

Facultad de Ingeniería - UNLP

E1301 – E0301 Introducción a los Sistemas Lógicos y Digitales
Curso 2025 - Trabajo Entregable 1

Isabella Valenzi Octavio Tomás - 03665/8

Rodriguez Fausto - 03766/2

Cladera Blas - 03694/3

Patané Baltazar - 03773/1

[Repositorio de GitHub del grupo](#)

Introducción

En el siguiente informe se explicará el proceso de creación del circuito lógico y eléctrico capaz de llevar a cabo el manejo de prioridades planteado.

Desarrollo

A partir de la consigna propuesta ideamos un circuito C, con 17 entradas y 14 salidas:

- 16 entradas corresponden a los 16 sensores (indicados con S_n según el número de sensor).
- 1 entrada corresponde al bit KA, que indica el tipo de display con el que se va a trabajar. ($KA=0 \rightarrow$ cátodo común, $KA=1 \rightarrow$ ánodo común).
- 7 salidas corresponden a los 7 leds del Display que representa las unidades del número a mostrar.
- 7 salidas corresponden a los 7 leds del Display que representa las decenas del número a mostrar.

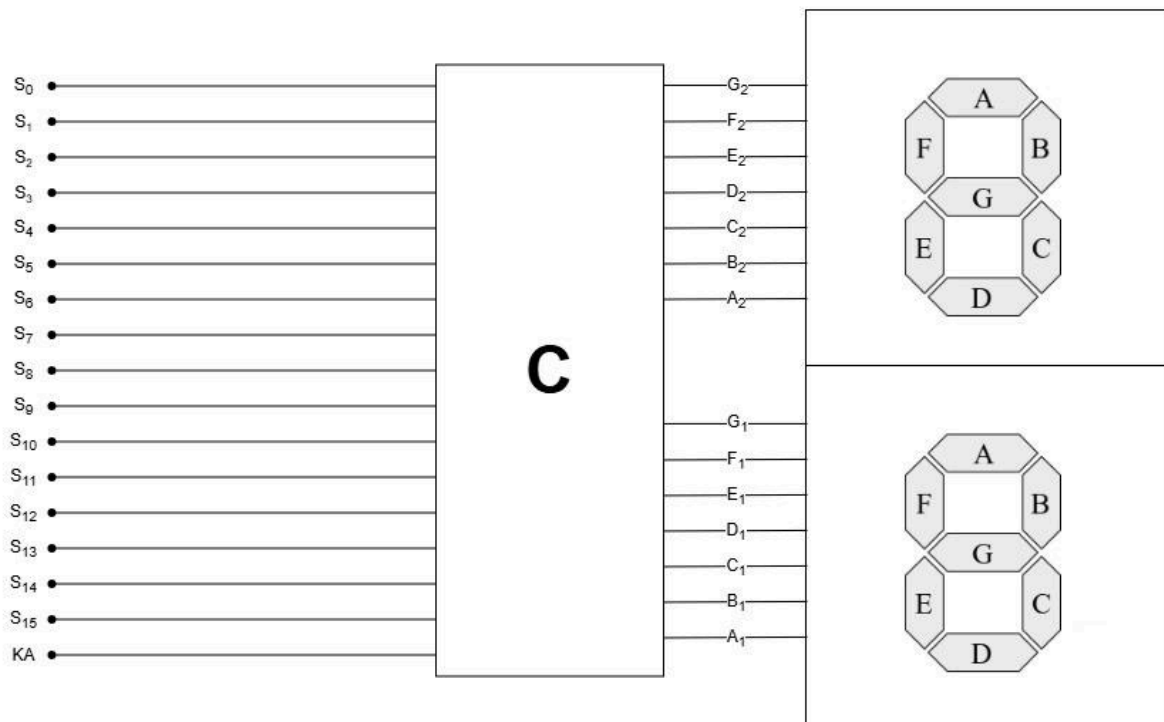


Fig. 1: Representación gráfica del circuito C.

