Tarea 1 INF-245:

Predictor de ganador en competencias mediante circuitos combinacionales

Integrantes:

- **Claudio Inal ROL: 201873060-2**
- **Iván Cano ROL: 202073543-3**

Resumen

Este informe ilustra un método para obtener un circuito combinacional que se encarga de realizar una tarea en específico, la de determinar un hipotético ganador en una competencia entre 2 contrincantes, todo esto mediante el uso de tablas de verdad, funciones, mapas de Karnaugh y circuitos en Logisim.

En el enunciado se nos informa de 8 competidores, es decir tenemos 64 posibles rondas entre 2 contrincantes. Se muestran los resultados del nivel de éxito del circuito para todos los casos a continuación:

$$\text{Éxitos totales} = \frac{64}{64} = 100\%$$

Introducción

El objetivo de este informe es dar a conocer una estrategia para conseguir obtener una solución a un problema dado mediante circuitos combinacionales.

El problema en cuestión se encuentra en el enunciado de la Tarea 1 dado, el cual descrito escuetamente nos pide que encontremos una forma de predecir quién será el ganador entre dos contrincantes en una competencia. Desde el enunciado obtenemos información acerca de estos competidores, como el nombre, peso, altura, fuerza, agilidad, resistencia y un número identificador que nos ayudará a sintetizar de mejor manera el resultado obtenido. También se nos da una fórmula que calcula una ponderación del “poder absoluto” de cada competidor en diferentes casos, así, prediciendo de forma efectiva el ganador de esa ronda. Esta fórmula la utilizaremos para establecer un ganador en cada caso, para luego crear una tabla de verdad en donde definiremos nuestras variables de entrada basándonos en el identificador de los dos competidores involucrados en la ronda, siendo nuestras variables de salida el identificador de aquel competidor que haya resultado ganador en los cálculos anteriores.

Luego de establecer nuestra tabla de verdad, haremos uso de los mapas de Karnaugh para obtener funciones minimizadas (OR de AND) que luego serán de utilidad para armar un circuito combinacional en Logisim, el cual dado 6 entradas (3 por cada competidor) produce 3 salidas dando cuenta del identificador del competidor ganador.

Desarrollo

Como se mencionó anteriormente en la introducción, calcularemos quién es el ganador de entre 2 contrincantes en diferentes casos.
Para cada uno de los 8 competidores tenemos la siguiente información:

- Nombre del competidor
- Identificador (número entre 0 y 7)
- Estatura (centímetros)
- Peso (kilogramos)
- Fuerza (STR)
- Agilidad (AGI)
- Resistencia (RES)

Fórmula para determinar el Poder Absoluto (PA) de cada competidor, en 3 diferentes casos:

- Caso 1: $PA(Peso < 75[kg]) = \frac{Estatura}{AGI} + \frac{100}{Estatura} + AGI$
- Caso 2: $PA(Peso \geq 75[kg]) = 5 \times Peso + 2 \times RES$
- Caso 3: $PA(Otro) = \frac{Estatura}{AGI} + 3 \times Peso + \frac{STR+AGI+RES}{3}$

Haciendo uso de una hoja de cálculo obtenemos el PA de cada competidor en todos los casos, los cuales se presentan en la siguiente tabla (los marcados en rojo, son casos imposibles):

Competidor	ID	Estatura	Peso	STR	AGI	RES	PA(Peso<75)	PA(Peso>75)	PA(Otro)
Tike Myson	0	169	64	3	8	4	29.7	328.0	218.1
Mey Rysterio	1	183	77	6	6	2	37.0	389.0	266.2
Chon Jena	2	178	73	7	4	3	49.1	371.0	268.2
Black Sword	3	172	59	7	8	1	30.1	297.0	203.8
Sun Ku-Wong	4	160	70	6	10	4	26.6	358.0	232.7
Class Joe	5	178	50	3	5	2	41.2	254.0	188.9
Mr. ManSand	6	196	129	10	3	5	68.8	655.0	458.3
Moro Gajima	7	186	80	4	8	7	31.8	414.0	269.6

Tabla 1: Información de cada competidor con su respectivo poder absoluto.

Haremos uso de una matriz 8x8 para dar a conocer quién es el ganador cuando compiten *i* contra *j* (*i,j* = 0, 1, …, 7), siendo cada una el identificador del competidor.

Aclaraciones:

- Si un competidor se enfrenta a sí mismo, este gana.
- Si ambos competidores tienen el mismo Poder Absoluto, gana cualquier competidor.
- Se ha marcado con color azul para los enfrentamientos en donde ambos competidores tienen un peso menor a 75[kg], rojo para aquellos donde ambos tengan un peso mayor o igual a 75[kg] y amarillo en otro caso.

