

## Seminario de Solucion de Problemas de Arquitectura de Computadoras

### Practica 4: Diseño con MUX

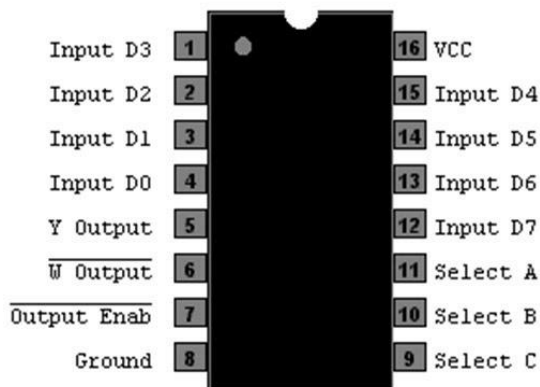
#### INTEGRANTES:

- Erick Martir Gonzalez
- Efrain Robles Pulido
- Carlos Manuel Hernandez Hernandez

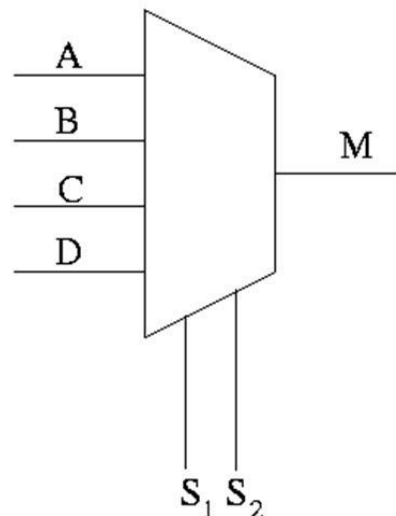
#### MATERIA:

- NRC: 153375
- Clave: I7024
- Sección: D11
- Docente: Garcia Hernandez Martin

# MULTIPLEXORES



Multiplexor (74LS151)



## OBJETIVO:

Comprobar de forma experimental el funcionamiento y utilidad de los MUX mediante la aplicación a un problema básico.

## INTRODUCCIÓN:

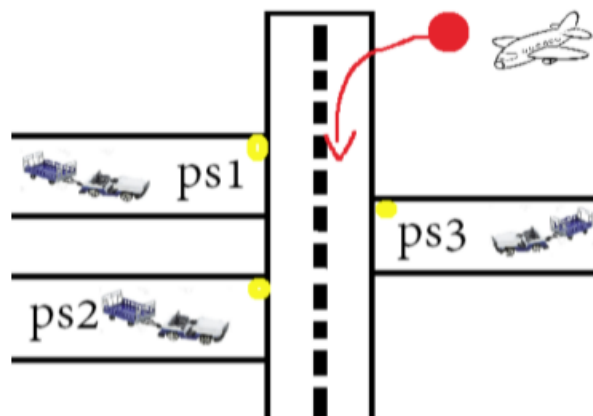
Los multiplexores son circuitos combinacionales con varias entradas de control capaces de seleccionar una, y solo una, de las entradas de datos para permitir su transmisión desde la entrada seleccionada hacia dicha salida.

En el campo de la electrónica el multiplexor se utiliza como dispositivo que puede recibir varias entradas y transmitir las por un medio de transmisión compartido.

## DESARROLLO:

### 1. Diseño de un control simple de tráfico.

En la Figura 1 muestra una entrada a un andén con 3 pistas secundarias de servicios: ps1, ps2 y ps3. Cada pista de servicio tiene un sensor que enciende si el carro maletero está en la pista principal de acceso al andén. La pista principal de acceso permite la entrada de un vuelo si hay solamente un carro de servicio o ninguno. En caso contrario no se permitiría el acceso al andén.



**Figura 1:** Entrada a un andén con pistas de servicio

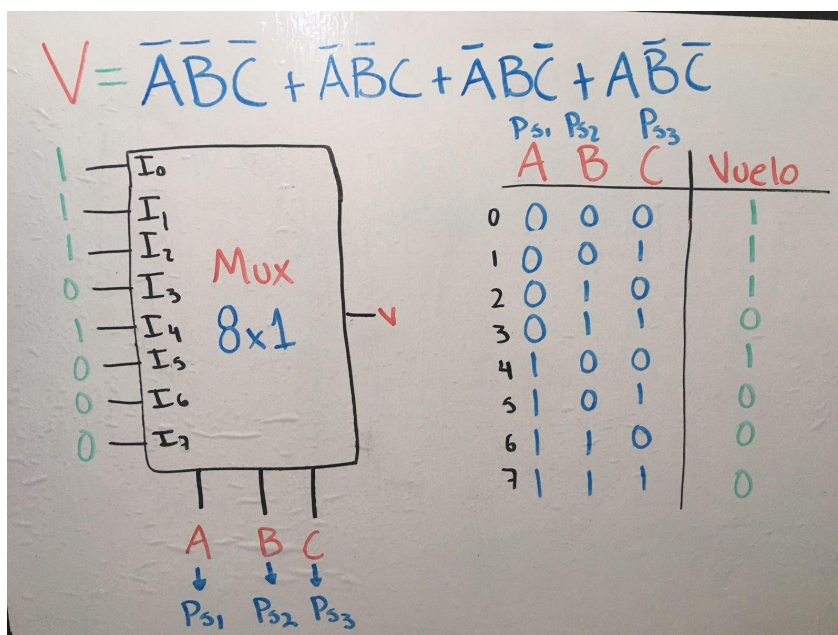
A) Escribimos la tabla de verdad y la función suma en el siguiente formato:  $F(s) = (....)$

	A	B	C	
	ps1	ps2	ps3	Vuelo
0	0	0	0	1
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	0

**Tabla 1:** Tabla de verdad para describir el comportamiento deseado del problema, en donde si hay al menos un sensor activado o ninguno de los tres sensores, se permitirá la entrada de un vuelo.

**Función:**  $f(A, B, C) = \sum(0, 1, 2, 4)$

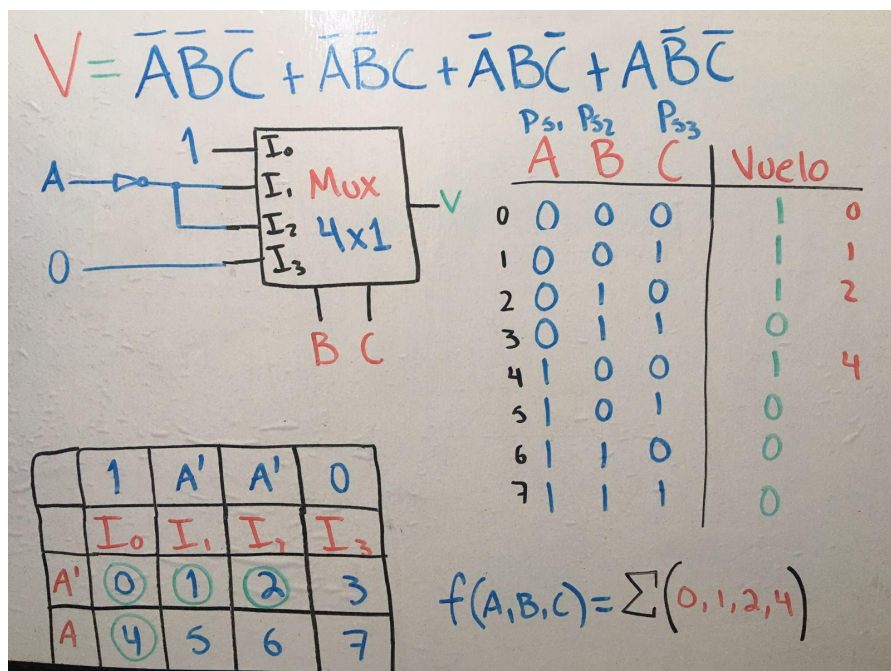
B) Se Implementó la función  $f(A, B, C) = \sum(0, 1, 2, 4)$  en un MUX 8 x 1