Habiendo aclarado las entradas y salidas de nuestro circuito C, podemos entrar más en detalle en qué hay dentro del circuito C:

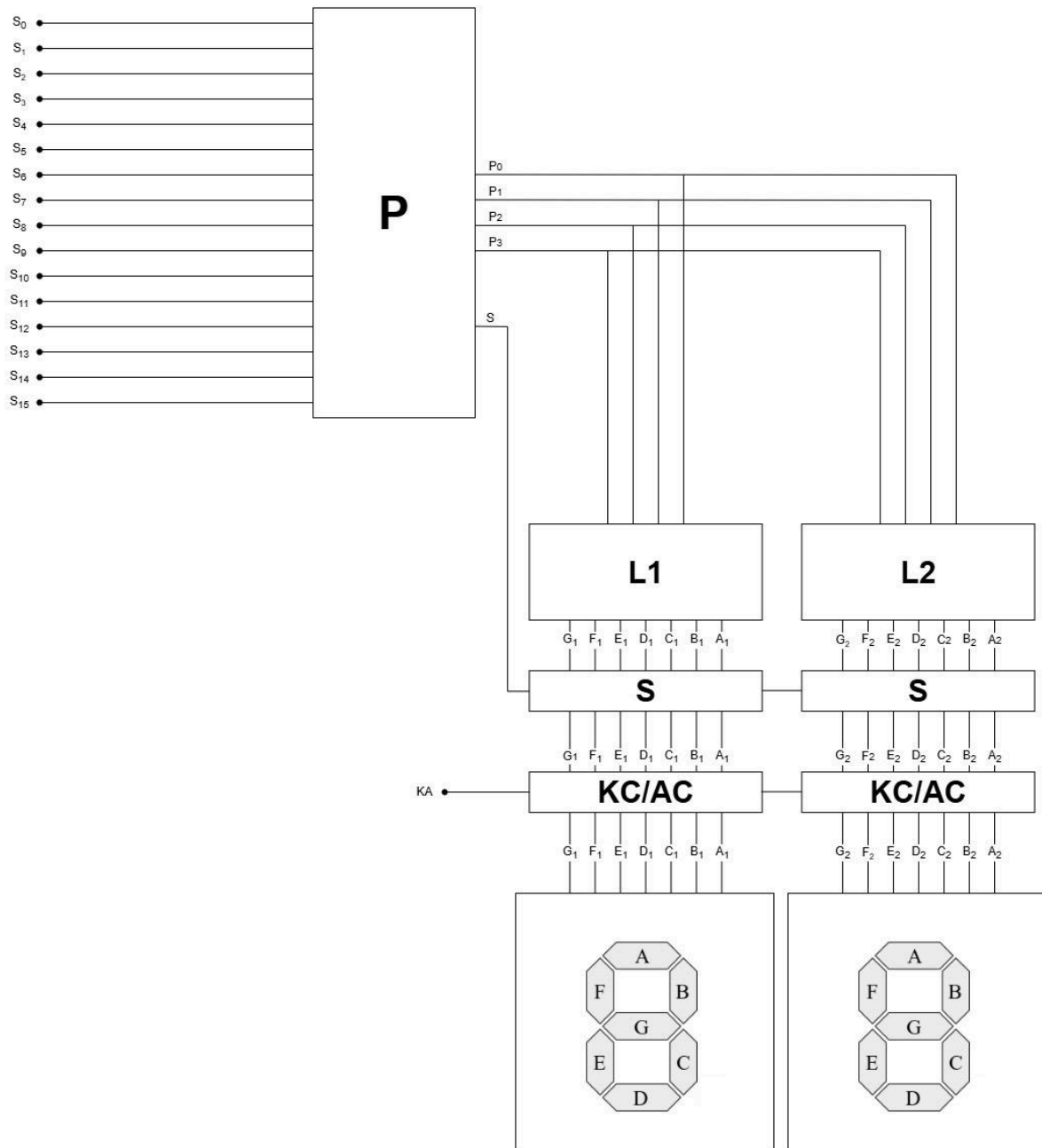


Fig. 2: Diagrama de bloques del circuito.

