leftarrow (XXXX) \cdot (AC)$

04	IOR	XXXX	Se realiza una operación OR entre los contenidos de la dirección XXXX y los contenidos del acumulador y el resultado se guarda en el acumulador, los contenidos de XXXX no son cambiados.	O LÓGICA $AC \leftarrow (XXXX) + (AC)$
05	NOT	XXXX	Cada bit del contenido del acumulador es reemplazado por su complemento lógico. El dominio de la dirección XXXX es ignorado.	NO LÓGICA $AC \leftarrow \text{NOT } (AC)$
06	LDA	XXXX	Los contenidos de XXXX se copian en el acumulador. Los anteriores contenidos del mismo se pierden. Los contenidos de XXXX no se alteran.	CARGAR ACUMULADOR $AC \leftarrow (XXXX)$
07	STA	XXXX	Los contenidos del acumulador se copian en XXXX. El anterior contenido de XXXX se pierde. Los contenidos del acumulador no se alteran.	ALMACENAR ACUMULADOR $XXXX \leftarrow (AC)$

10	SRJ	XXXX	Los contenidos del contador de programa (esto es la instrucción actual más uno) se copian en los últimos 12 bits del acumulador y los 4 bits restantes se ponen a cero luego el número XXXX se copia en el contador de programa de modo que la próxima instrucción será tomada desde la ubicación XXXX. Esta instrucción se usa para saltar a una subrutina.	SALTO A SUBROUTINA $AC<11..0> \leftarrow (PC) + 1$ $AC<15..12> \leftarrow 0$ $PC \leftarrow XXXX$
11	JMA	XXXX	Si el bit de signo del acumulador es 1 (acumulador negativo) el número XXXX es ubicado en el contador de programa, y la siguiente instrucción se toma desde la ubicación XXXX. Si el bit de signo del acumulador es 0 (acumulador cero o positivo), esta instrucción no hace nada y la próxima a ser ejecutada se toma de la ubicación actual más uno.	SALTO CONDICIONAL Si $AC<15> = 1 \therefore PC \leftarrow XXXX$ Si $AC<15> \neq 1 \therefore PC \leftarrow (PC+1)$
12	JMP	XXXX	El número XXXX se copia en el contador de programa. La siguiente instrucción es por lo tanto tomada siempre desde la ubicación XXXX.	SALTO INCONDICIONAL $PC \leftarrow XXXX$

13	INP	XXYY	Los 8 bits superiores del acumulador se borran a 0 y los caracteres de los 8 bits restantes que provienen del dispositivo de entrada YY se ingresan en la parte inferior del acumulador. La parte XX del dominio de la dirección es ignorada. La siguiente instrucción no se ejecuta hasta que la transferencia de datos se completa.	ENTRADA DE DATOS DESDE PERIFERICO $AC<15..8> \leftarrow 0$ $AC<7..0> \leftarrow (YY)$
14	OUT	XXYY	Los 8 bits más significativos del acumulador se envían al depósito de salida YY. La parte XX del dominio de la dirección es ignorado. Si el dispositivo de salida no puede aceptar los datos en ese momento, la máquina espera hasta que los datos sean aceptados antes de comenzar la ejecución de la siguiente instrucción.	SALIDA DE DATOS HACIA PERIFERICO $(YY) \leftarrow AC<15..8>$
15	RAL	XXXX	Los contenidos del acumulador son rotados un lugar a la izquierda. El bit cambiado de AC_{15} se ingresa dentro de AC_0 , de modo que el cambio es cíclico. El dominio de la dirección es ignorado.	ROTACION A LA IZQUIERDA
16	CSA	XXXX	El juego de números dispuesto en el registro de llaves de la consola reemplaza al contenido del acumulador. El dominio de la dirección XXXX es ignorado.	COPIAR LLAVES EN EL ACUMULADOR $AC \leftarrow (SR)$
17	NOP	XXXX	Esta instrucción no tiene ningún efecto. El dominio de la dirección XXXX es ignorado.	NO OPERACION

