

Ejercicio 10:

a) Siendo que las direcciones son de 8 bits, y que el direccionamiento es a palabra, lo cual es de 8 bits, eso que en una cadena de 8 bits pueda hacer 2^8 combinaciones (al ser cada bit 0 o 1) lo cual corresponde a la mayor cantidad de unidades direccionables. Con esto, eso que el tamaño máximo de memoria física equivale al producto entre la cantidad de unidades direccionables y su tamaño, quedando $2^8 \times 8 \text{ b} = \frac{16}{8} = 2^8 \text{ B}$

b) El tamaño del stack pointer debe ser de 8 bits, pues al estar el stack representado dentro de la memoria, este registro debe contener la dirección de memoria (de tamaño 8 bits) de la celda en la que pueden agregarse nuevos elementos a la misma.

c) Para el diseño del formato eso que las instrucciones deben ser de longitud fija, todas las instrucciones tienen un modo de direccionamiento definido, y hay 4 registros que se pueden distinguir con $\log_2(4) = 2$ bits. Con ello, eso las instrucciones ADD, JG y SWAP las cuales requieren que usen 4 b correspondiente a dos operandos de registros en cada uno (son los que requieren mayor espacio a usen). A su vez, eso que debe distinguirse los 3 con códigos de operación y ahí una cadena de 4 bits es suficiente pues $2^4 = 16 > 3$. Esto deja 13 combinaciones disponibles de dichos 4 para el código de operación del resto de las operaciones (son 3 pues la longitud de las instrucciones es de una palabra que es de 8 b). Comencemos los bits más significativos para el código de operación. Luego tomamos las operaciones LCTE4 y LCLE4 que deben usen 4 bits para la constante, y para los 4 bits superiores tienen 13 combinaciones disponibles para distinguir su código de los anteriores (dejando 11). Luego, para las 4 instrucciones rebotas debe usen 2 bits para números de registros y para los 4 bits superiores, tengo 11 combinaciones disponibles, y a su vez, por cada una tengo todas las combinaciones de los dos rebotas pues los primeros 4 ya los distinguí de los anteriores. Esto nos deja $11 \cdot 2 = 44$ combinaciones disponibles para distinguir 4 códigos, lo cual es más que suficiente y al diseño termino. Ahora que a hora tomaremos un diseño particular de todos los implementables: Formato completo

Instrucción	Formato	Instrucción	Formato
ADD R _i R _j	0000 Registro (2b) Registro (2b)	STORE R _i	0101 00 Registro (2b)
JG R _i R _j	0001 Registro (2b) Registro (2b)	LOAD R _i	0101 01 Registro (2b)
SWAP R _i R _j	0010 Registro (2b) Registro (2b)	ZERO R _i	0101 10 Registro (2b)
LCTE4 cte4	0010 Constante de 4 bits	NOT R _i	0101 11 Registro (2b)
LCLE4 cte4	0100 Constante de 4 bits		

d) Siendo que el formato de instrucciones es de longitud fija de una palabra (8 b) eso que si se pudieran agregar instrucciones, estas deberían tener un código de operación que los distinga del resto. Para los 4 bits superiores, eso