Bloque P

Este primer bloque que llamamos “P” (porque es el que trabaja con las prioridades de los sensores) será el encargado de decodificar una entrada de hasta 16 sensores en una salida de 4 bits que representan el número de sensor de mayor prioridad.

Para armar la tabla de verdad de este circuito, y a partir de ella obtener su circuito de compuertas lógicas hicimos el siguiente razonamiento:

Si el sensor 15 está encendido, no importa si el resto de sensores están encendidos o no, la salida será 15 dado que es el sensor de mayor prioridad. Si el sensor 14 está encendido, la salida será 14 siempre y cuando el sensor 15 esté apagado. Si el sensor 13 está encendido, la salida será 13 siempre y cuando los sensores 15 y 14 estén apagados.

Seguindo esta lógica para todos los valores construimos la siguiente tabla de verdad:

S15	S14	S13	S12	S11	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1	S0		P3	P2	P1	P0	S
0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X		1	1	1	1	0
1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X		1	1	1	0	0
1	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X		1	1	0	1	0
1	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X		1	1	0	0	0
1	1	1	1	0	X	X	X	X	X	X	X	X	X	X	X		1	0	1	1	0
1	1	1	1	1	0	X	X	X	X	X	X	X	X	X	X		1	0	1	0	0
1	1	1	1	1	1	0	X	X	X	X	X	X	X	X	X		1	0	0	1	0
1	1	1	1	1	1	1	0	X	X	X	X	X	X	X	X		1	0	0	0	0
1	1	1	1	1	1	1	1	0	X	X	X	X	X	X	X		1	0	0	0	0
1	1	1	1	1	1	1	1	1	0	X	X	X	X	X	X		0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	0	X	X	X	X	X		0	1	1	0	0
1	1	1	1	1	1	1	1	1	1	0	X	X	X	X	X		0	1	0	1	0
1	1	1	1	1	1	1	1	1	1	1	0	X	X	X	X		0	1	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	0	X	X	X		0	0	1	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	0	X	X		0	0	1	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	X		0	0	0	1	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0		0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1		0	0	0	0	1

Fig. 3: Tabla de verdad del circuito P.

Se puede observar que se agregó una salida adicional, el bit S (de signo, aunque no trabaja directamente con el signo de los números). Al idear el circuito llegamos a un problema: el circuito tiene 17 posibles resultados:

- 16 salidas corresponden al número de prioridad, desde el 0 hasta al 15.
- 1 salida corresponde a la situación en la que ningún sensor está encendido.

Como estamos trabajando con 17 salidas como mínimo se necesitan 5 bits para poder representar este número de combinaciones. Por esto el bit S solo vale 1 en el caso donde todos los sensores están apagados.

A partir de la tabla de verdad del circuito P usando el método de suma de productos construimos el circuito de compuertas lógicas de P:

[Fig. 4: Circuito lógico P.](#)

Bloque L1

El bloque L1 (L de led) usará la salida ya codificada por P para interpretar cuáles leds del display de 7 segmentos deben encenderse según el número recibido. Este bloque es el encargado de las decenas del número a representar, por lo que se presentan 2 casos:

- Números del 0 al 9: el display debe mostrar el número "0".
- Números del 10 al 15: el display debe mostrar el número "1".