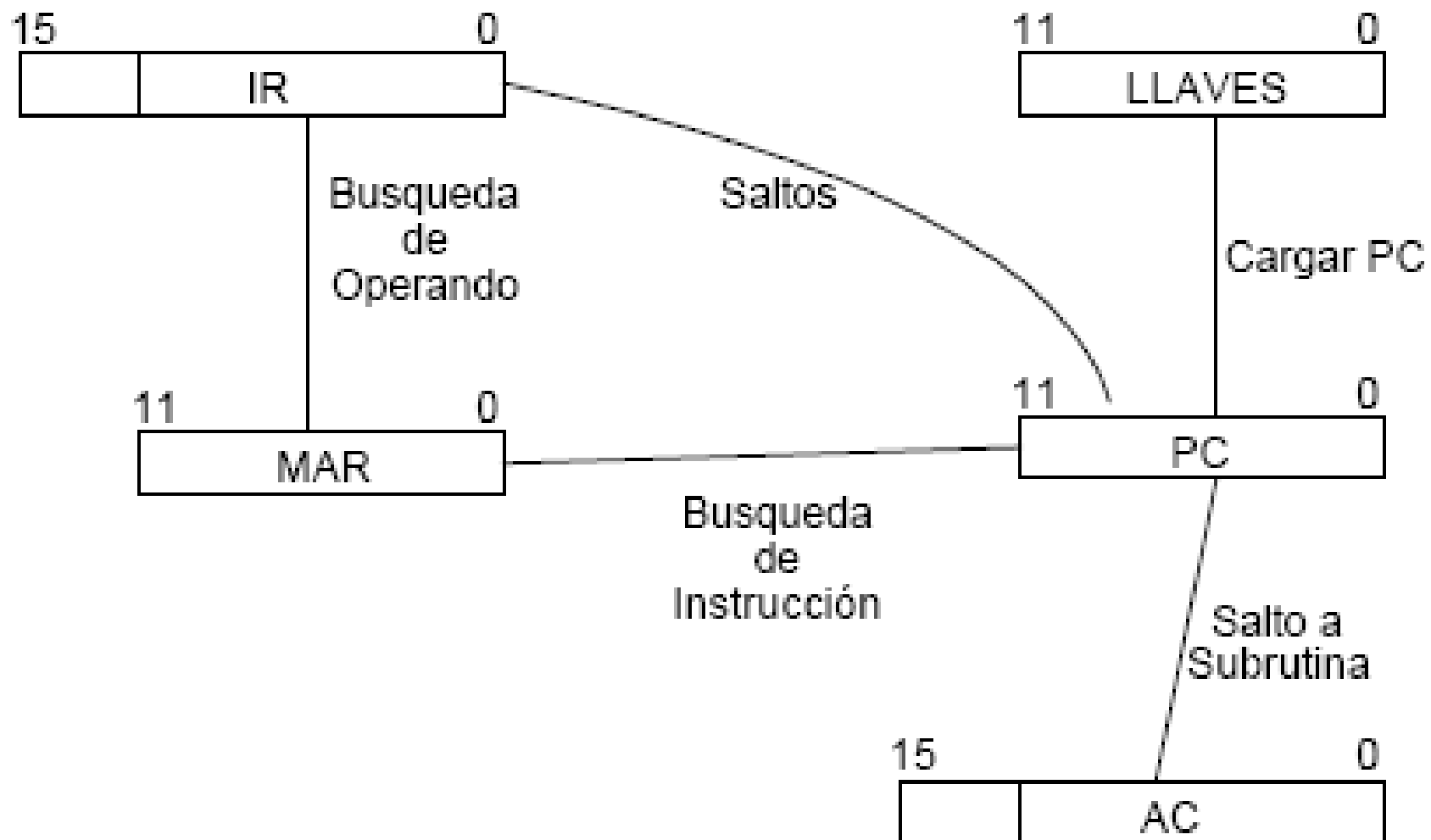


Fig.3: La transferencia de direcciones

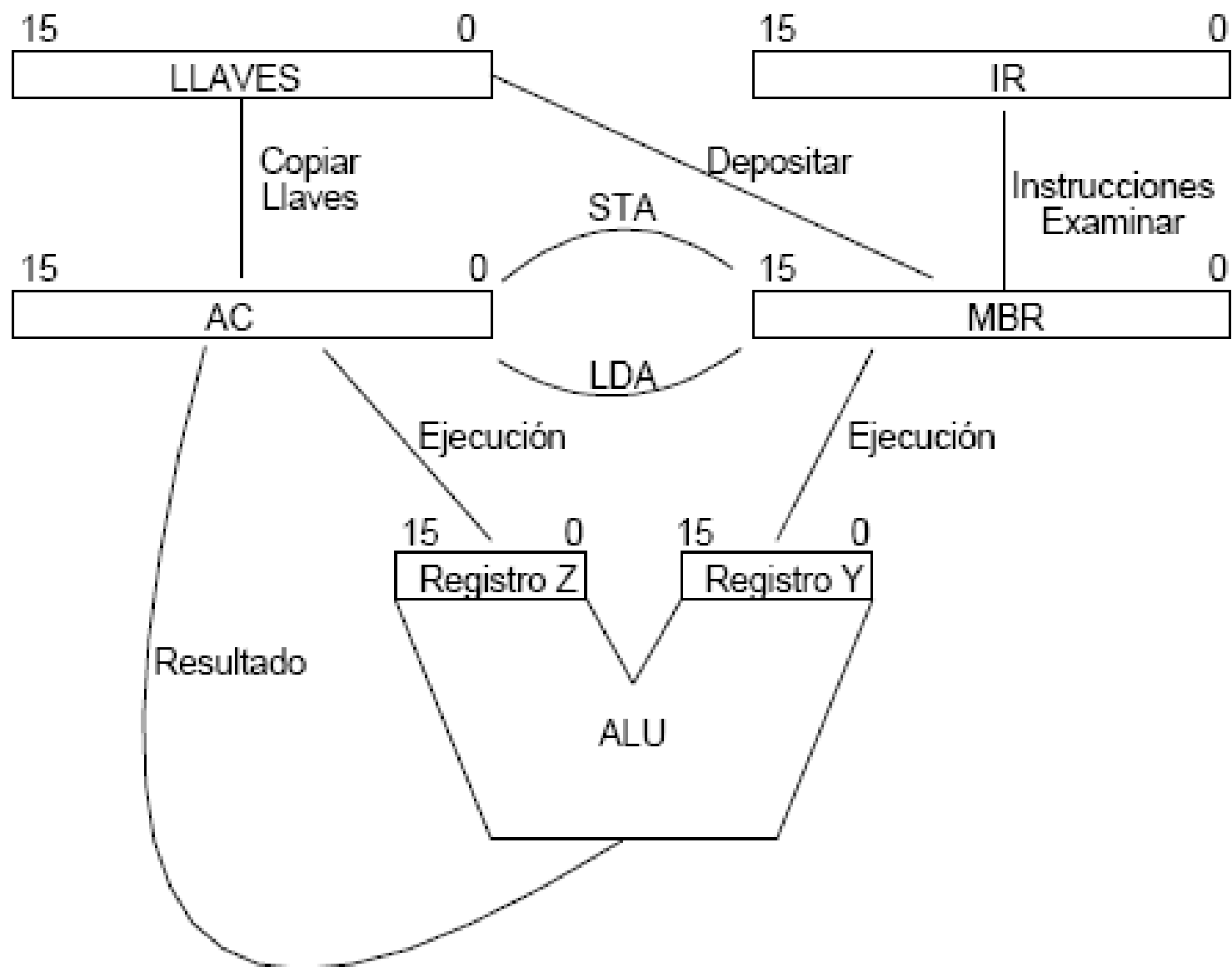


Fig.4: La transmisión de sentencias y operandos

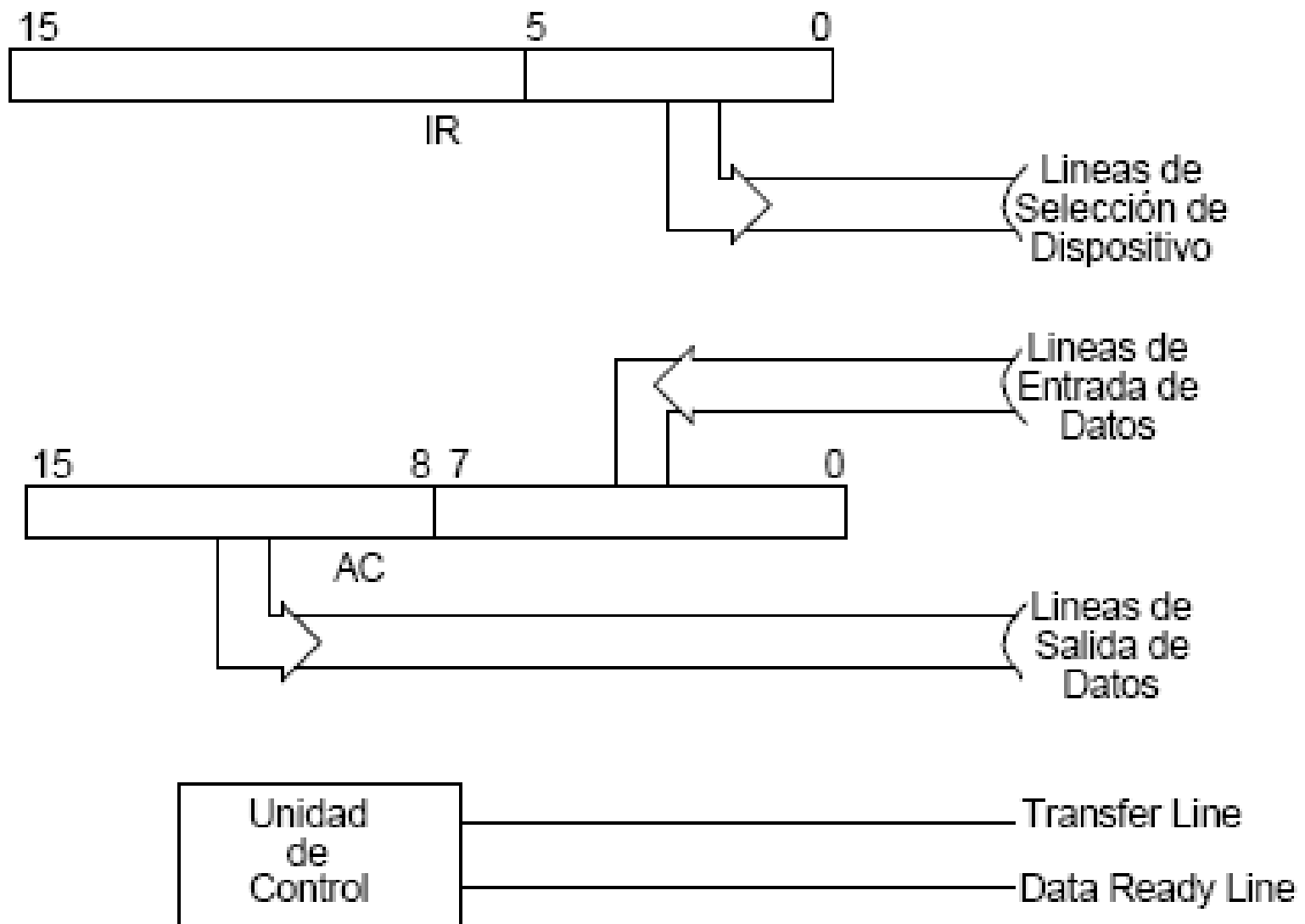


Fig.5: Las líneas de entrada-salida y selección y control

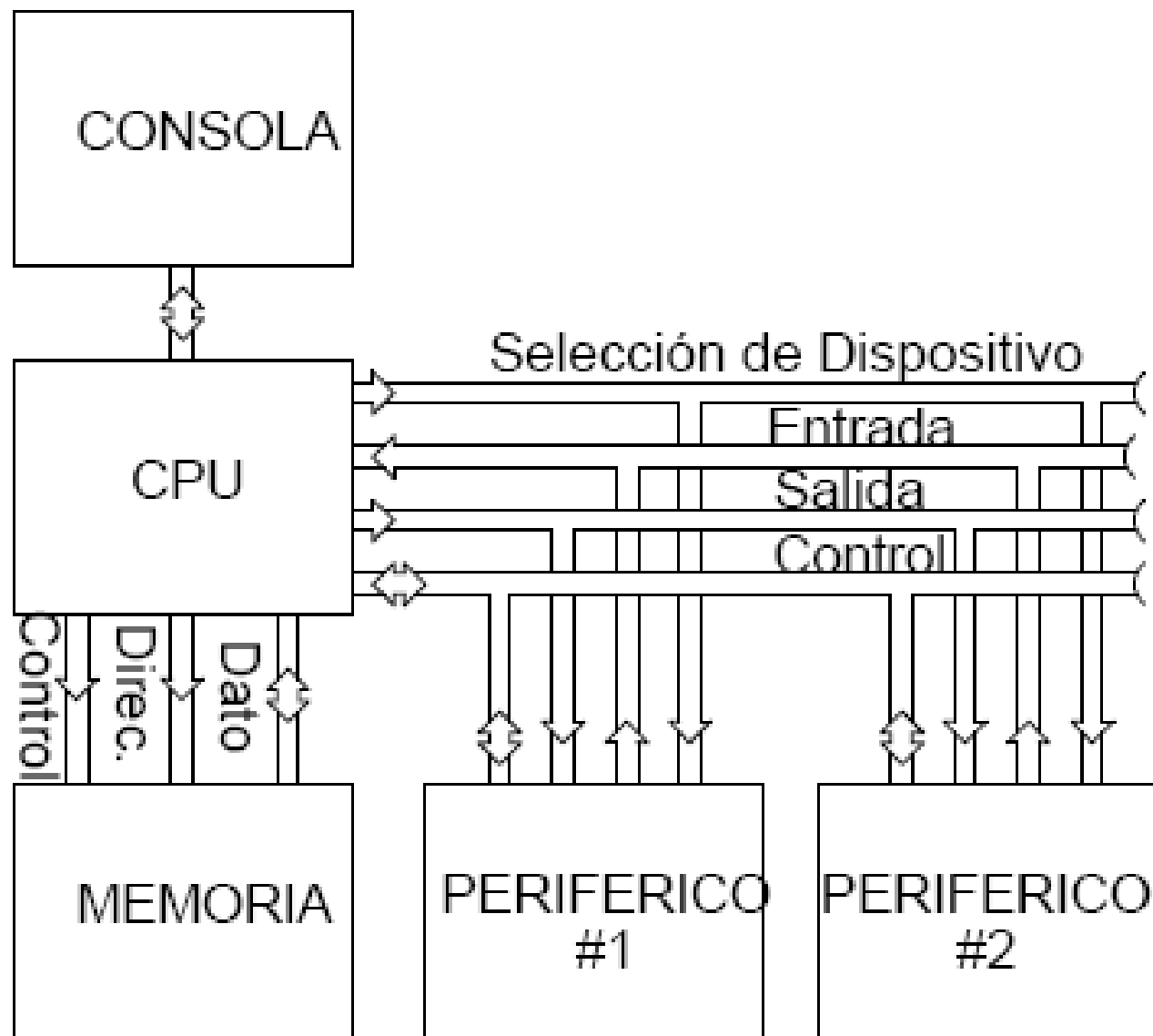


Fig 6: Esquema completo de la maquina

