

Las direcciones y los datos son bits, pero se pueden representar

en formato hexadecimal (cada grupo de 4 bits) es un nible

por lo que las direcciones se separan en nibles.

Si al dividir en grupos de 4 no se acompleta un nible, aun asi se

representa usando un nible.

Por ejemplo 13/4 = 3 nibles y sobra 1 bit. Quedando el rango :

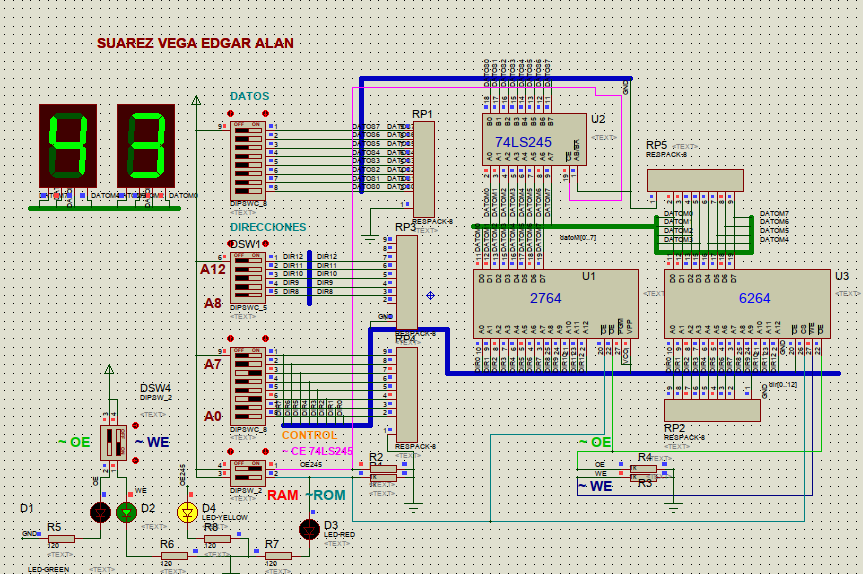
**0000 0000 0000** hasta **1 1111 1111 1111**

en hexadecimal **0 0000** hasta **1 FFFF**

Ejercicio LEER en la ROM la Direccion

0 024 : que direccion nos da

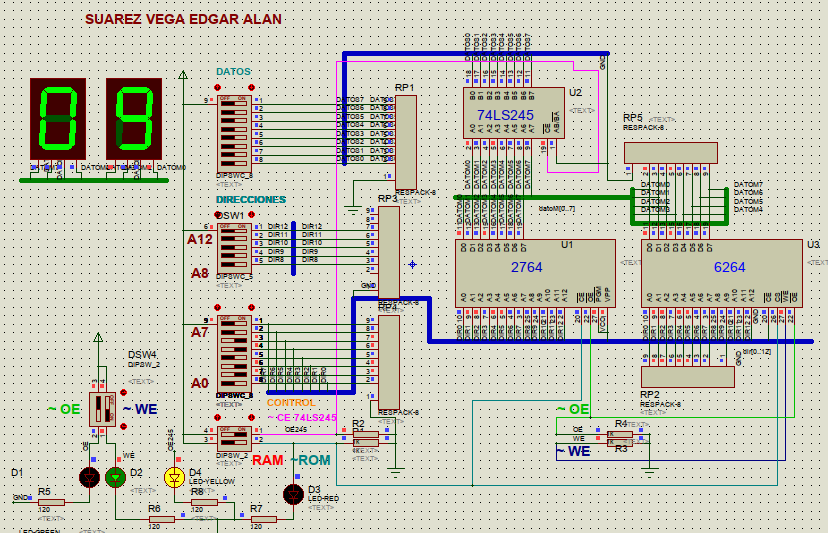
0 0000 0010 0100 : **0100 0011 (43)**



Ejercicio LEER en la ROM la Direccion

0 036 : que direccion nos da

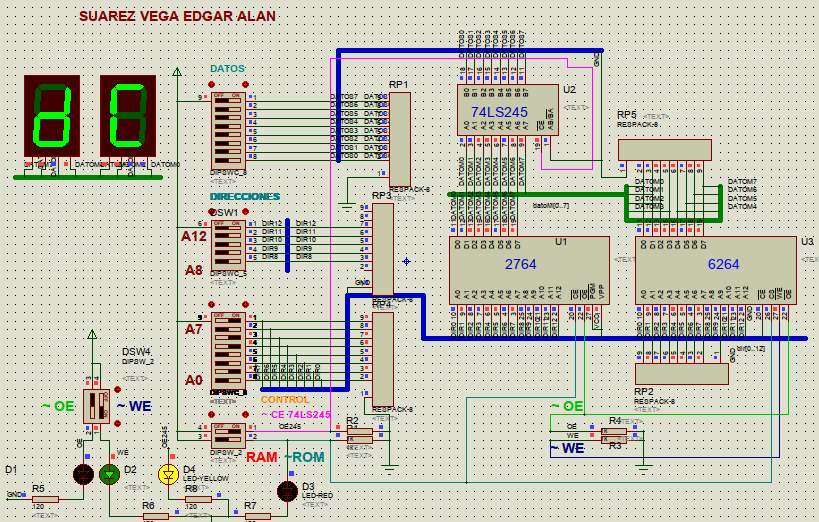
**0 0000 0011 0110 : 0000 1001 (09)**



Ejercicio LEER en la ROM la Direccion

0 0B2 : que direccion nos da

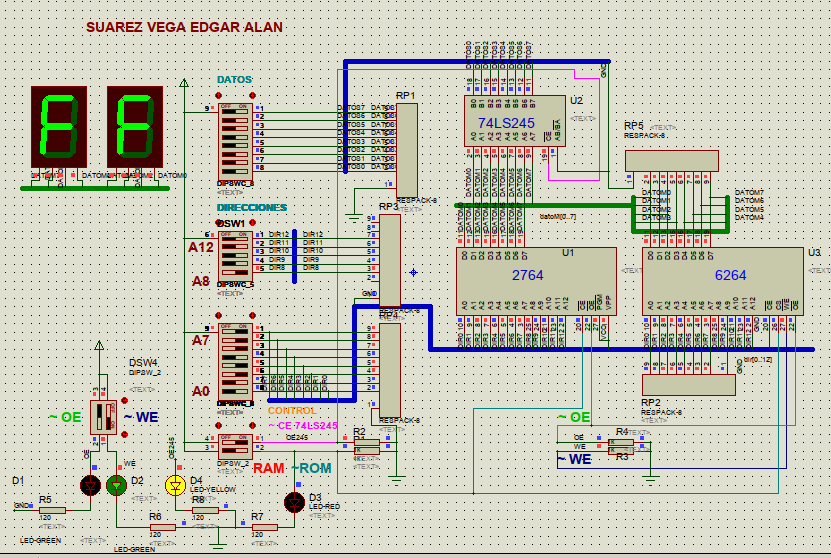
**0 0000 1011 0010 : 1101 1100 (DC)**



Ejercicio LEER en la ROM la Direccion

0 1EC : que direccion nos da

**0 0001 1110 1100 : 1111 1111 (FF)**



Un archivo .Hex para simular tiene que tener el valor de entrada fila 1

y uno de cierre (ultima linea).

El contenido de las direcciones empieza después de los primeros 8 bits de la 2 linea y así en cada linea después de sus primeros 8 bits , la primera linea tiene de 0-15 y así sucesivamente

Un archivo .Hex para simular tiene que tener el valor de entrada fila 1

y uno de cierre (ultima linea).