Teniendo en cuenta lo anterior armamos la siguiente tabla de verdad:

#	P3	P2	P1	P0	G	F	E	D	C	B	A
0	0	0	0	0	0	1	1	1	1	1	1
1	0	0	0	1	0	1	1	1	1	1	1
2	0	0	1	0	0	1	1	1	1	1	1
3	0	0	1	1	0	1	1	1	1	1	1
4	0	1	0	0	0	1	1	1	1	1	1
5	0	1	0	1	0	1	1	1	1	1	1
6	0	1	1	0	0	1	1	1	1	1	1
7	0	1	1	1	0	1	1	1	1	1	1
8	1	0	0	0	0	1	1	1	1	1	1
9	1	0	0	1	0	1	1	1	1	1	1
10	1	0	1	0	0	0	0	0	1	1	0
11	1	0	1	1	0	0	0	0	1	1	0
12	1	1	0	0	0	0	0	0	1	1	0
13	1	1	0	1	0	0	0	0	1	1	0
14	1	1	1	0	0	0	0	0	1	1	0
15	1	1	1	1	0	0	0	0	1	1	0

Fig. 5: Tabla de verdad del circuito L1.

Con esta tabla de verdad hicimos el mapa de Karnaugh de cada una de las salidas de la G a la A, cada salida representando el estado encendido/apagado de los leds.

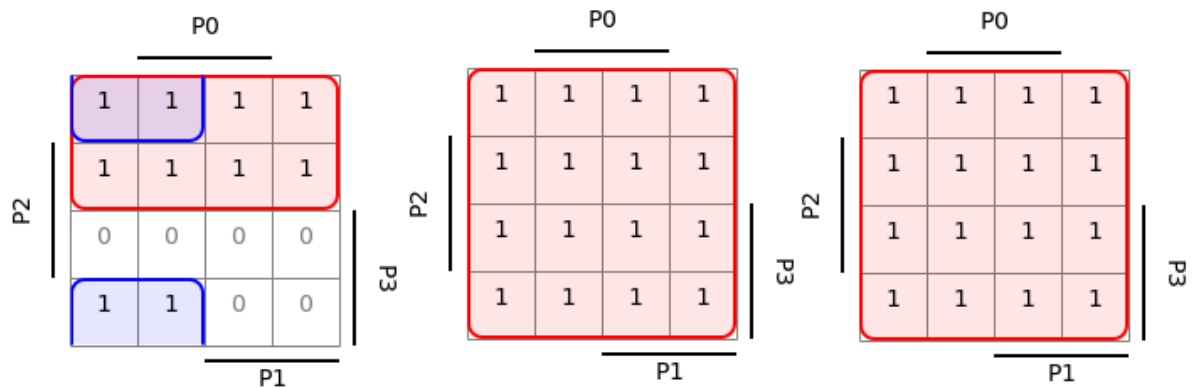


Fig. 6a: K-Map salidas A, B y C respectivamente.

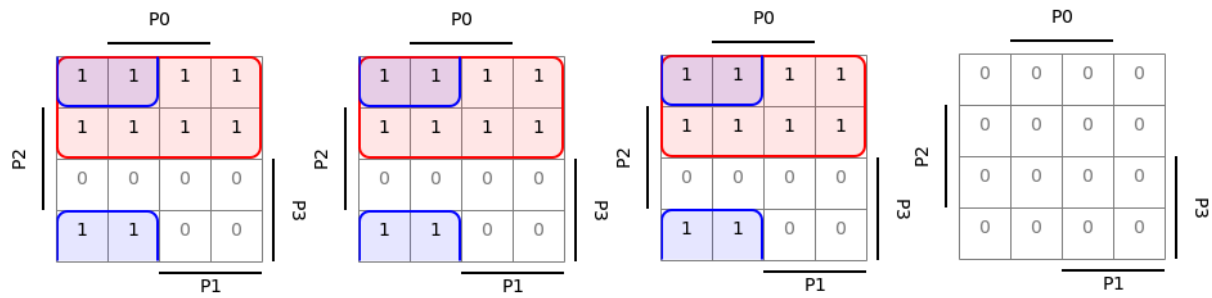


Fig. 6b: K-Map salidas D, E, F y G respectivamente.