**Figura 2:**

Implementación de un Multiplexor de 8 x 1, en base a las salidas de la tabla de verdad y utilizando un cambio de variable (A, B, C) para diferenciar con más facilidad los términos de entrada de la ecuación resultante, en base a los sensores de la pista de servicio.

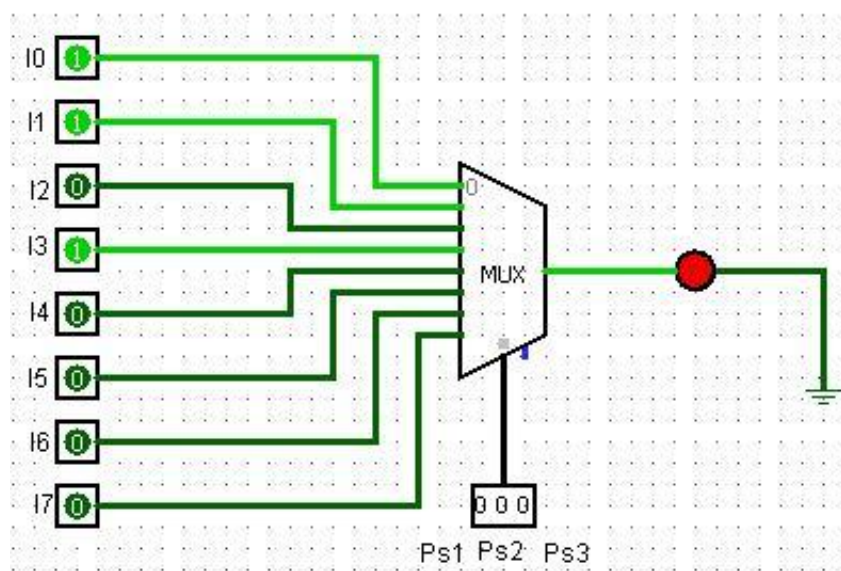
**C)** Se hizo la reducción de la función  $f(A, B, C) = \Sigma(0, 1, 2, 4)$  a un MUX de  $4 \times 1$



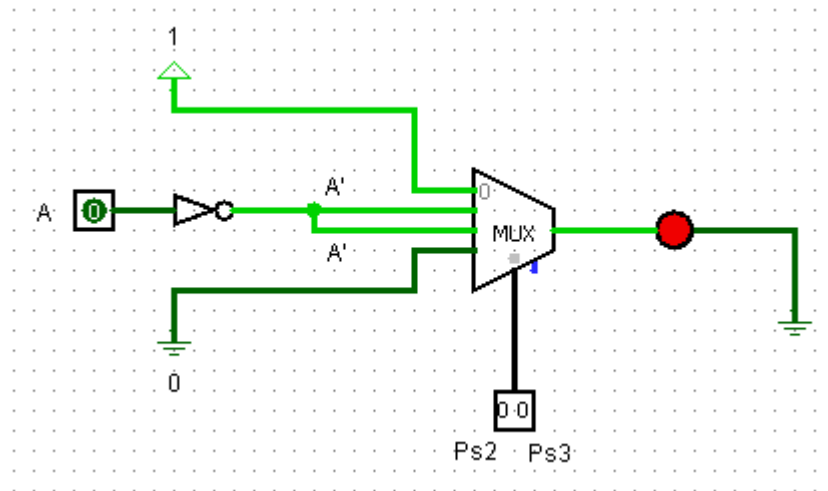
**Figura 3:**

Implementación de un Multiplexor de  $4 \times 1$  con su respectiva tabla de verdad para hacer la reducción, todo en base a los minterminos, haciendo uso del mapa de reducción de multiplexores para obtener los valores en cada una de las entradas resultantes de nuestro MUX de  $4 \times 1$ .

**D)** Se construyó los multiplexores en el simulador:



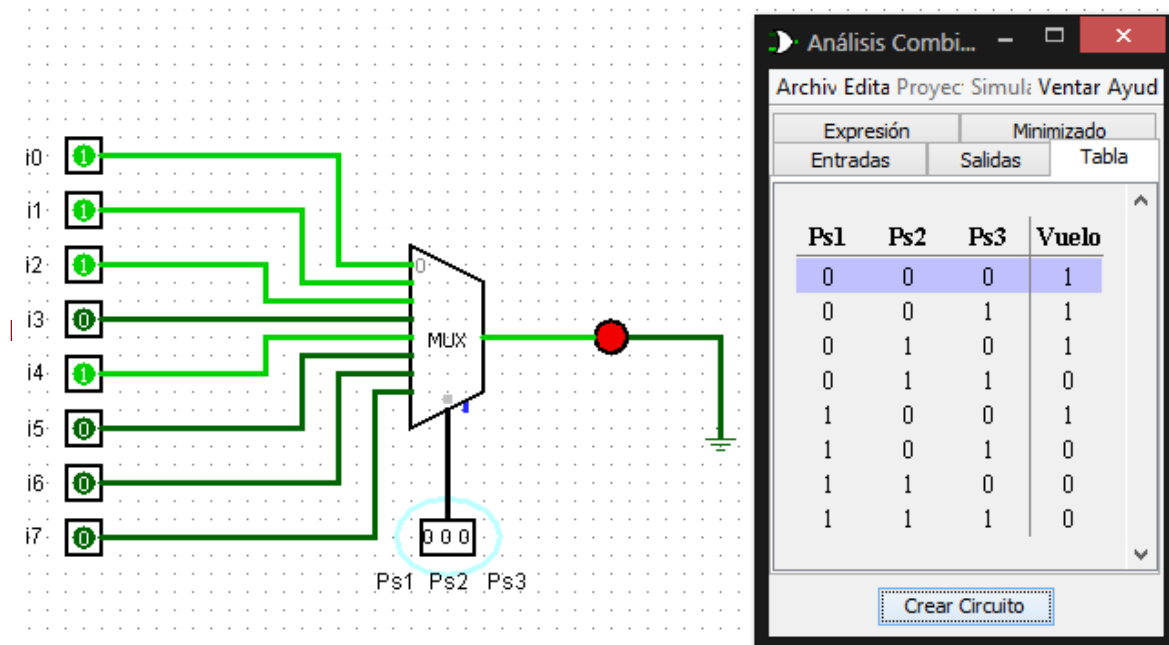
**Figura 4:** . Circuito del MUX  $8 \times 1$ . En donde en sus entradas se le introducirá los valores de salida de tabla de verdad (vuelo), que se verá a continuación.



