Tarea 1:

Números Binarios y Circuitos Combinacionales

Mauricio Solar

Arquitectura de Computadores

Primer Semestre de 2016

Benjamín Alberto Jorquera Jorquera

201473521-9

Tabla de Contenido

1. Introducción
2. Desarrollo
3. Resultados y Conclusiones
4. Bibliografía

1. Para el desarrollo de esta tarea, se ha pedido utilizar el programa LOGISIM para la creación de ciertos circuitos propuestos, para así aprender sobre su funcionamiento y su respectiva optimización. El programa trabajará con números binarios, ósea solo pueden ser 0 o 1, al igual que su salida. A continuación se muestran las tablas con las operaciones realizadas.

2.1 Ejercicio 1:

El ejercicio 1 consta de diseñar un circuito que pueda realizar la operación Complemento a 2, con una entrada que será X de 4 bits y una salida –X. Para esto se analizara la siguiente tabla:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | Y1 | Y2 | Y3 | Y4 |
| 0 | 0 | 0 | 0 | 0 | X | X | X | X |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |
| 2 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 4 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 |
| 6 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
| -8 | 1 | 0 | 0 | 0 | X | X | X | X |
| -7 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| -6 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 |
| -5 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 1 |
| -4 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| -3 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 |
| -2 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 |
| -1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 |

La tabla 1 muestra por cada fila el número correspondiente a los bits de las entradas (A, B, C, D), y las salidas (Y1, Y2, Y3, Y4) muestra su respectivo complemento a 2, para lograr esto se realiza un cambio en el número binario original y sumarle un 1, en este caso manualmente. En la tabla existen dos números que tendrán X (ósea DONT CARE) como salida, eso quiere decir que no existe -0 ni -8 como binario.

A continuación se procede a elaborar los mapas de Karnaugh y ver sus agrupaciones para elaborar las salidas finales correspondientes en forma de ecuación, llenamos la tabla con las salidas generadas de 4 en 4 bits, el K-MAP será transformado a ecuacion luego de calcular que entradas oscilan y cuáles no, en caso de que no asegurarse de si están negadas o no (filas: AB, columnas: CD):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | X | 1 | 0 | X |
| 01 | 1 | 1 | 0 | 0 |
| 11 | 1 | 1 | 0 | 0 |
| 10 | 1 | 1 | 0 | 0 |

Y1=

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | X | 1 | 1 | X |
| 01 | 1 | 0 | 0 | 1 |
| 11 | 1 | 0 | 0 | 1 |
| 10 | 1 | 0 | 0 | 1 |

Y2=

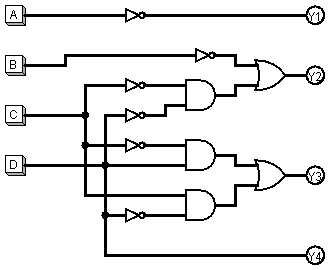
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | 0 | 0 | 0 | 0 |
| 10 | 1 | 1 | 1 | 1 |

Y3=

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 00 | 01 | 11 | 10 |
| 00 | 0 | 0 | 0 | 0 |
| 01 | 1 | 1 | 1 | 1 |
| 11 | 1 | 1 | 1 | 1 |
| 10 | 0 | 0 | 0 | 0 |

Y4=

3.1 Ya con esto damos paso a nuestro circuito de la siguiente forma:



2.2 Ejercicio 2:

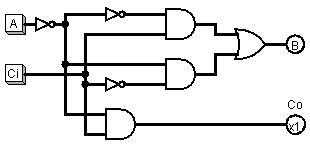
Como en el ejercicio 1 (ver tabla 1), esta vez para una implementación de 16 bits, se tendrá que dar el uso de un mismo circuito de suma binaria y volverla una suma binaria completa, es decir, replicarlo, este circuito consta de un bucle que reacciona volteando el numero binario original y sumarle un CarryIn, el resultante serán 2 salidas, el nuevo número de la suma y un CarryOut que será nuestro nuevo CarryIn, de esta manera tendremos la siguiente tabla con su respectivo K-MAP:

|  |  |  |  |
| --- | --- | --- | --- |
| A | Carryin | Yi | Carryout |
| 0 | **0** | **0** | **0** |
| 0 | **1** | **1** | **0** |
| 1 | **0** | **1** | **0** |
| 1 | **1** | **0** | **1** |

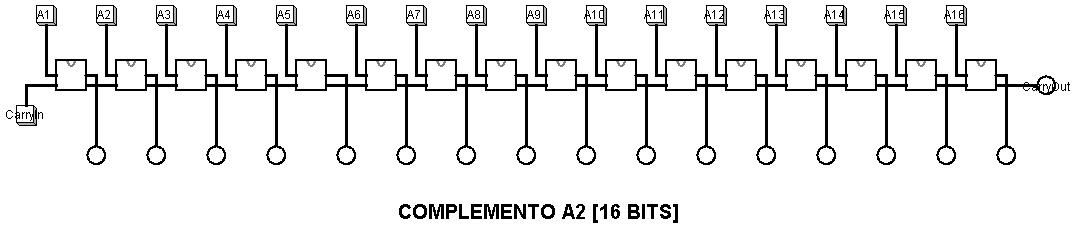
|  |  |  |
| --- | --- | --- |
| Yi | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 0 |

|  |  |  |
| --- | --- | --- |
| CarryOut | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 0 |

3.2



Este circuito corresponde a la suma



4.

<http://www.angelfire.com/electronic2/logicacomputacional/Respuestas/cmplogicas.htm>

<http://slideplayer.es/slide/4194061/>

http://www.cburch.com/logisim/docs/2.1.0-es/guide/prop/oscillate.html