A partir de estos K-Maps llegamos funciones lógicas de cada salida, que viendo los mapas se puede observar que las salidas B y C son siempre 1 y la salida G es siempre 0, mientras que las funciones del resto de salidas son:

- $G = 0$
- $F = (P2' \cdot P1') + P3'$
- $E = (P2' \cdot P1') + P3'$
- $D = (P2' \cdot P1') + P3'$
- $C = 1$
- $B = 1$
- $A = (P2' \cdot P1') + P3'$

Estas funciones corresponden a los siguientes circuitos lógicos:

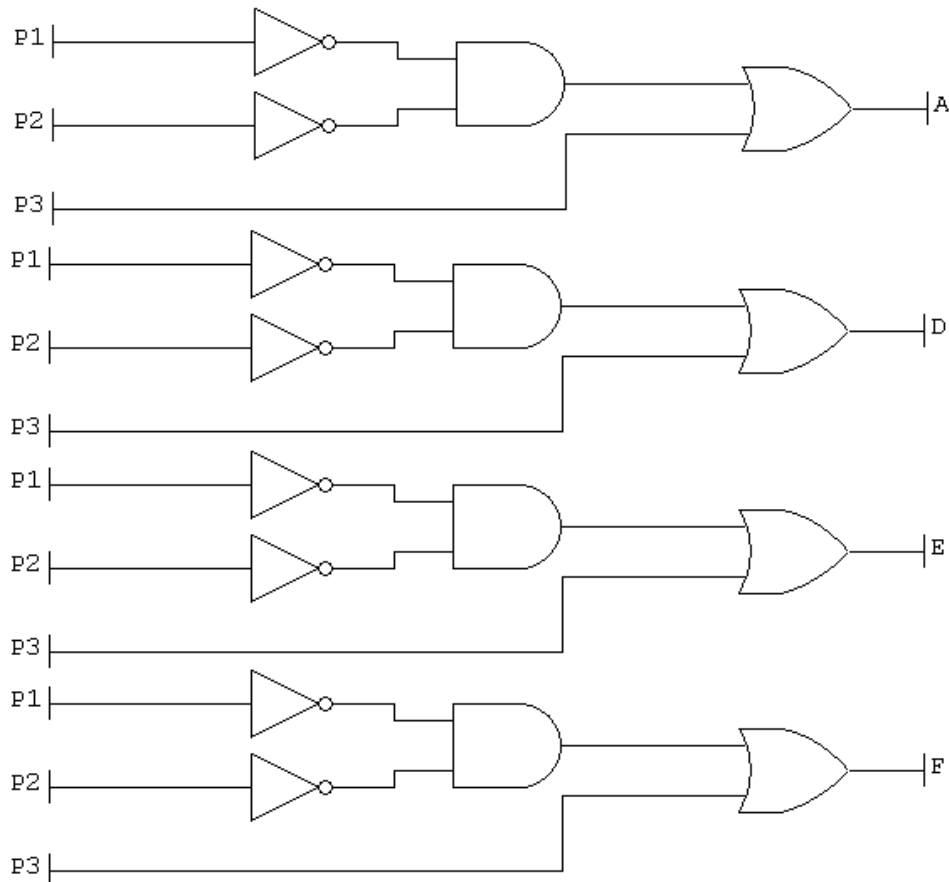


Fig. 7: Circuitos lógicos salidas A, D, E y F.