ID	0	1	2	3	4	5	6	7
0	0	1	2	3	0	5	6	7
1	1	1	2	1	1	1	6	7
2	2	2	2	2	2	2	6	7
3	3	1	2	3	3	5	6	7
4	0	1	2	3	4	5	6	7
5	5	1	2	5	5	5	6	7
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	6	7

Tabla 2: Matriz de ganadores según los competidores de la ronda.

Notar que hay 64 posibles situaciones, sin embargo hay casos repetidos debido a que una ronda en donde se enfrenten *i* contra *j,* es igual que si se enfrentan *j* contra *i,* por lo que tenemos una matriz simétrica con respecto a la diagonal. A pesar de este inconveniente, llevaremos la información de esta matriz a una tabla de verdad con todos los 64 casos, luego el uso de mapas de Karnaugh se hará cargo de minimizar la cantidad de funciones obtenidas en caso de repeticiones.
Dado que tenemos 2 pines (A y B) de 3 bits cada uno para la entrada de competidores, transformaremos nuestros identificadores de competidor de decimal a binario, y con un pin (G) de salida de 3 bits, haremos lo mismo con el identificador del ganador:

A2	A1	A0	B2	B1	B0	G2	G1	G0
0	0	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0	1
0	0	0	0	1	0	0	1	0
0	0	0	0	1	1	0	1	1
0	0	0	1	0	0	0	0	0
0	0	0	1	0	1	1	0	1
0	0	0	1	1	0	1	1	0
0	0	0	1	1	1	1	1	1
0	0	1	0	0	0	0	0	1
0	0	1	0	0	1	0	0	1
0	0	1	0	1	0	0	1	0
0	0	1	0	1	1	0	0	1
0	0	1	1	0	0	0	0	1
0	0	1	1	0	1	0	0	1
0	0	1	1	1	0	1	1	0
0	0	1	1	1	1	1	1	1

A2	A1	A0	B2	B1	B0	G2	G1	G0
0	1	0	0	0	0	0	1	0
0	1	0	0	0	1	0	1	0
0	1	0	0	1	0	0	1	0
0	1	0	0	1	1	0	1	0
0	1	0	1	0	0	0	1	0
0	1	0	1	0	1	0	1	0
0	1	0	1	1	0	1	1	0
0	1	0	1	1	1	1	1	1
0	1	1	0	0	0	0	1	1
0	1	1	0	0	1	0	0	1
0	1	1	0	1	0	0	1	0
0	1	1	0	1	1	0	1	1
0	1	1	1	0	0	0	1	1
0	1	1	1	0	1	1	0	1
0	1	1	1	1	0	1	1	0
0	1	1	1	1	1	1	1	1

Tabla 3 y 4: Tabla de verdad para las entradas A y B, y salida G.

A2	A1	A0	B2	B1	B0	G2	G1	G0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	1
1	0	0	0	1	0	0	1	0
1	0	0	0	1	1	0	1	1
1	0	0	1	0	0	1	0	0
1	0	0	1	0	1	1	0	1
1	0	0	1	1	0	1	1	0
1	0	0	1	1	1	1	1	1
1	0	1	0	0	0	1	0	1
1	0	1	0	0	1	0	0	1
1	0	1	0	1	0	0	1	0
1	0	1	0	1	1	1	0	1
1	0	1	1	0	0	1	0	1
1	0	1	1	0	1	1	0	1
1	0	1	1	1	0	1	1	0
1	0	1	1	1	1	1	1	1

A2	A1	A0	B2	B1	B0	G2	G1	G0
1	1	0	0	0	0	1	1	0
1	1	0	0	0	1	1	1	0
1	1	0	0	1	0	1	1	0
1	1	0	0	1	1	1	1	0
1	1	0	1	0	0	1	1	0
1	1	0	1	0	1	1	1	0
1	1	0	1	1	0	1	1	0
1	1	0	1	1	1	1	1	0
1	1	1	0	0	0	1	1	1
1	1	1	0	0	1	1	1	1
1	1	1	0	1	0	1	1	1
1	1	1	0	1	1	1	1	1
1	1	1	1	0	0	1	1	1
1	1	1	1	0	1	1	1	1
1	1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	1	1

Tabla 5 y 6: Tabla de verdad para las entradas A y B, y salida G.