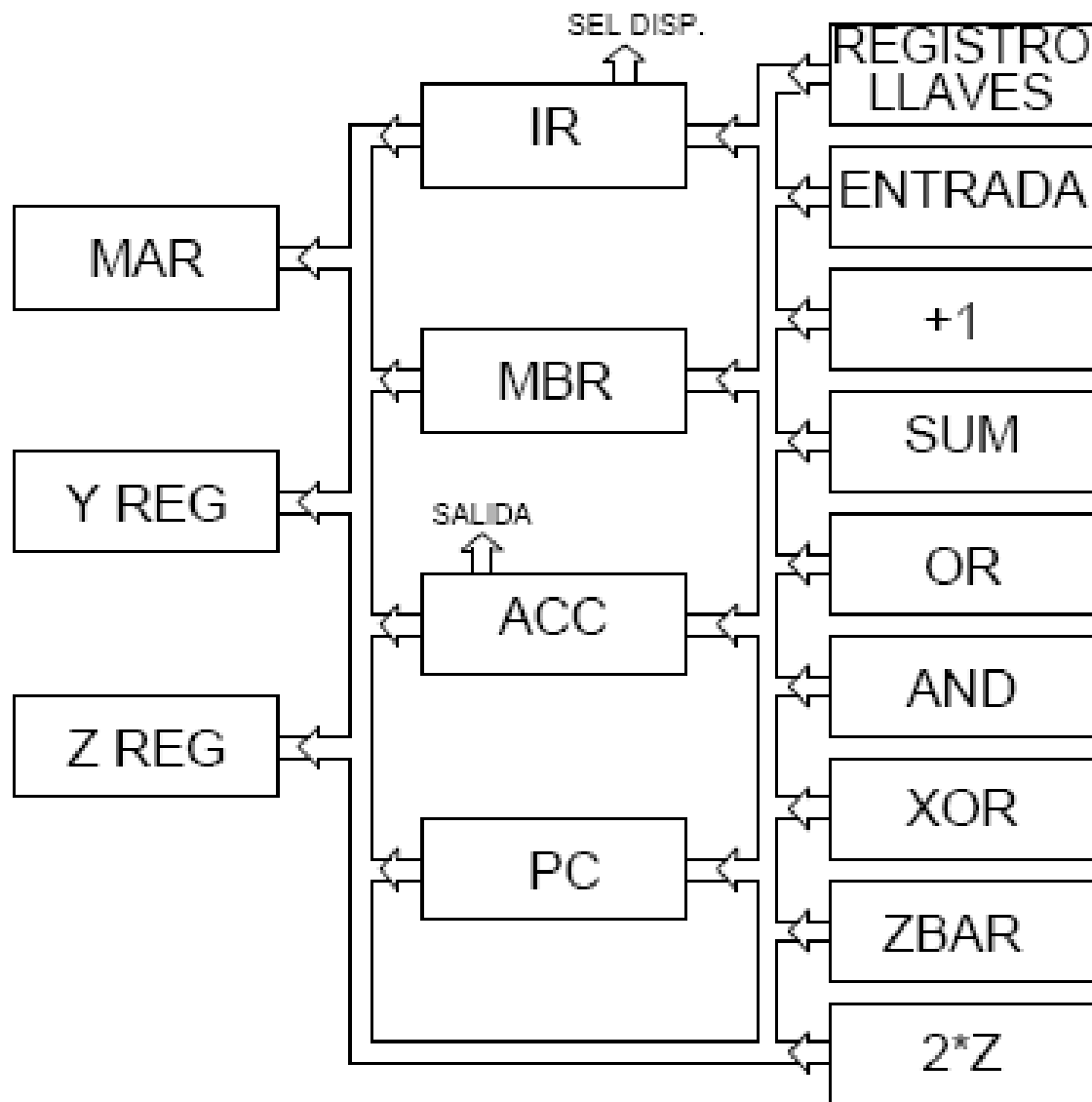


Fig. 7 El esquema de bus común usado en "BLUE".

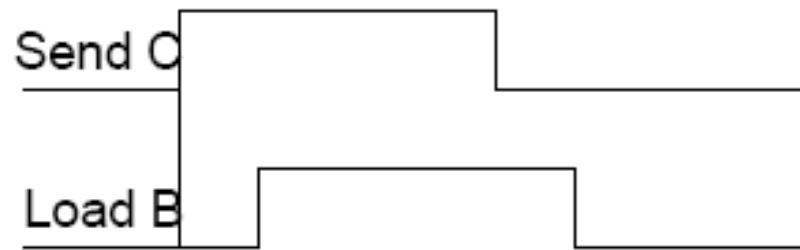


Fig. 9: Diagrama de los pulsos requeridos para copiar información desde el Registro C al Registro B.

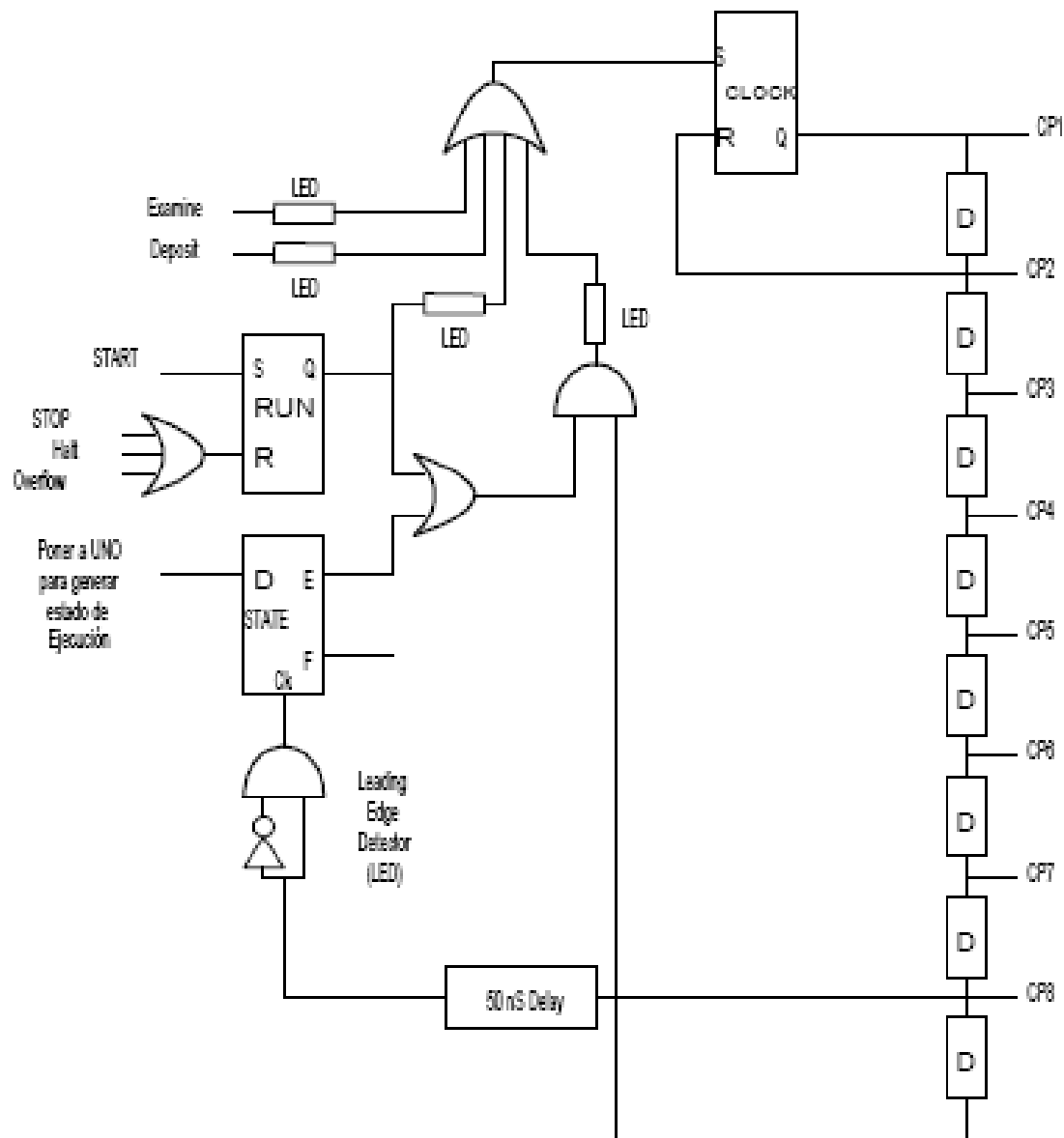


Fig. 10: Circuito de control y de reloj.