Bloque L2:

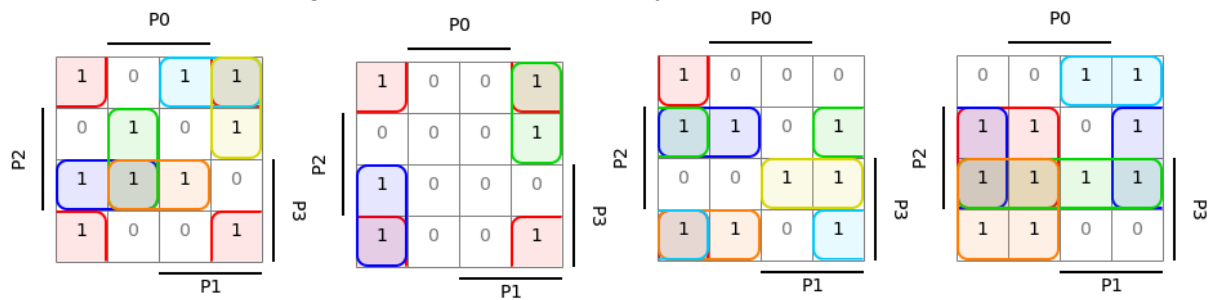
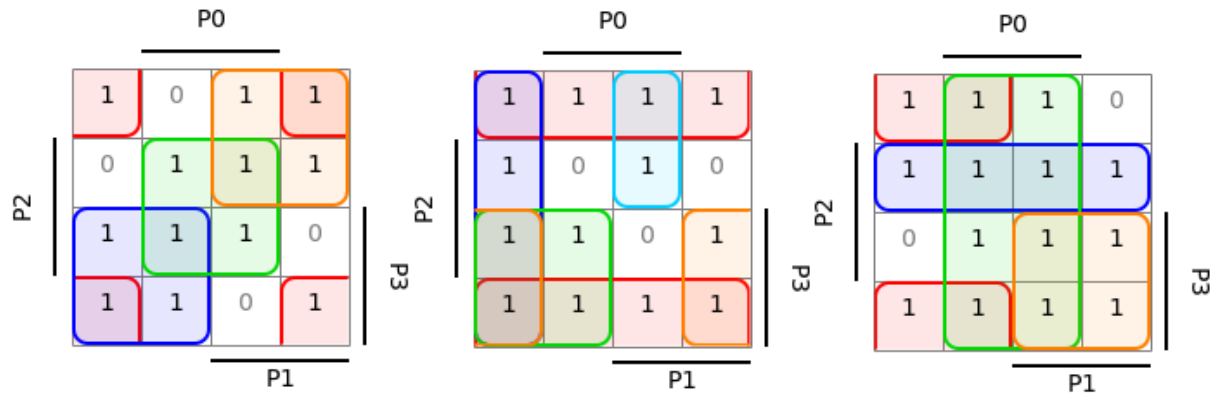
El bloque L2 sigue la misma lógica que el L1, solo que este encenderá los leds necesarios para representar las unidades del número de prioridad decodificado. Es decir que ahora hay que representar todos los números desde “0” a “9”.

Responde a la siguiente tabla de verdad:

#	P3	P2	P1	P0	G	F	E	D	C	B	A
0	0	0	0	0	0	1	1	1	1	1	1
1	0	0	0	1	0	0	0	0	1	1	0
2	0	0	1	0	1	0	1	1	0	1	1
3	0	0	1	1	1	0	0	1	1	1	1
4	0	1	0	0	1	1	0	0	1	1	0
5	0	1	0	1	1	1	0	1	1	0	1
6	0	1	1	0	1	1	1	1	1	0	1
7	0	1	1	1	0	0	0	0	1	1	1
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	0	0	1	1	1
10	1	0	1	0	0	1	1	1	1	1	1
11	1	0	1	1	0	0	0	0	1	1	0
12	1	1	0	0	1	0	1	1	0	1	1
13	1	1	0	1	1	0	0	1	1	1	1
14	1	1	1	0	1	1	0	0	1	1	0
15	1	1	1	1	1	1	0	1	1	0	1

Fig. 8: Tabla de verdad del circuito L2.

Con esta tabla de verdad hicimos el mapa de Karnaugh de cada una de las salidas de la G a la A, cada salida representando el estado encendido/apagado de los leds.