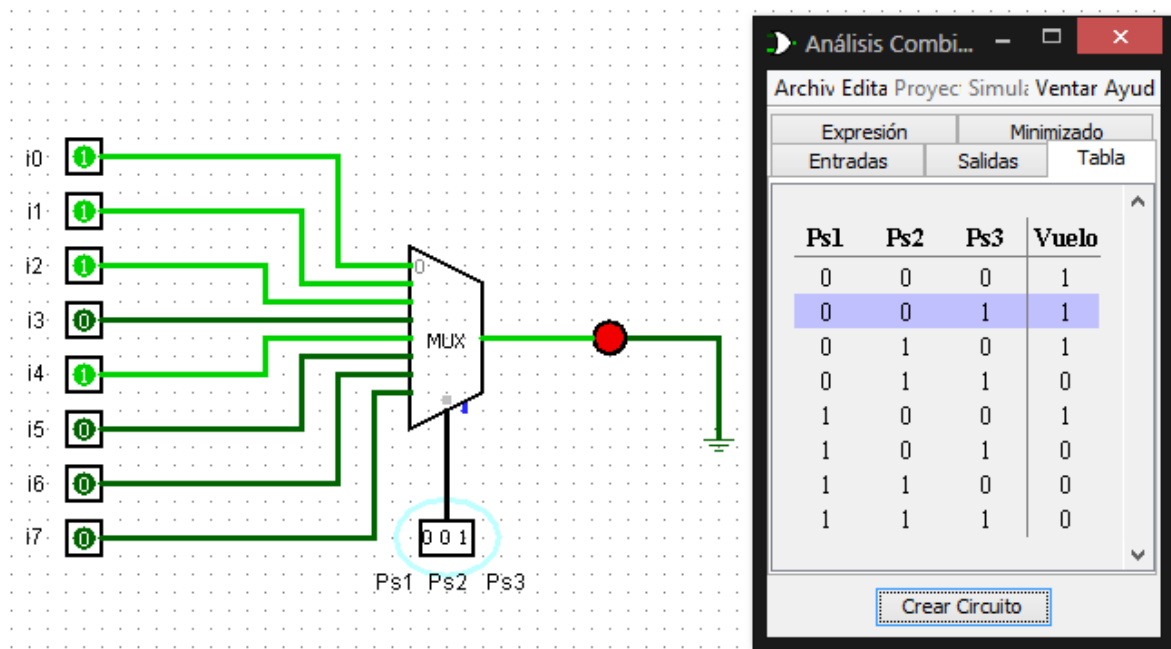
**Figura 5:** . Circuito del MUX 4 x 1. En donde en sus entradas se le introducirá los valores encontrados en el mapa de reducción de los multiplexores,  $i_0=1, i_1=i_2=Ps_1' (A'), i_3=0$ .

**E)** Se verificaron todos y cada uno de los estados en cada multiplexor, tanto el de la **Figura 4** (8 x 1) como el de la **Figura 5** (4 x 1). Se comprobó que, tanto en el cambio de los estados del MUX 8 x 1 como el del MUX 4 x 1 reducido, son exactamente iguales en su salida y mediante cada combinación también se comprobó que son iguales.

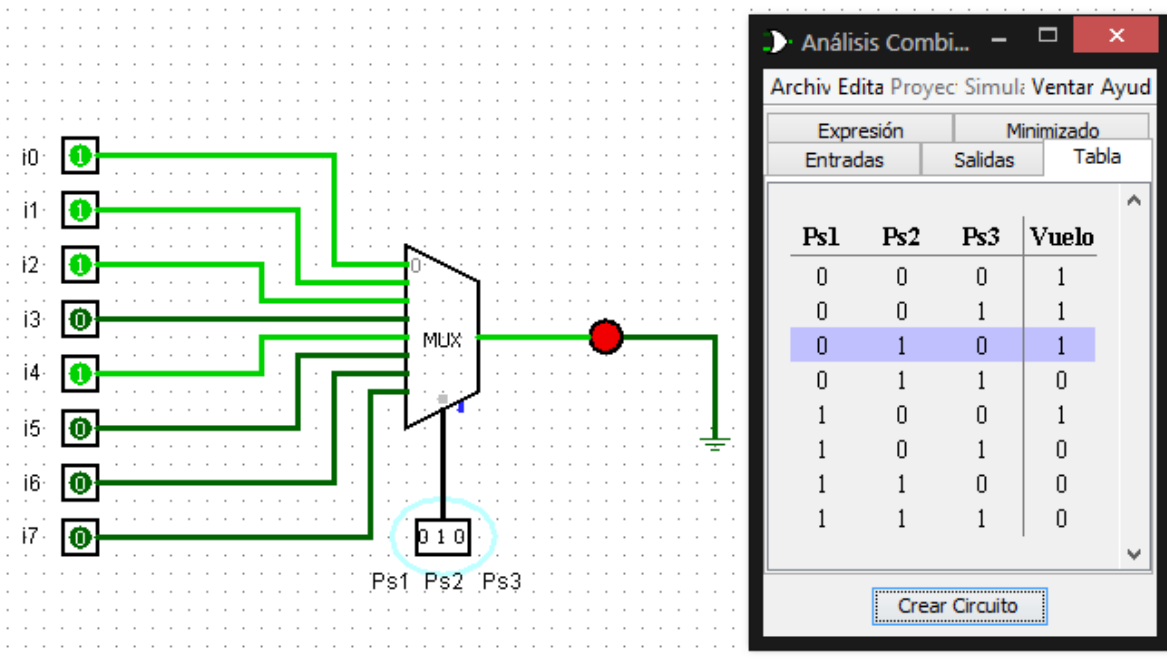
- Estados de Multiplexor 8 x 1:



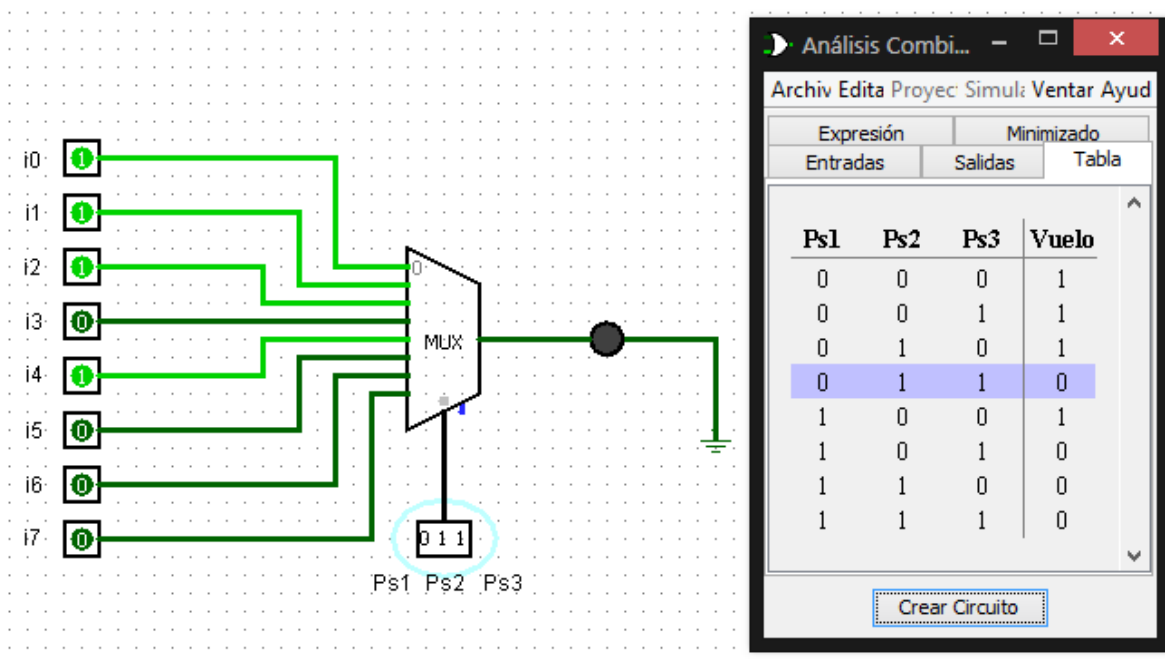
**Figura 4.1:** Cuando los tres estados están en  $Ps_1=0, Ps_2=0, Ps_3=0$  permite el aterrizaje de un avión (se encenderá el led).



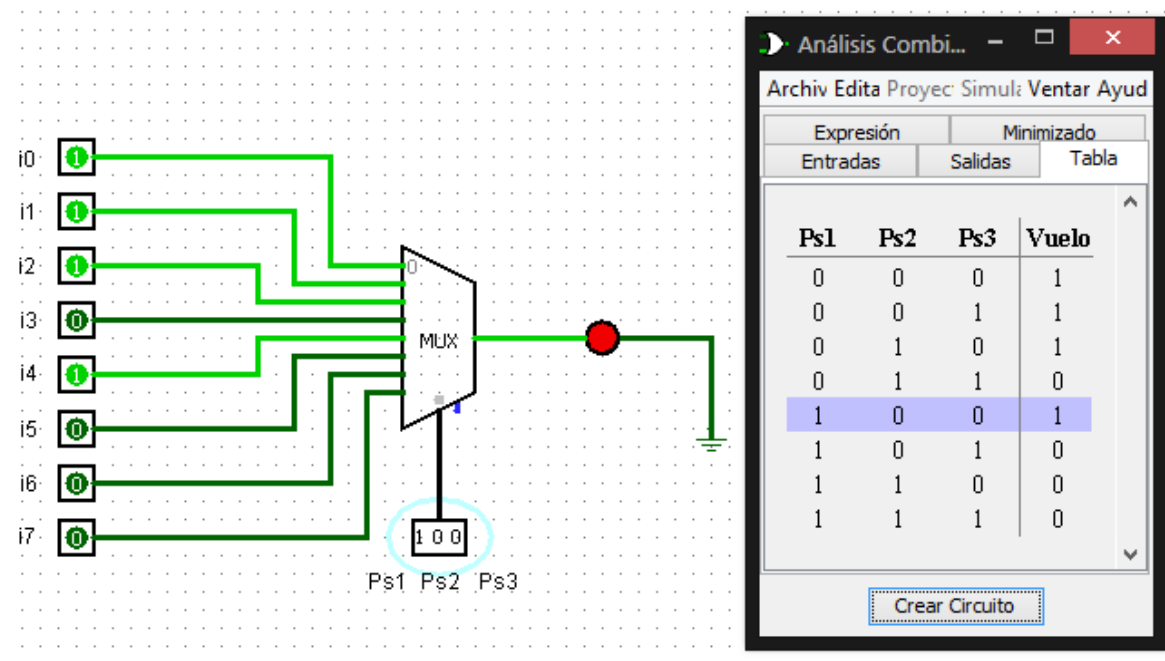
**Figura 4.2:** Cuando los tres estados están en  $Ps1=0$ ,  $Ps2=0$ ,  $Ps3=1$  permite el aterrizaje de un avión (se encenderá el led).



**Figura 4.3:** Cuando los tres estados están en  $Ps1=0$ ,  $Ps2=1$ ,  $Ps3=0$  permite el aterrizaje de un avión (se encenderá el led).

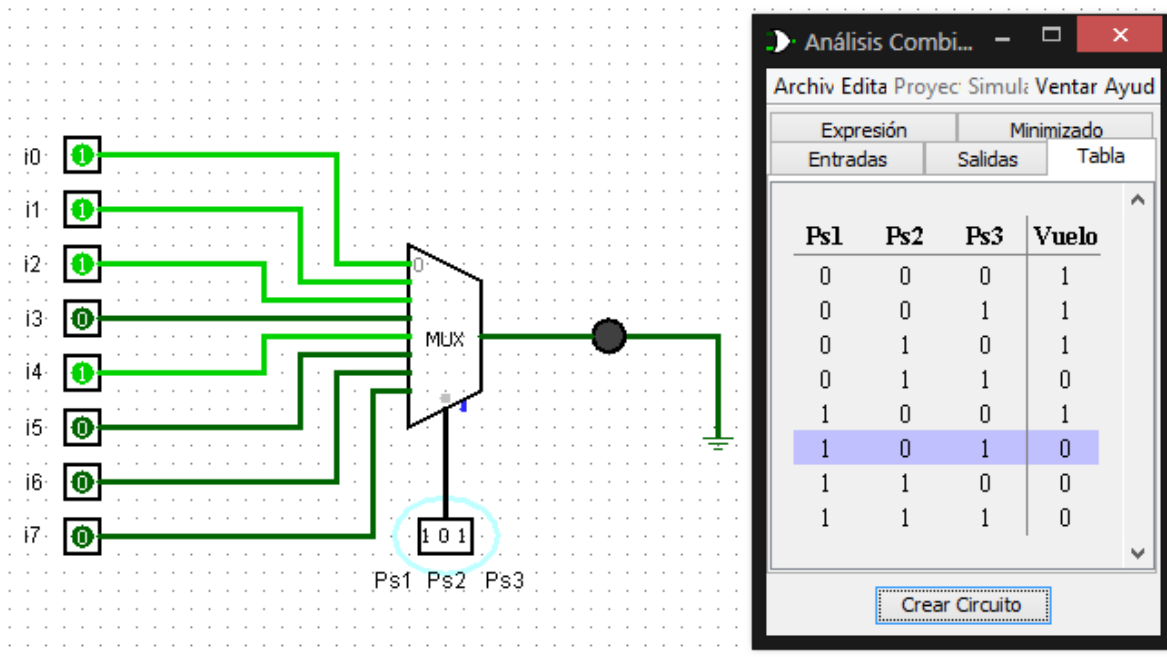


**Figura 4.4:** Cuando los tres estados están en  $Ps1=0, Ps2=1, Ps3=1$  no permite el aterrizaje de un avión ( y por lo tanto no se encenderá el led).