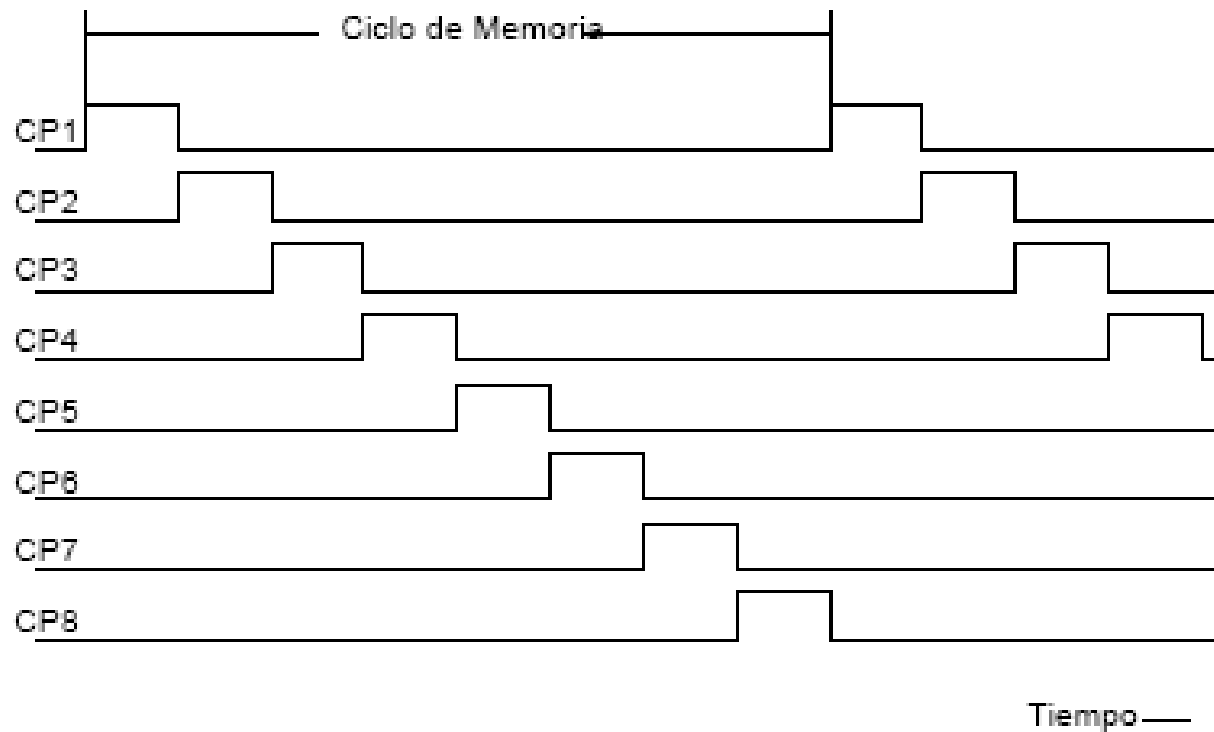


Fig. 11 Diagrama de tiempos de las señales de las líneas CP.

Reloj	Acción	Comentarios
1	Send PC, Load MAR, Load Z Memory Read	Copia el PC en MAR y Z. Inicia ciclo de lectura de memoria.
2	Send +1, Load Y	Hace Y igual a +1
3		
4	Send SUM, Load PC	Incrementa el PC en uno, de manera que apunte a la siguiente instrucción.
5	Strobe MBR	Carga el MBR con el dato proveniente de memoria.
6	Send MBR, Load IR	Copia la nueva instrucción en el registro de instrucciones y comienza su decodificación.
7 8		Disponibles para la decodificación de la instrucción y tal vez para la ejecución.

Tabla 2: El ciclo de búsqueda.

Reloj	HALT	NOP	JMP	JMA	SRJ	CSA	NOT	RAL
7	-	-	-	-	Send PC Load A	-	Send A Load Z	Send A Load Z
8	Off→ RUN	-	Send IR Load PC	Si A15=1 Send IR Load PC	Send IR Load PC	Send SR Load A	Send Z Bar Load A	Send 2*Z Load A

Tabla.3 Las ocho instrucciones de un ciclo en el final del ciclo de búsqueda.

Reloj	Acción		Comentarios
	Entrada INP	Salida OUT	
7	1 → TRA	1 → TRA	Transmite IR 5-0 a través de las líneas de selección de dispositivo
8	E → State	E → State	A continuación se procede con un estado de ejecución.
Ciclo de Búsqueda			
Ciclo de Ejecución			
1	-	-	
2	-	-	
3	-	-	
4	-	-	
5	-	-	
6	-	-	
7	Si R=1 Send Data Inp Lines Load A 0 → TRA	Si R=1 0 → TRA	Si el dispositivo está listo (R=1), se copia el dato y se limpia TRA
8	Si TRA=0 F → State	Si TRA=0 F → State	Si se ha completado el ciclo de Entrada/Salida, se procede con la búsqueda de una nueva instrucción. De otra manera, se repite el ciclo de ejecución.

Tabla.4 Las instrucciones de entrada/salida.

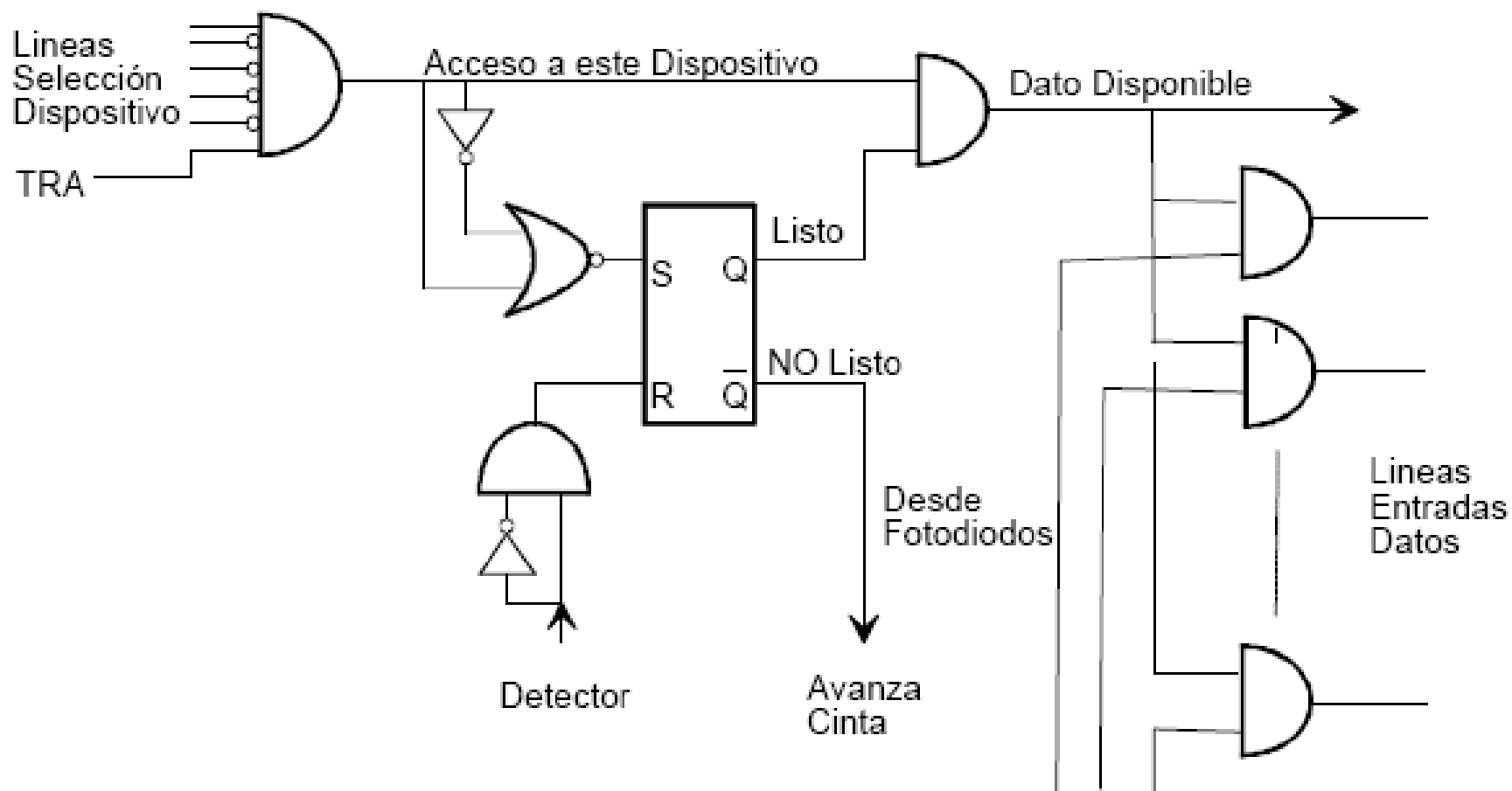


Fig 12: Los circuitos de la lectora de cinta de papel en BLUE.