A partir de estos K-Maps llegamos a las funciones lógicas de cada salida:

- $G = (P2 \cdot (P3 + P1' + P0')) + (P3' \cdot P2' \cdot P1) + (P3 \cdot P1')$
- $F = ((P3 \cdot P2') \cdot (P0' + P1')) + ((P3' \cdot P2) \cdot (P1' + P0')) + (P2' \cdot P1' \cdot P0') + (P3 \cdot P2 \cdot P1)$
- $E = P0' \cdot ((P3' \cdot P1) + P2' + (P3 \cdot P1'))$
- $D = ((P3 \cdot P2) \cdot (P0 + P1')) + ((P3' \cdot P1) \cdot (P0' + P2')) + (P2' \cdot P0') + (P2 \cdot P1' \cdot P0)$
- $C = (P3 \cdot P1) + P0 + (P3' \cdot P2) + (P2' \cdot P1')$
- $B = (P3 \cdot (P1' + P0')) + (P3' \cdot P1 \cdot P0) + P2' + (P1' \cdot P0')$
- $A = (P3' \cdot P1) + (P2 \cdot P0) + (P3 \cdot P1') + (P2' \cdot P0')$

Estas funciones corresponden a los siguientes circuitos lógicos:

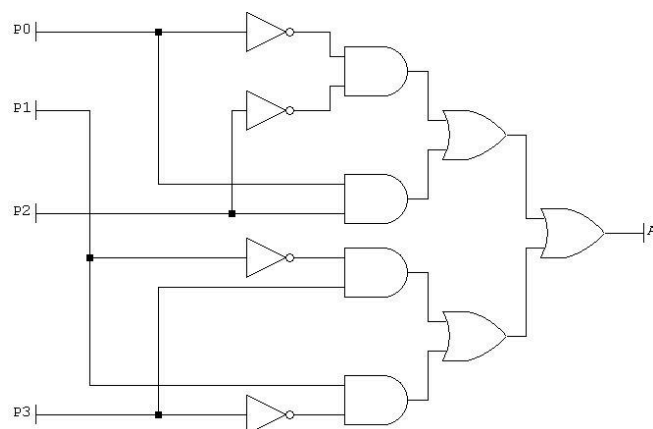


Fig. 10.a: Circuito lógico salida A.

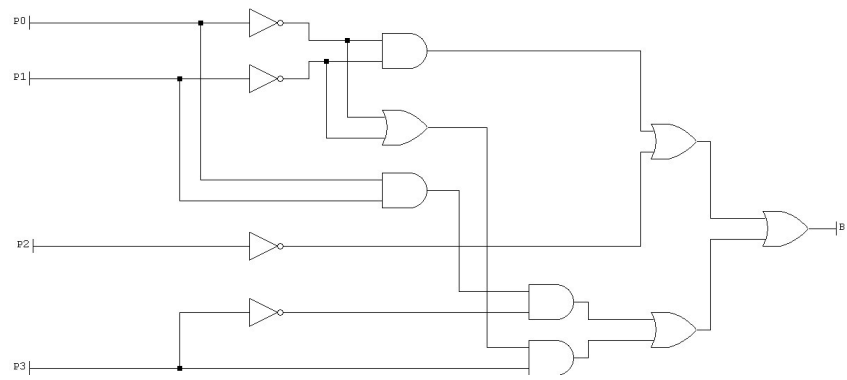


Fig. 10.b: Circuito lógico salida B.