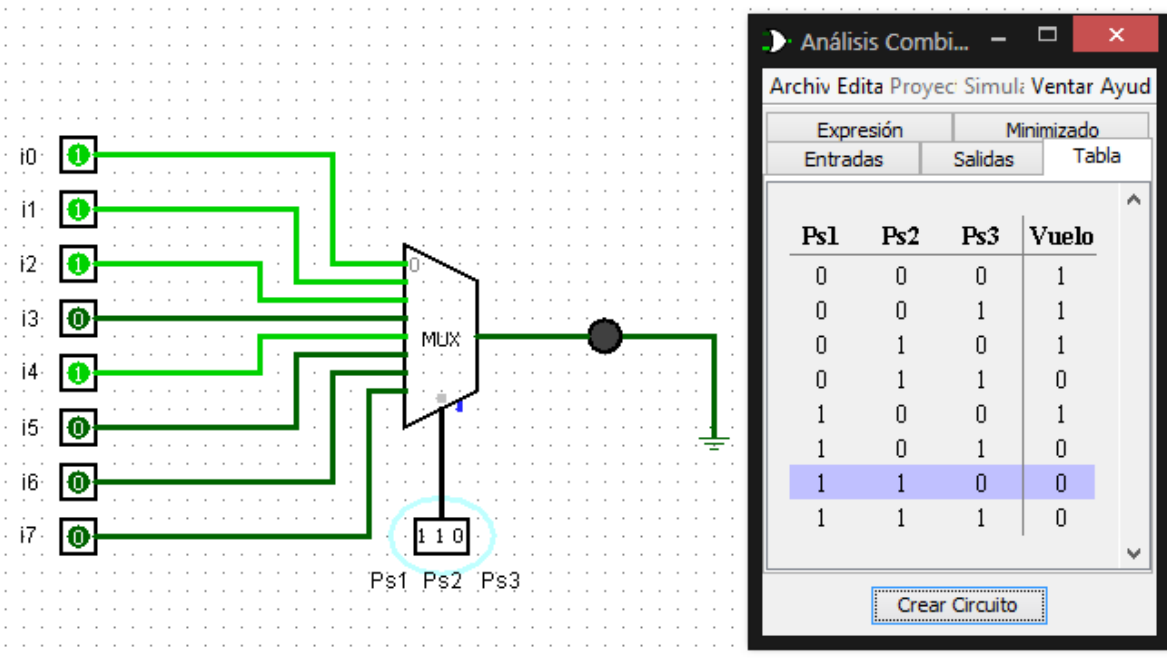


**Figura 4.5:** Cuando los tres estados están en  $Ps1=, Ps2=0, Ps3=0$  permite el aterrizaje de un avión (se encenderá el led).



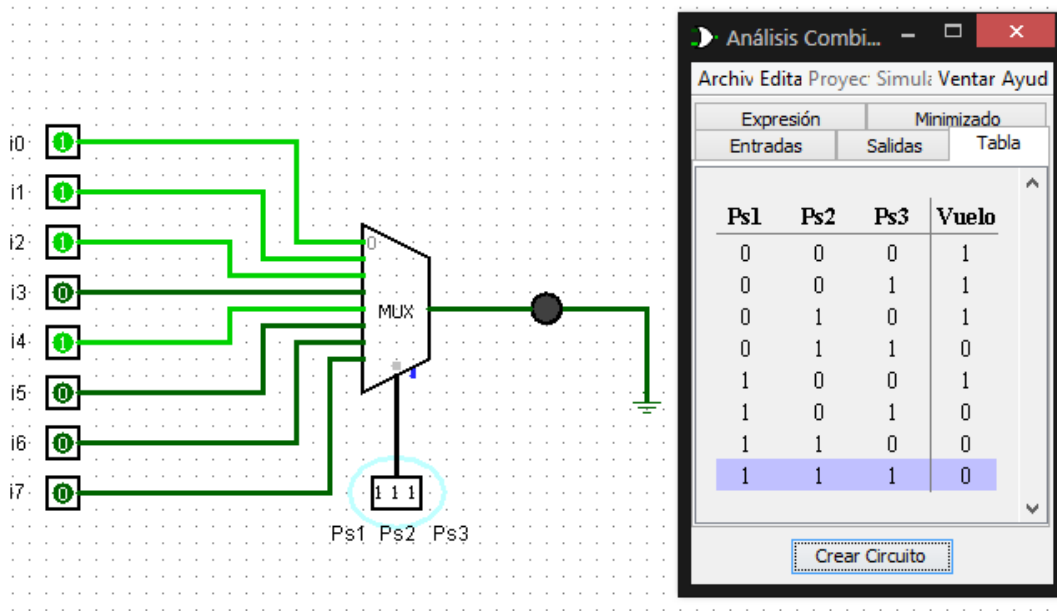


**Figura 4.6:** Cuando los tres estados están en Ps1=1, Ps2=0, Ps3=1 no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).



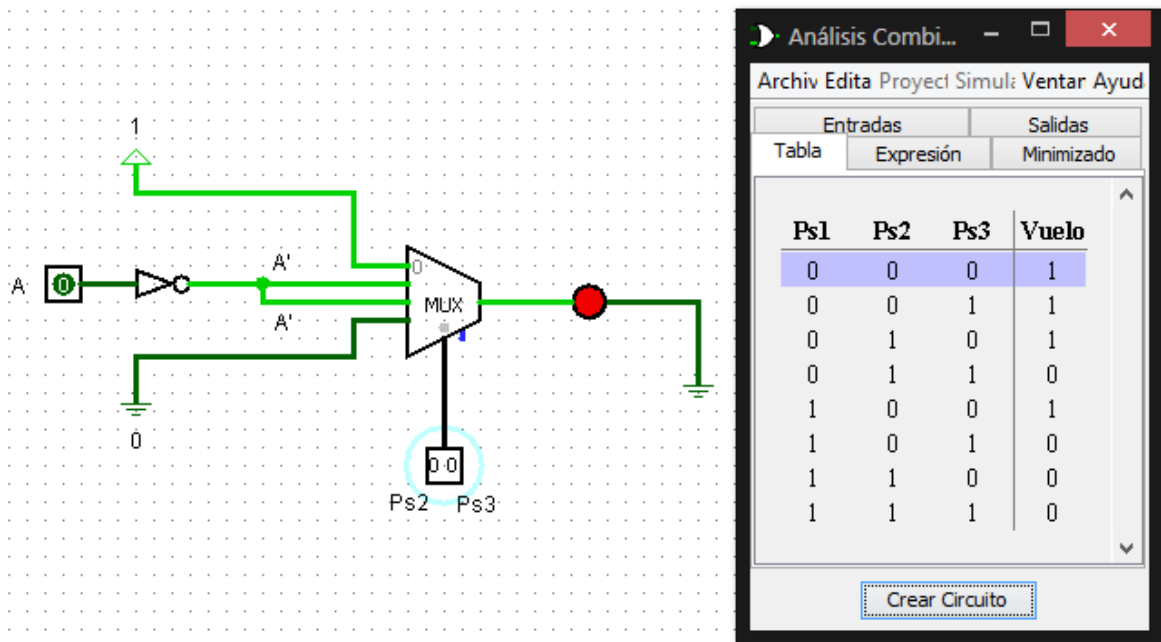
**Figura 4.7:** Cuando los tres estados están en Ps1=1, Ps2=1, Ps3=0 no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).