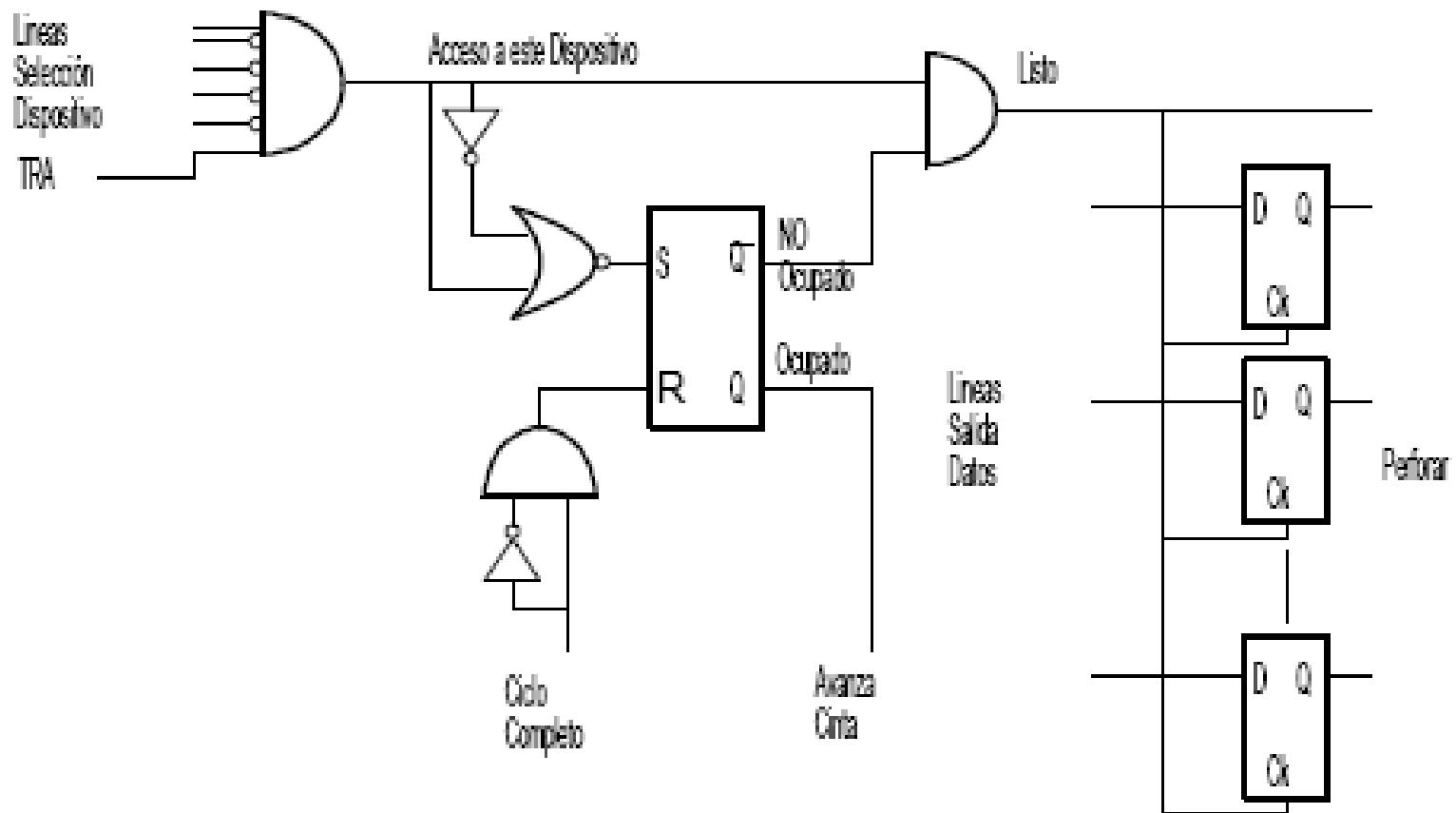


Fig.13 Circuito de la perforadora de papel

Reloj	Acción		Comentarios
	LDA	STA	
7			
8	E → State	E → State	A continuación se procede con un estado de ejecución.
Ciclo de Búsqueda			
Ciclo de Ejecución			
1	Send IR Load MAR Memory Read	Send IR Load MAR	Copia el IR ₁₂₋₀ en MAR. Inicia ciclo de lectura de memoria.
2	-	-	
3	-	-	
4	Strobe MBR	Send A Load MBR Memory Write	Carga MBR con el dato proveniente de memoria/acumulador. Inicia ciclo de escritura.
5	Send MBR Load A	-	Copia MBR (Dato) en acumulador.
6	-	-	
7	-	-	
8	F → State	F → State	Se procede con la búsqueda de una nueva instrucción.

Tabla 5 Instrucciones de carga y almacenamiento del acumulador.

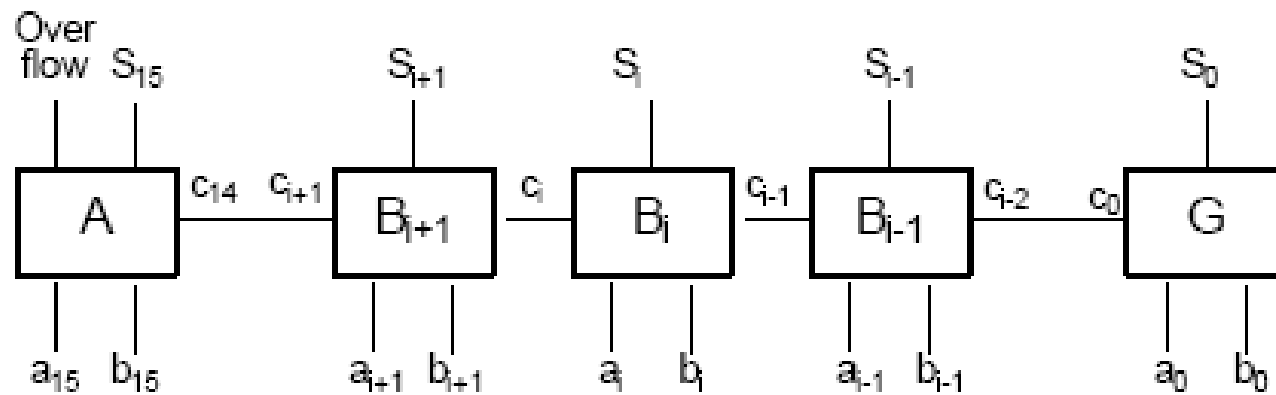


Fig 14 Un esquema de un sumador binario paralelo $A+B=S$ [JY15]

α

a_{15}	b_{15}	c_{14}	S_{15}	OV
0	0	0	0	0
0	0	1	1	1
0	1	0	1	0
0	1	1	0	0
1	0	0	1	0
1	0	1	0	0
1	1	0	0	1
1	1	1	1	0

 β

a_i	b_i	c_{i-1}	S_i	c_i
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

 γ

a_0	b_0	S_0	c_0
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Tabla. 6 Tabla de verdad de α, β y γ .

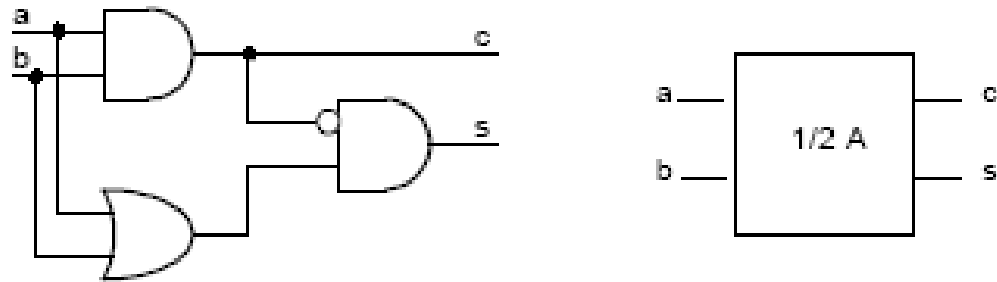


FIG.15 Semi Sumador y su representación simbólica.

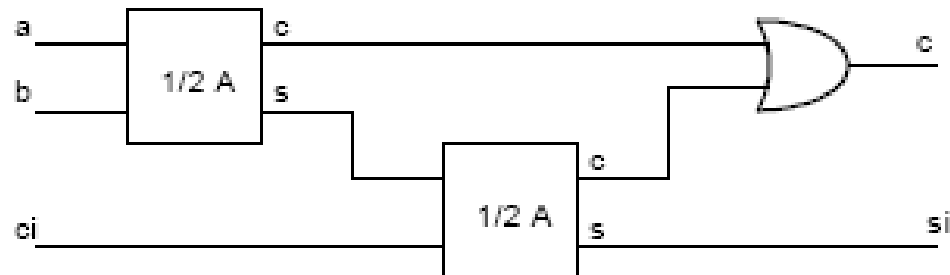


Fig.16: Un Sumador Total construido con 2 Semi Sumadores

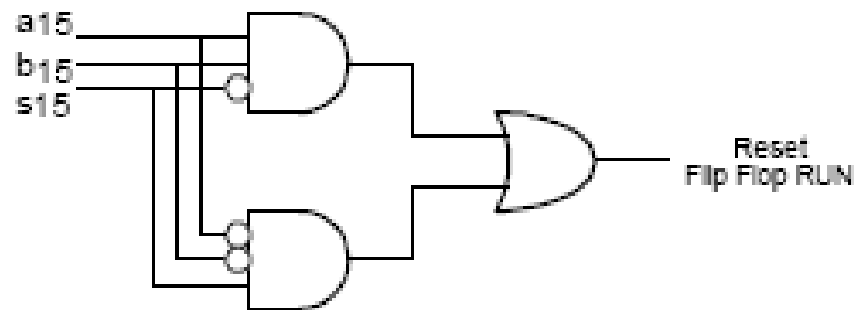


Fig.17: Circuito que genera una señal de rebosamiento.