A continuación trasladamos la información de las tablas de verdad a mapas de Karnaugh, tenemos 6 variables, por lo tanto tendremos $2^6 = 64$ celdas en cada mapa, por cada función de salida G existente, empezando por G_2, luego G_1 y terminando por G_0. Hay que tener en cuenta que este es un K-map de 6 variables, por lo tanto vamos a dividir el mapa en 4 cuadrantes en donde 2 de las 6 variables (A2 y B2) estarán fijas por cada cuadrante, es decir, en el cuadrante superior izquierdo A2=0 y B2=0, en el cuadrante superior derecho A2=0 y B2=1, etc. Esto nos servirá para disminuir un K-map de 6 variables a uno de solamente 4. También aprovechamos de marcar inmediatamente los grupos de 1's adyacentes en cada K-map:

	B1B0	$\overline{B2}$				$B2$			
A1A0		00	01	11	10	00	01	11	10
$\overline{A2}$	00	0	0	0	0	0	1	1	1
	01	0	0	0	0	0	0	1	1
	11	0	0	0	0	0	1	1	1
	10	0	0	0	0	0	0	1	1
$A2$	00	0	0	0	0	1	1	1	1
	01	1	0	1	0	1	1	1	1
	11	1	1	1	1	1	1	1	1
	10	1	1	1	1	1	1	1	1

	B1B0	$\overline{B2}$				$B2$			
A1A0		00	01	11	10	00	01	11	10
$\overline{A2}$	00	0	0	1	1	0	0	1	1
	01	0	0	0	1	0	0	1	1
	11	1	0	1	1	1	0	1	1
	10	1	1	1	1	1	1	1	1
$A2$	00	0	0	1	1	0	0	1	1
	01	0	0	0	1	0	0	1	1
	11	1	1	1	1	1	1	1	1
	10	1	1	1	1	1	1	1	1

Tablas 7 y 8: Mapas de Karnaugh para las funciones de salida G2 y G1 respectivamente.

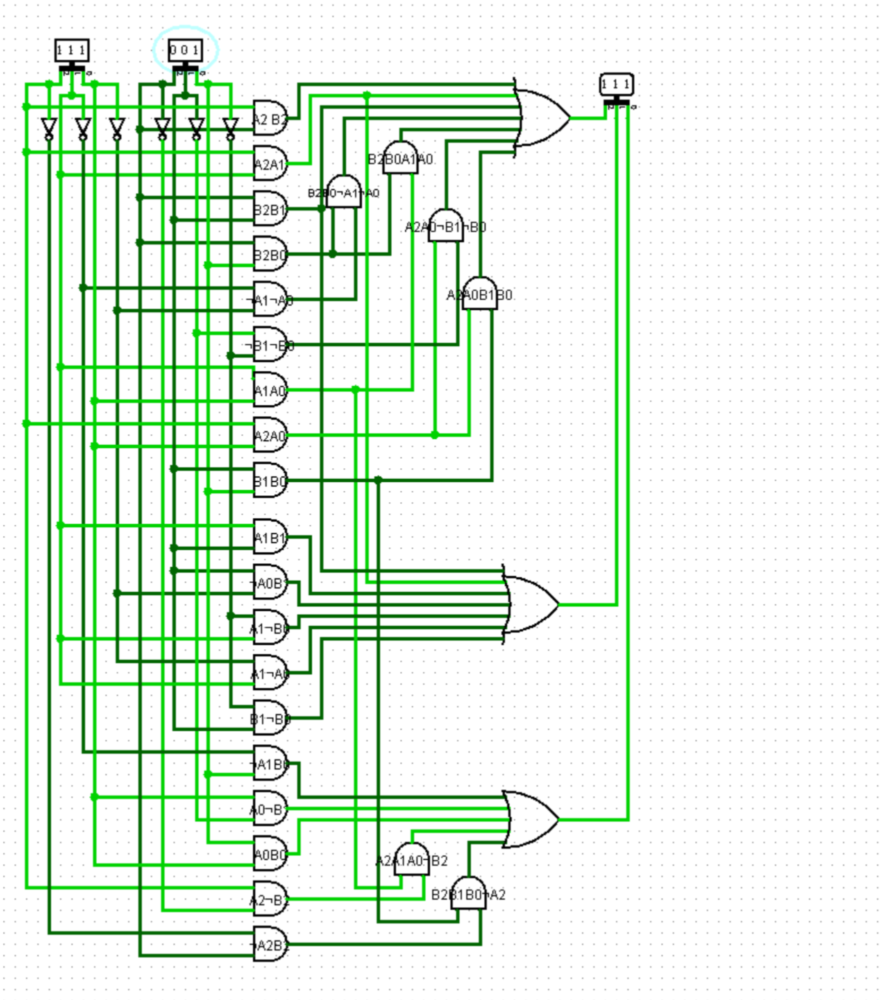
	B1B0	$\overline{B2}$				$B2$			
A1A0		00	01	11	10	00	01	11	10
$\overline{A2}$	00	0	1	1	0	0	1	1	0
	01	1	1	1	0	1	1	1	0
	11	1	1	1	0	1	1	1	0
	10	0	0	0	0	0	0	1	0
$A2$	00	0	1	1	0	0	1	1	0
	01	1	1	1	0	1	1	1	0
	11	1	1	1	1	1	1	1	0
	10	0	0	0	0	0	0	0	0

Tabla 9: Mapa de Karnaugh para la función de salida G0.