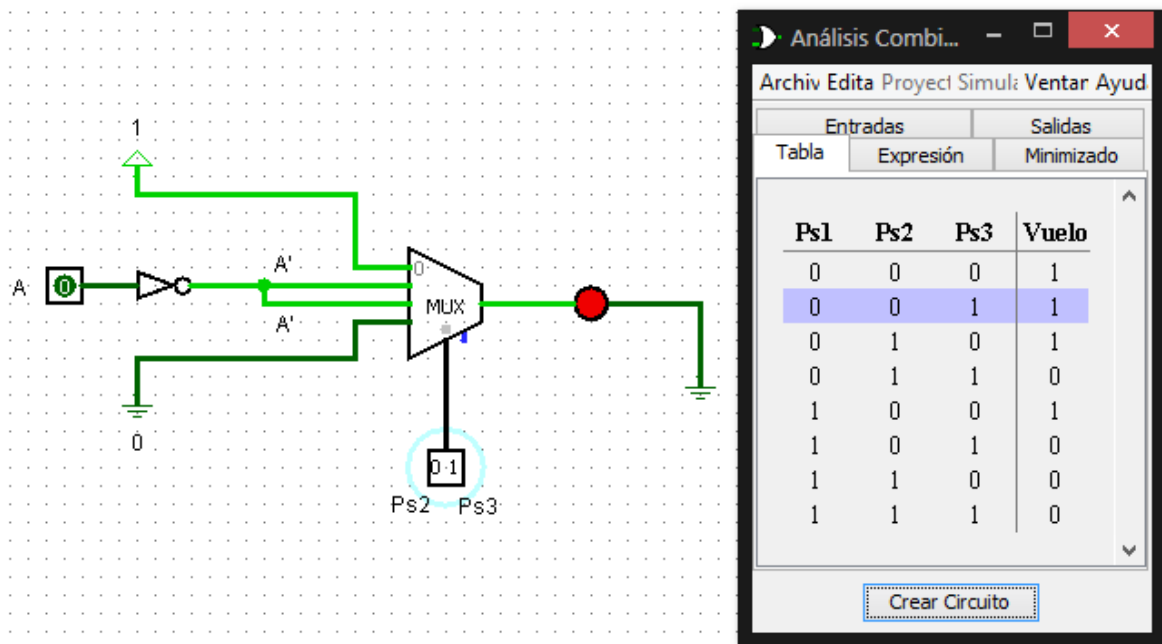


**Figura 4.8:** Cuando los tres estados están en Ps1=1, Ps2=1, Ps3=1 no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).

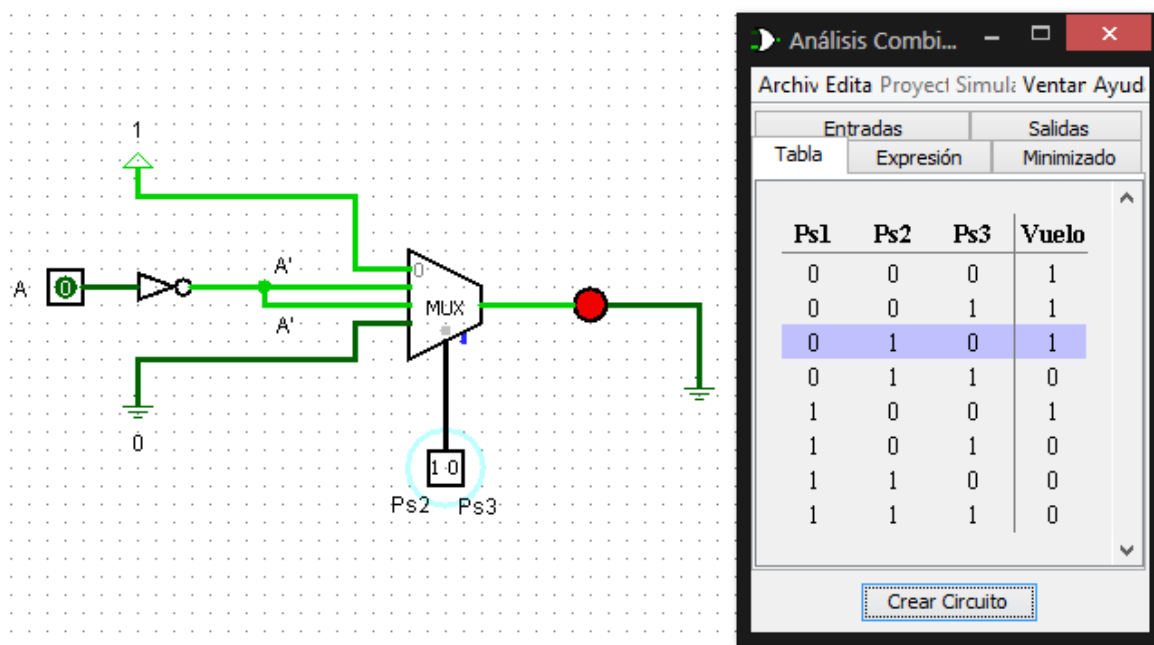
- Estados de Multiplexor 4 x 1:



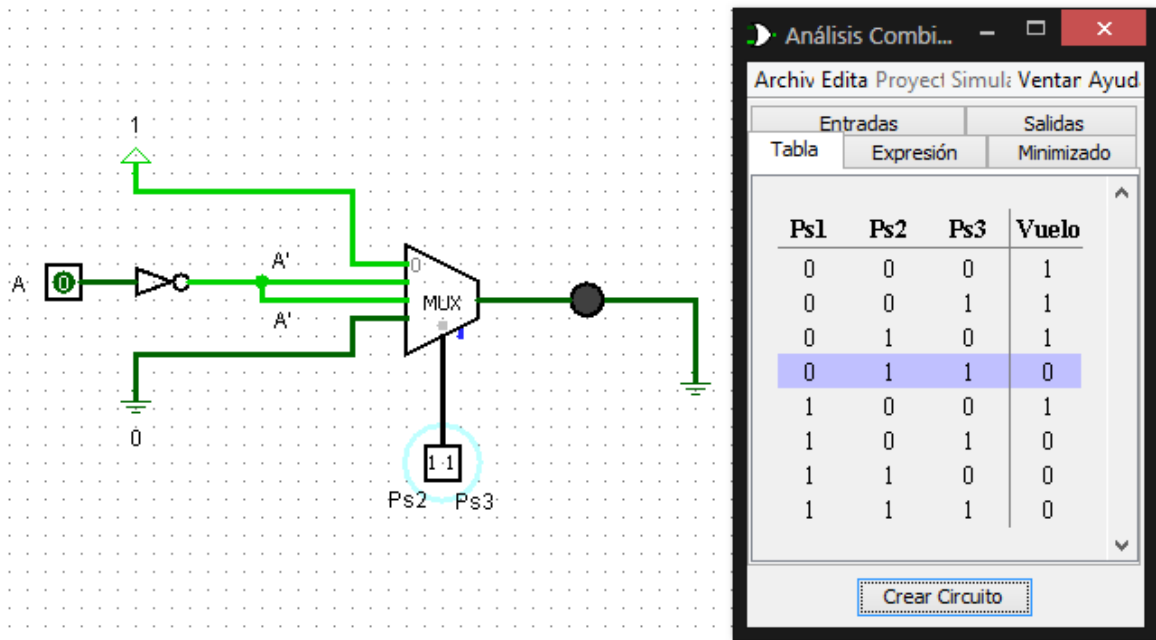
**Figura 5.1:** Cuando los tres estados están en A(Ps1)=0, Ps2=0, Ps3=0 permite el aterrizaje de un avión (se encenderá el led).