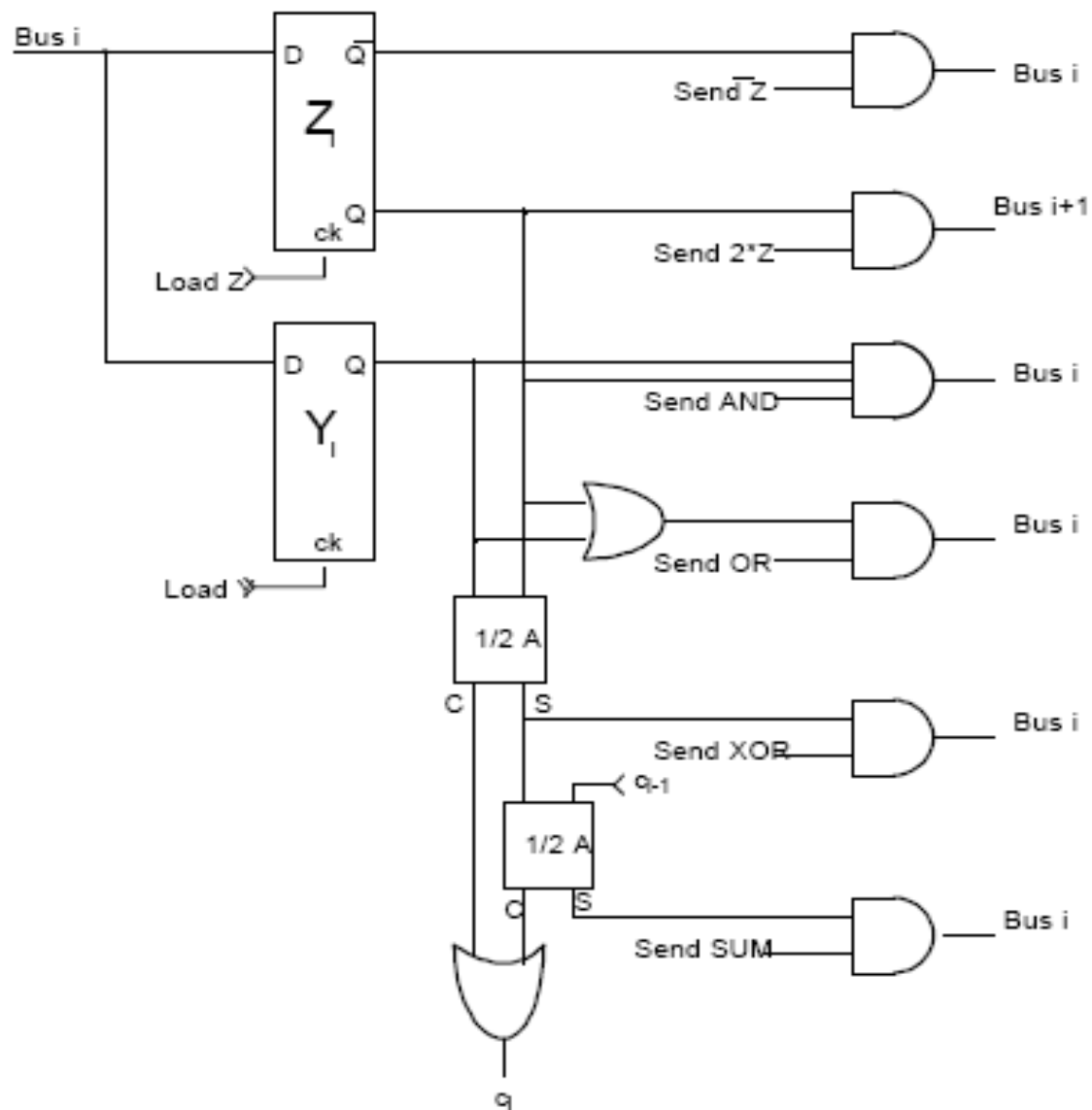


Fig.18: Un bit típico de la ALU.

Reloj	Acción			
	ADD	XOR	AND	IOR
7	-	-	-	-
8	E → State	E → State	E → State	E → State
Ciclo de Búsqueda				
Ciclo de Ejecución				
1	Send IR Load MAR Memory Read	Send IR Load MAR Memory Read	Send IR Load MAR Memory Read	Send IR Load MAR Memory Read
2	Send A Load Z	Send A Load Z	Send A Load Z	Send A Load Z
3	-	-	-	-
4	-	-	-	-
5	Strobe MBR	Strobe MBR	Strobe MBR	Strobe MBR
6	Send MBR Load Y	Send MBR Load Y	Send MBR Load Y	Send MBR Load Y
7	-	-	-	-
8	Send SUM Load A F → State	Send XOR Load A F → State	Send AND Load A F → State	Send OR Load A F → State

Tabla 7 La ejecución de las cuatro instrucciones aritméticas con referencia a memoria.

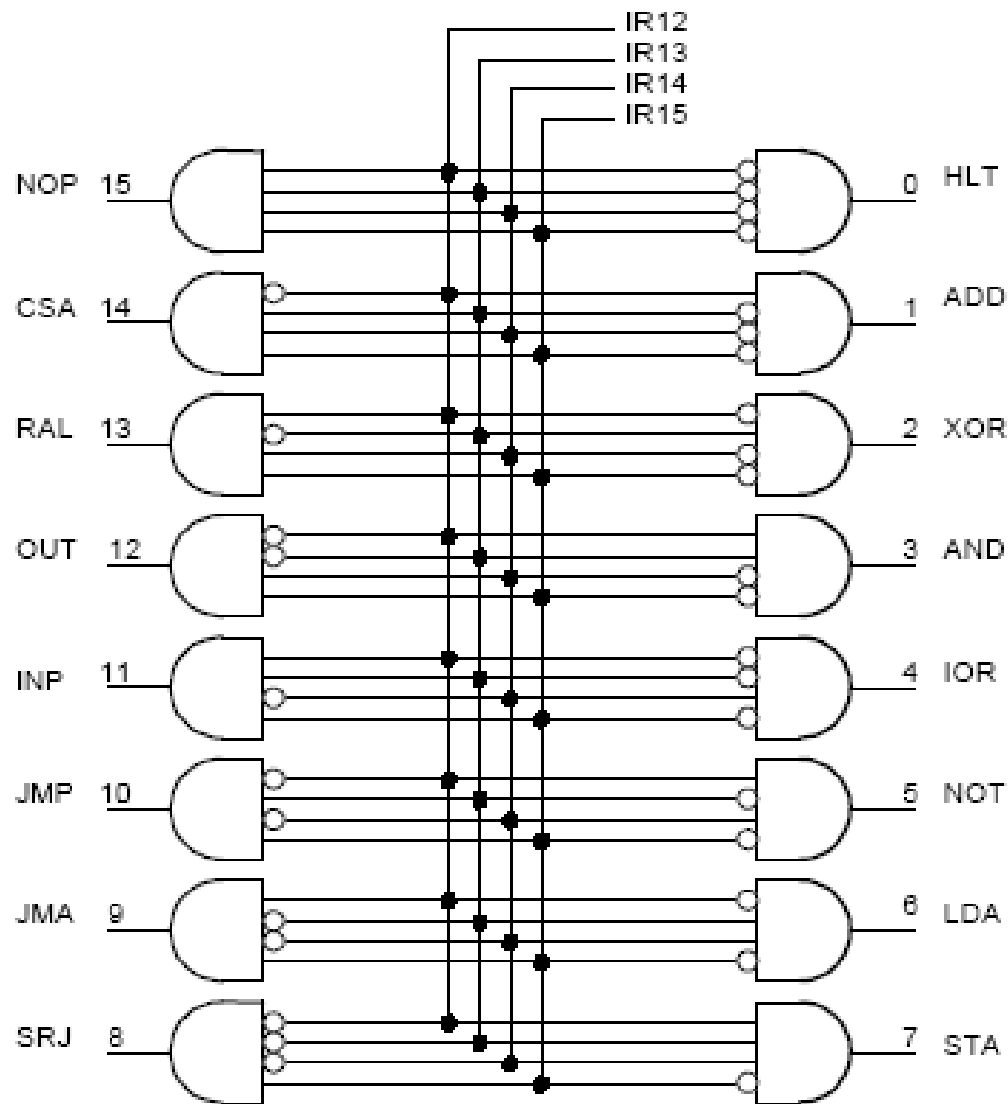


Fig 19: Arbol de decodificación completa.