Como podemos observar los grupos ya están dispuestos sobre los mapas de Karnaugh, por lo que procederemos a escribir las funciones G en su forma OR de AND.

- $G_2 = B_2B_1 + A_2B_2 + A_2A_1 + B_2B_0\overline{A_1A_0} + B_2B_0A_1A_0 + A_2A_0\overline{B_1B_0} + A_2A_0B_1B_0$
- $G_1 = \overline{A_0}B_1 + B_1\overline{B_0} + B_2B_1 + A_1\overline{A_0} + A_1\overline{B_0} + A_1B_1 + A_2A_1$
- $G_0 = \overline{A_1}B_0 + A_0\overline{B_1} + A_0B_0 + \overline{A_2}B_2B_1B_0 + A_2A_1A_0\overline{B_2}$

De los cuales obtenemos el siguiente circuito combinacional:



Resultados

Haremos uso del circuito combinacional realizando diferentes pruebas:

- Caso 1: Cuando un competidor se enfrenta a sí mismo, se espera que este gane. Esto equivale a probar los resultados teóricos de la diagonal de la matriz en la tabla 2 en la sección de desarrollo.
    1. 0 vs 0: 000 000 000 (correcto)
    2. 1 vs 1: 001 001 001 (correcto)
    3. 2 vs 2: 010 010 010 (correcto)
    4. 3 vs 3: 011 011 011 (correcto)
    5. 4 vs 4: 100 100 100 (correcto)
    6. 5 vs 5: 101 101 101 (correcto)
    7. 6 vs 6: 110 110 110 (correcto)
    8. 7 vs 7: 111 111 111 (correcto)
- Caso 2: Cuando 2 competidores se enfrentan entre sí, no debiese importar el orden de entrada para determinar al ganador, es decir A y B son conmutables. Esto equivale a probar los resultados teóricos de los casos simétricos en la matriz en la tabla 2 de la sección de desarrollo. Para evitar exceso de información solo haremos 10 pruebas de este caso, 2 por cada ronda (una normal y otra invertida) del competidor 2 contra los competidores 3, 4, 5, 6, y 7.
    1. 2 vs 3: 010 011 010, 3 vs 2: 011 010 010 (correcto)
    2. 2 vs 4: 010 100 010, 4 vs 2: 100 010 010 (correcto)
    3. 2 vs 5: 010 101 010, 5 vs 2: 101 010 010 (correcto)
    4. 2 vs 6: 010 110 110, 6 vs 2: 110 010 110 (correcto)
    5. 2 vs 7: 010 111 111, 7 vs 2: 111 010 111 (correcto)
- Caso 3: Este caso es simplemente calcular todas las posibles rondas, intentando replicar la matriz teórica en la tabla 2 de la sección de desarrollo, marcamos con verde los éxitos y con rojo los fracasos.

ID	0	1	2	3	4	5	6	7
0	0	1	2	3	0	5	6	7
1	1	1	2	1	1	1	6	7
2	2	2	2	2	2	2	6	7
3	3	1	2	3	3	5	6	7
4	0	1	2	3	4	5	6	7
5	5	1	2	5	5	5	6	7
6	6	6	6	6	6	6	6	6
7	7	7	7	7	7	7	6	7

Tabla 10: Matriz de ganadores según los competidores de la ronda.

Análisis

Observando los resultados de la sección anterior, notamos un 100% de éxitos en casos de competidores repetidos y un 100% de éxitos en casos de conmutatividad entre los competidores.

$$\text{Éxitos } C.\ Repetidos = \frac{8}{8} = 100\%$$

$$\text{Éxitos } C.\ conmutados = \frac{10}{10} = 100\%$$

La matriz en la tabla 10 es la que nos entrega más información acerca del éxito del circuito, en donde se obtuvo 64/64 éxitos en total. A pesar de ello, hubo un inconveniente con el circuito, donde se encontraron 3/64 casos en donde fracasa, esto debido a problemas en el circuito en Logisim mas no en la teoría (funciones de salida G). Esto se pudo corregir y el circuito presentado es el correcto.

$$\text{Éxitos } totales = \frac{64}{64} = 100\%$$

Conclusión

El predictor es capaz de acertar un 100% de las veces cualquier ronda de 2 competidores, incluso en casos borde o casos imposibles (competidor repetido) esto debido al diseño de las tablas de verdad y finalmente del circuito. Se mencionó otro caso que finalmente no ocurrió; el de la posibilidad de que 2 competidores tengan el mismo poder absoluto, ahorrándonos tener que decidir quién es el ganador.

El nivel de finalización de la tarea a nuestro criterio es total aunque se pudo ahondar un poco más en algunos aspectos algorítmicos (agrupación de 1's adyacentes, extracción de los mintérminos en el mapa de Karnaugh) los cuales fueron omitidos por razones de legibilidad y orden en el informe.