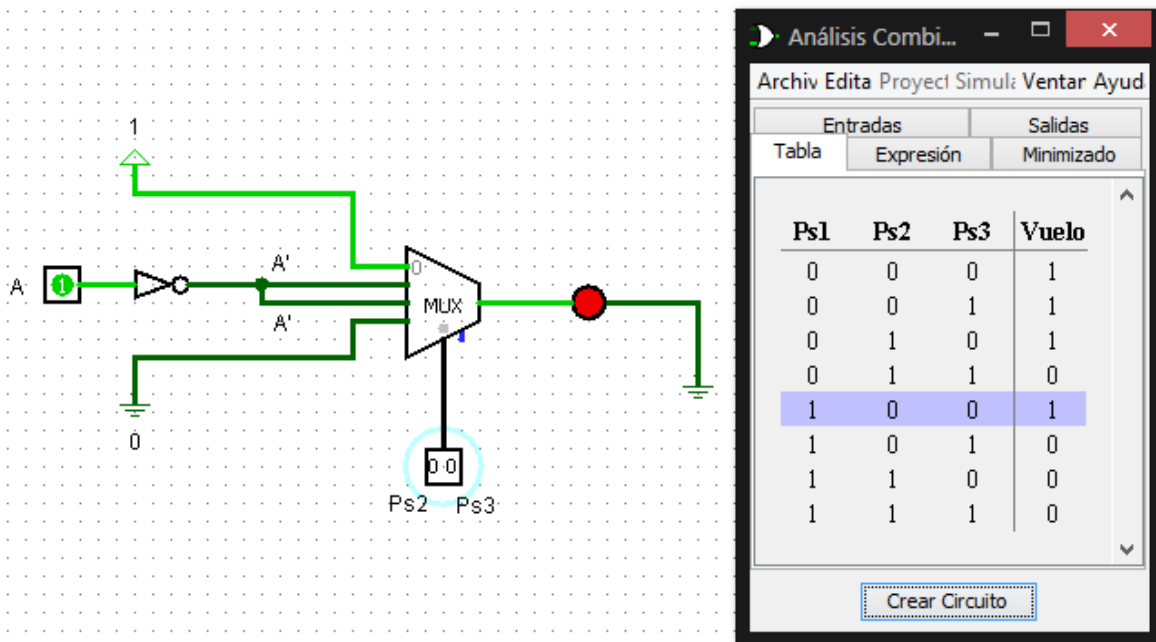
**Figura 5.2:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=0$ ,  $Ps3=1$  permite el aterrizaje de un avión (se encenderá el led).



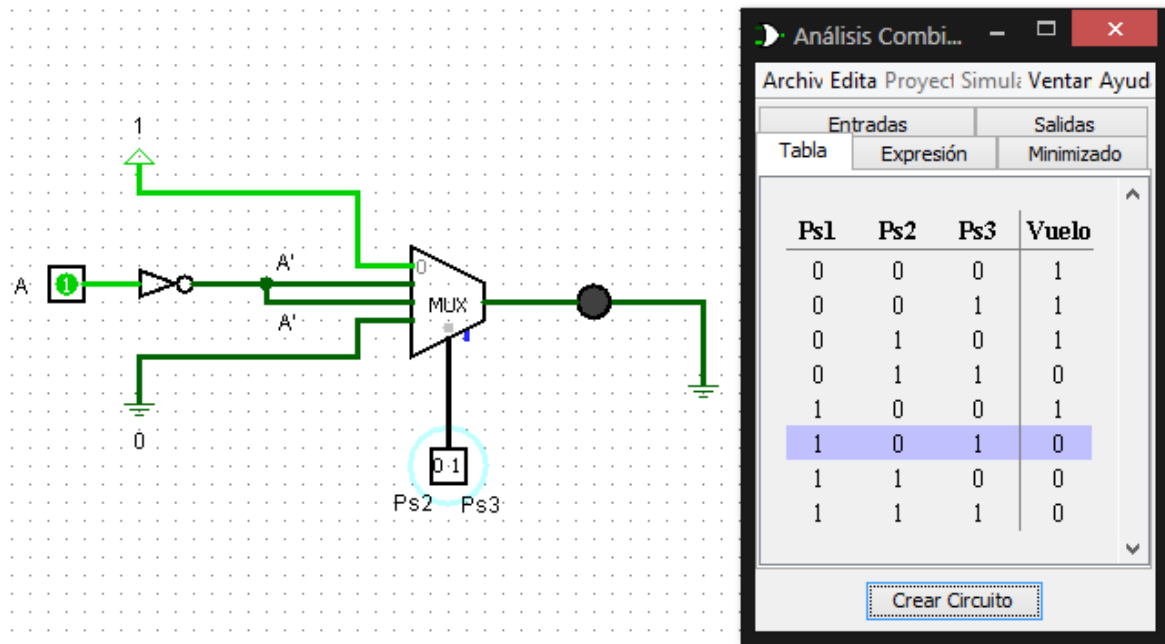
**Figura 5.3:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=1$ ,  $Ps3=0$  permite el aterrizaje de un avión (se encenderá el led).