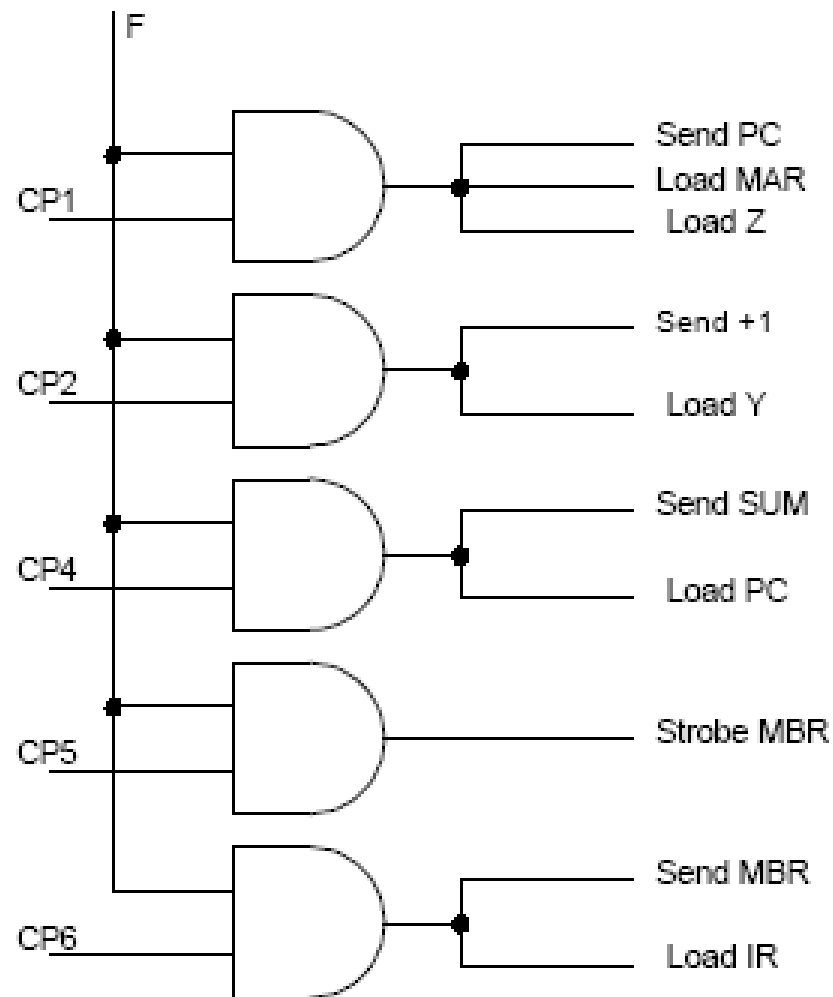


Fig. 20: Compuertas de la primera parte del ciclo de búsqueda

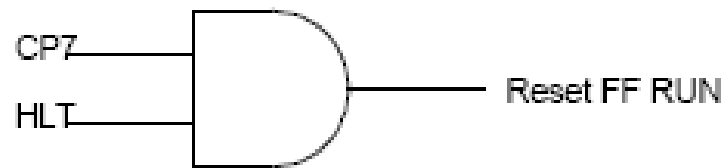


Fig. 21: Implementación de la instrucción Parar (Halt)

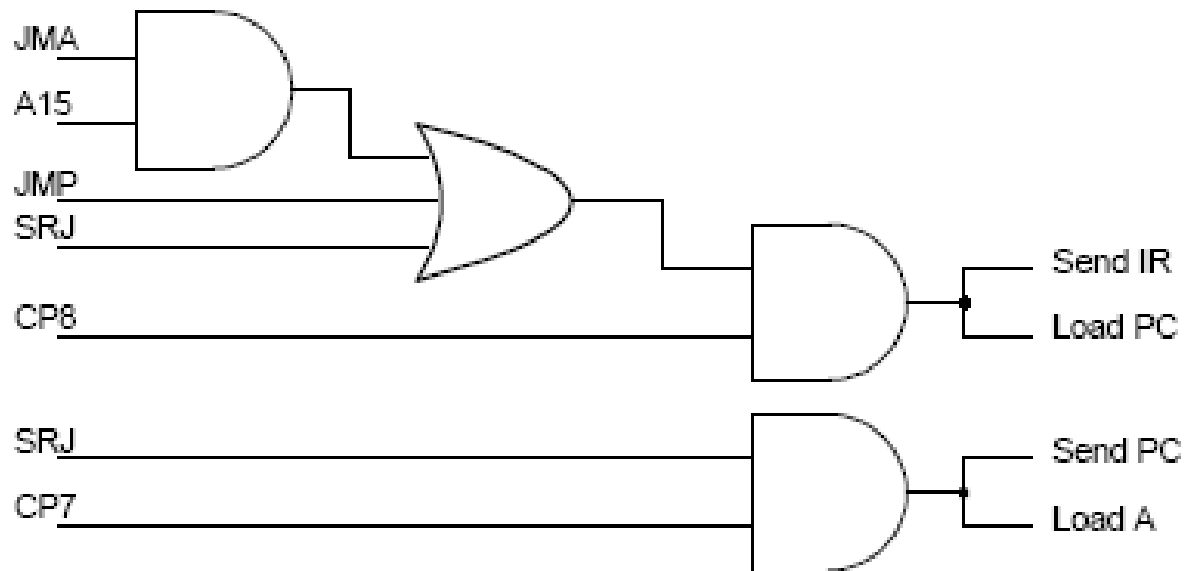


Fig. 22: La tres instrucciones de salto.