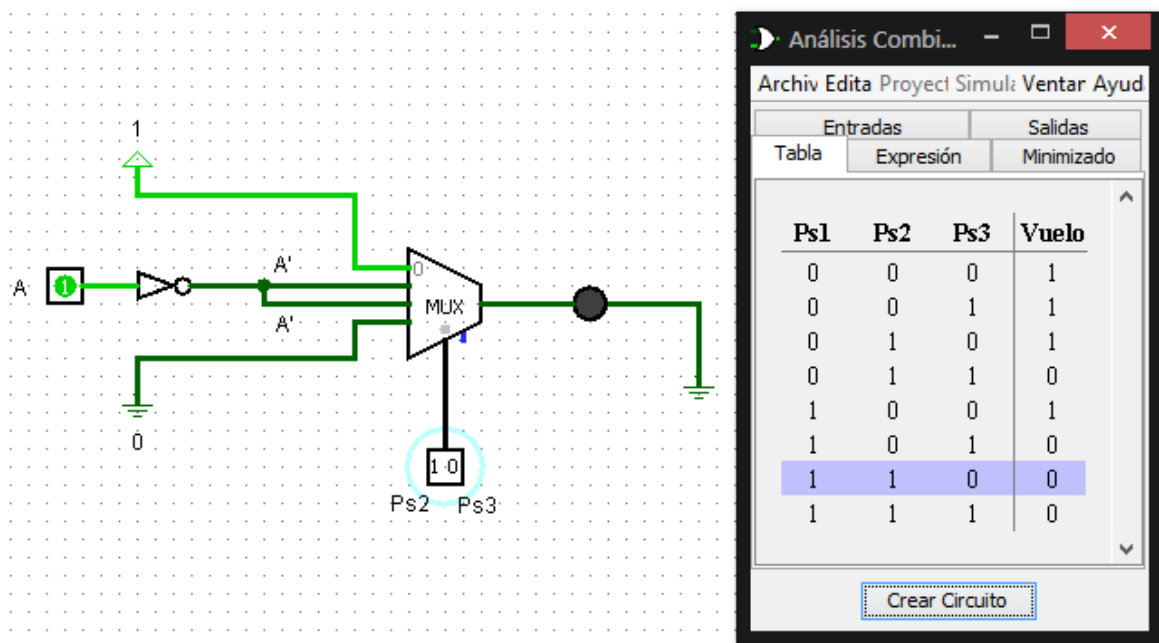
**Figura 5.4:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=1$ ,  $Ps3=1$  no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).



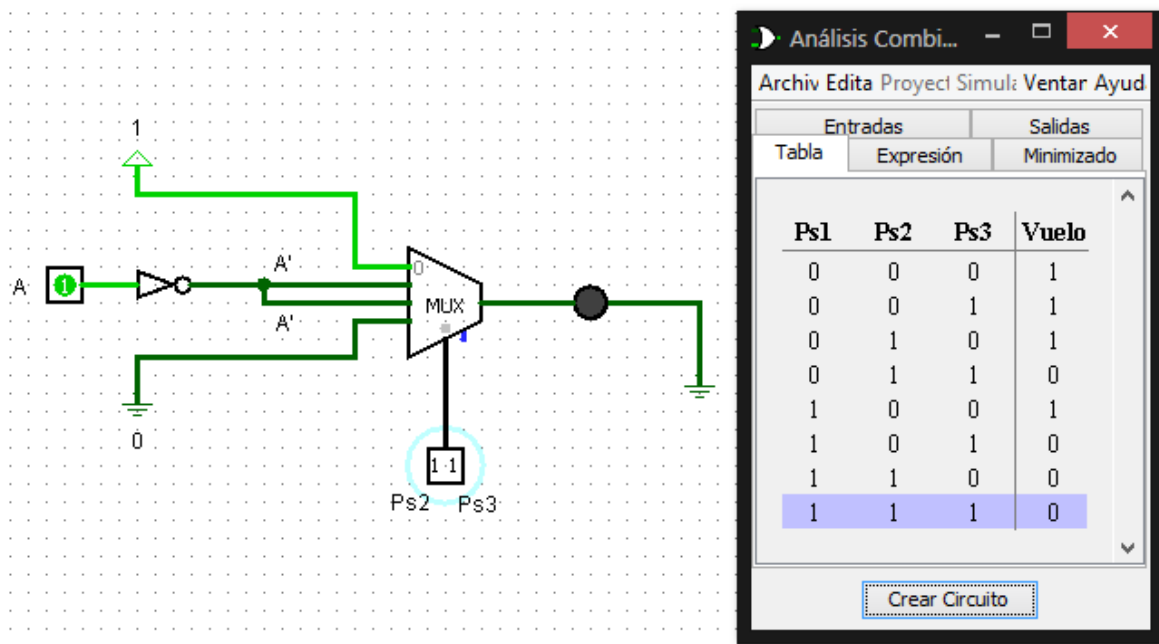
**Figura 5.5:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=0$ ,  $Ps3=0$  permite el aterrizaje de un avión (se encenderá el led).



**Figura 5.6:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=0$ ,  $Ps3=1$  no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).



**Figura 5.7:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=1$ ,  $Ps3=0$  no permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).



**Figura 5.8:** Cuando los tres estados están en  $A(Ps1)=0$ ,  $Ps2=1$ ,  $Ps3=1$  no se permitirá el aterrizaje de un avión (y por lo tanto no se encenderá el led).

### RESULTADOS:

En la tabla de verdad nos dimos cuenta cuáles fueron las que sí se permitirá la entrada del vuelo y las que no, cumpliendo con los requerimientos de un control simple de tráfico, en donde se conectó a un MUX 8x1 para hacer la reducción a un MUX 4x1 donde obtuvimos el mismo comportamiento en ambos MUX, permitiéndonos utilizar un MUX con menos entradas posibles.

## DISCUSIÓN Y CONCLUSIONES:

**Erick:** Esta práctica me fue muy interesante, ya que el uso de un Multiplexor nos permite obtener una selección de datos de una manera más eficiente y que también podemos reducir mediante un mapa y en base a la regla para la reducción de los MUX, obtener un multiplexor más pequeño y así obtener los mismos resultados en sus combinaciones de estados. Dicho así, me fue de gran utilidad aprender este tema, y con esto, puedo concluir que el uso y la implementación de los multiplexores es muy interesante, ya que un MUX nos ayuda a recibir múltiples entradas y transmitir las a una única salida, y con esto podemos dividir el medio de transmisión en múltiples canales para que varios de estos puedan comunicarse al mismo tiempo.

**Carlos:** Esta práctica se me hizo muy padre ya que con ayuda de mis compañeros aprendí a hacer el MUX de 8x1 y 4x1 y vi que solamente cuando hay una sola entrada 1 hay un vuelo y vi que también con el MUX de 4x1 se saca la sumatoria de productos, en las prácticas poco a poco puedo ir simulando los circuitos y ya no se me hace tan complicado. Vimos que el MUX nos puede servir para conmutar entradas múltiples y con sola una salida, a través de las diferentes combinaciones podemos ver cuáles son las entradas que si nos dan salidas 1 y también vemos cuáles nos dan salidas 0.

**Efrain:** En esta práctica puede aprender cómo podemos reducir las entradas a utilizar de un MUX a otro MUX con menos entradas que el inicial, demostrando así que el comportamiento de la salida es la misma que los MUXs, también vi cómo podemos aplicar estos conocimientos de la lógica para solucionar problemas de la vida. Fue sencillo de realizar la reducción con la ayuda de las reglas del MUX para encontrar la combinación de entradas en el MUX reducido. Será útil utilizar este método de reducción del multiplexor para el diseño de circuitos lógicos para aparatos electrónicos, en donde buscaremos una salida determinada según el caso de las variables de selección del multiplexor.

## REFERENCIA:

Multiplexores. Recuperado el 24 de Septiembre de 2021, de <https://sites.google.com/site/serviciodelainformatica/multiplexores>