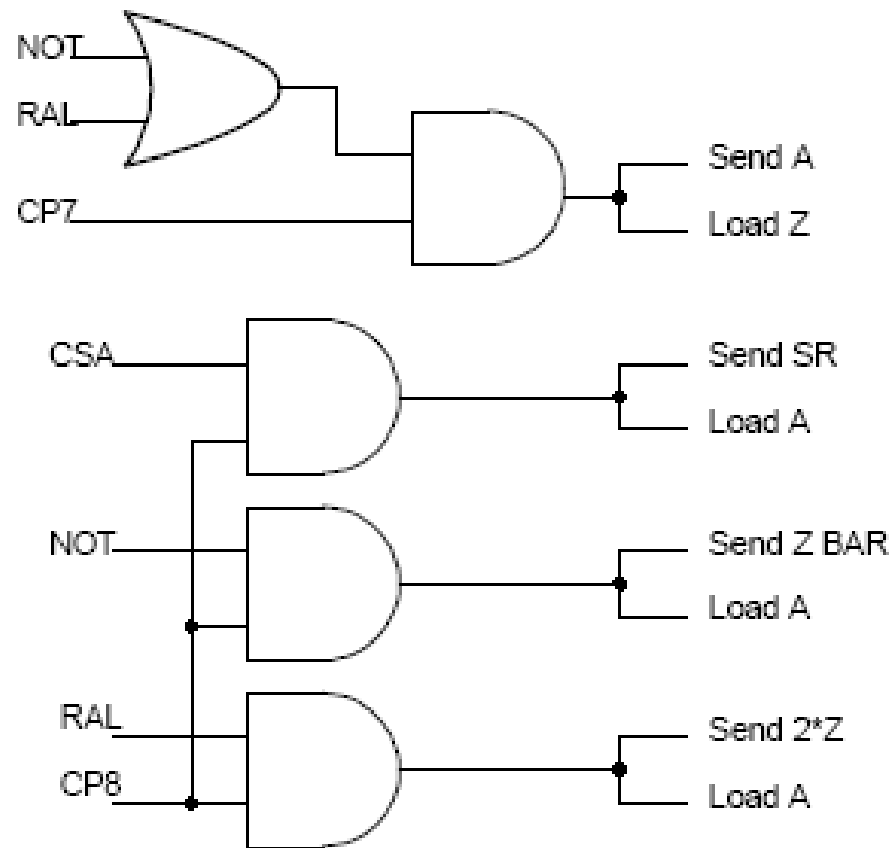


Fig. 23: Implementación de las instrucciones

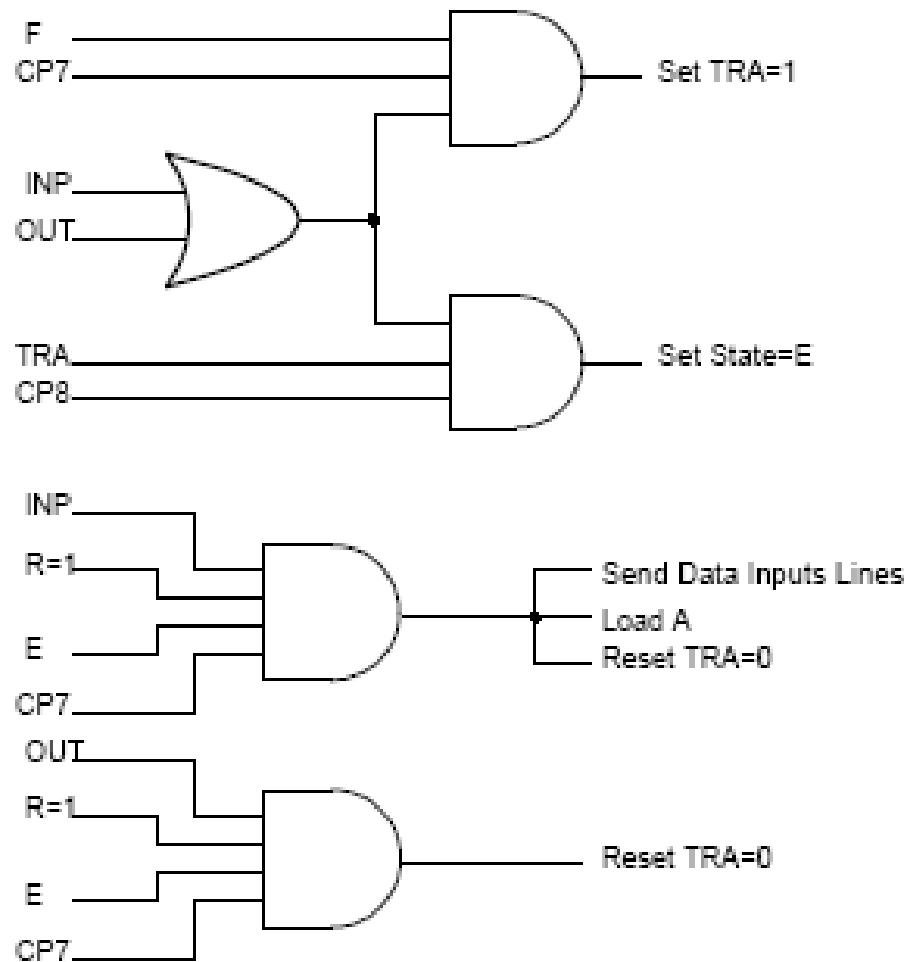


Fig. 24: Las instrucciones de Entrada y Salida.

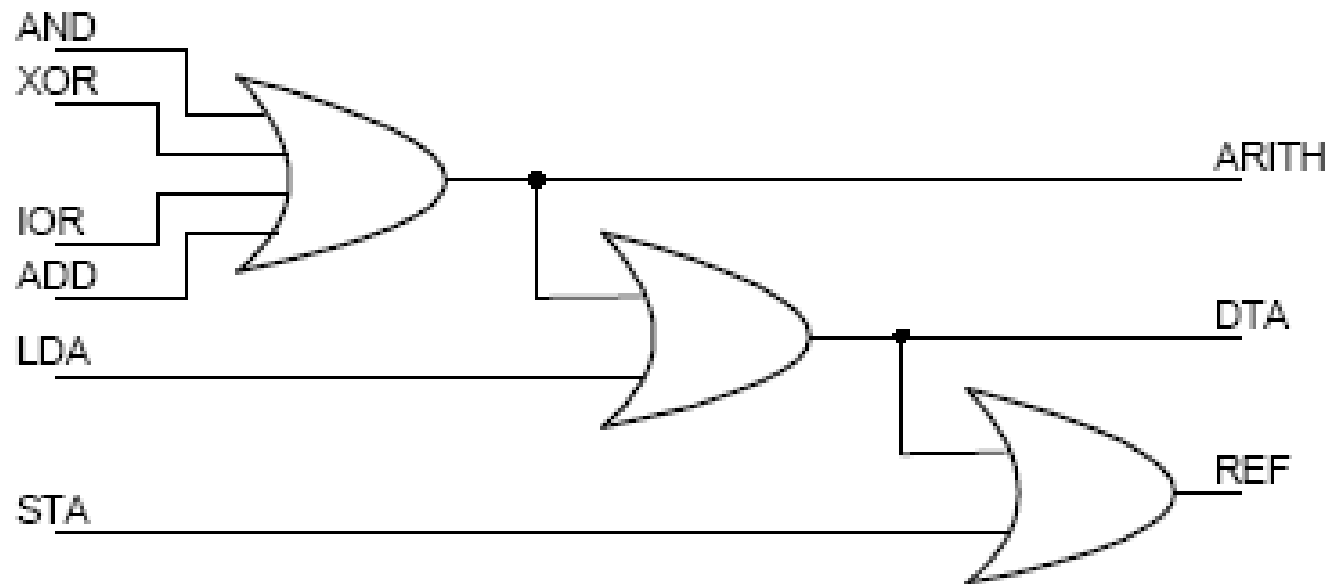


Fig. 25: Generación de las señales ARITH, DTA y REF.

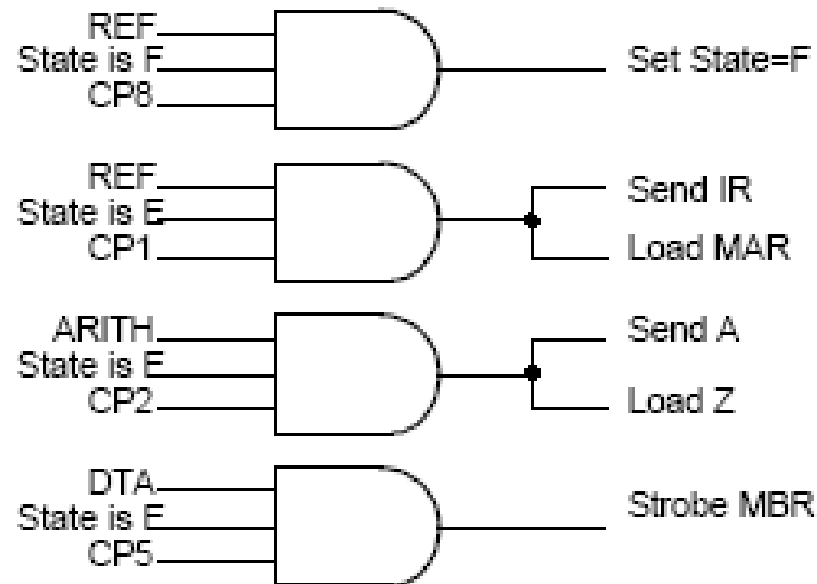


Fig. 26: Primeros pasos de las instrucciones de referencia a memoria

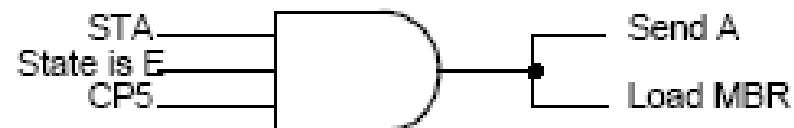


Fig. 27: El resto de la instrucción almacenar el acumulador (STORE)

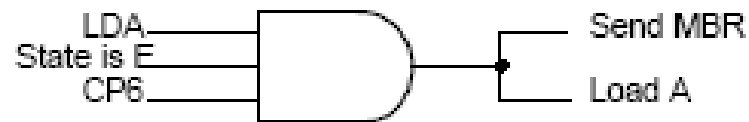


Fig. 28: El resto de la instrucción cargar el acumulador (LOAD)

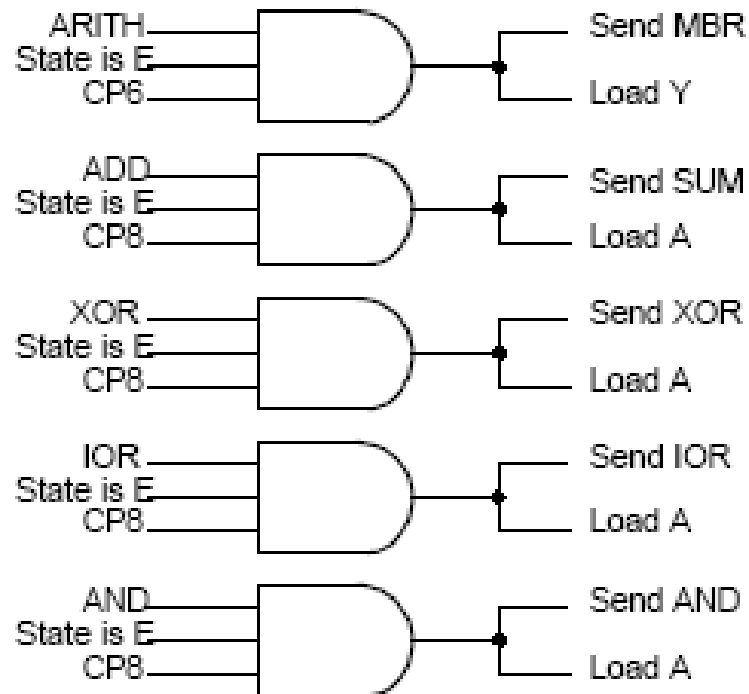


Fig. 29: El resto de las cuatro instrucciones aritméticas

Reloj	Acción		Comentarios
	Examine	Deposit	
1	Send PC Load MAR Load Z Memory Read	Send PC Load MAR Load Z Memory Read	Copia el PC en MAR y Z. Inicia ciclo de lectura de memoria.
2	Send +1 Load Y	Send +1 Load Y	Copia +1 en Y de la ALU.
3	-	-	
4	Send SUM Load PC	Send SUM Load PC	Incrementa PC en 1
5	Send MBR Load A	Send LLaves Load MBR Memory Write	
6	Send MBR Load IR	-	
7	-	-	
8	-	-	-

Tabla 8 Detalle de las operaciones generadas por la llaves Examinar y Depositar del panel frontal. Se asume aquí y en la Fig. 10 que dichas llaves proveen una señal limpia (sin rebotes) de al menos 1µs