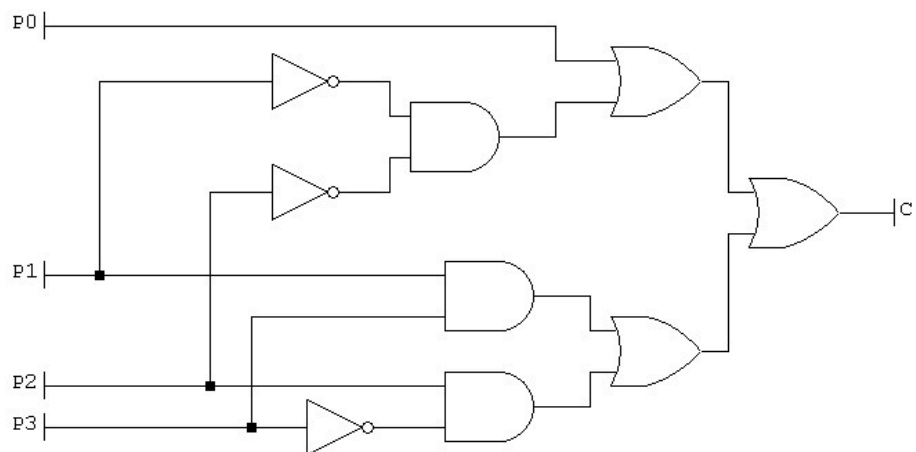


Fig. 10.c: Circuito lógico salida C.

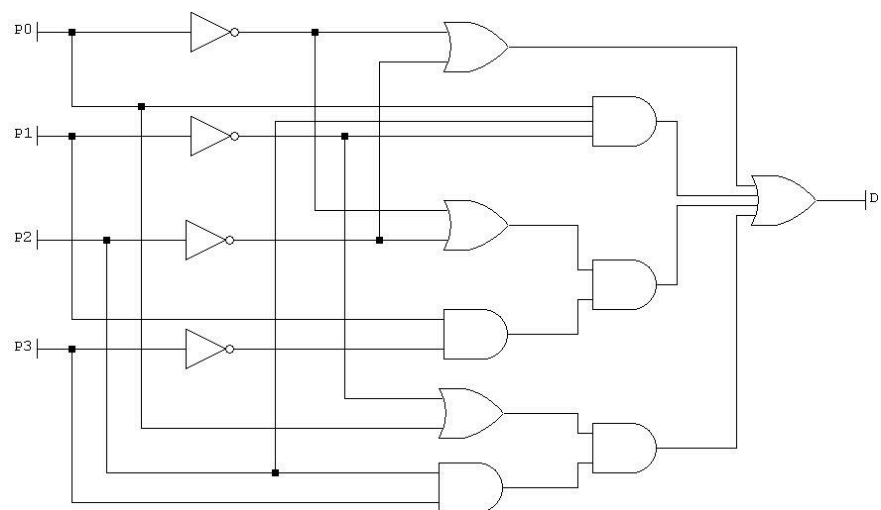


Fig. 10.d: Circuito lógico salida D.

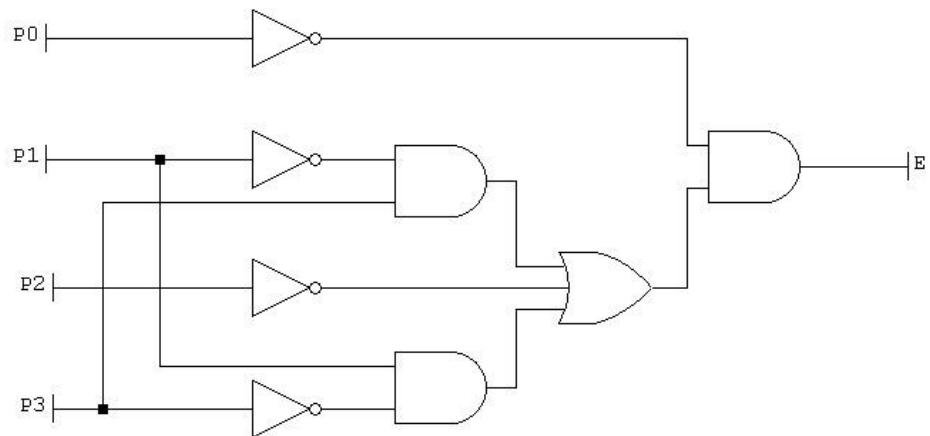


Fig. 10.e: Circuito lógico salida E.

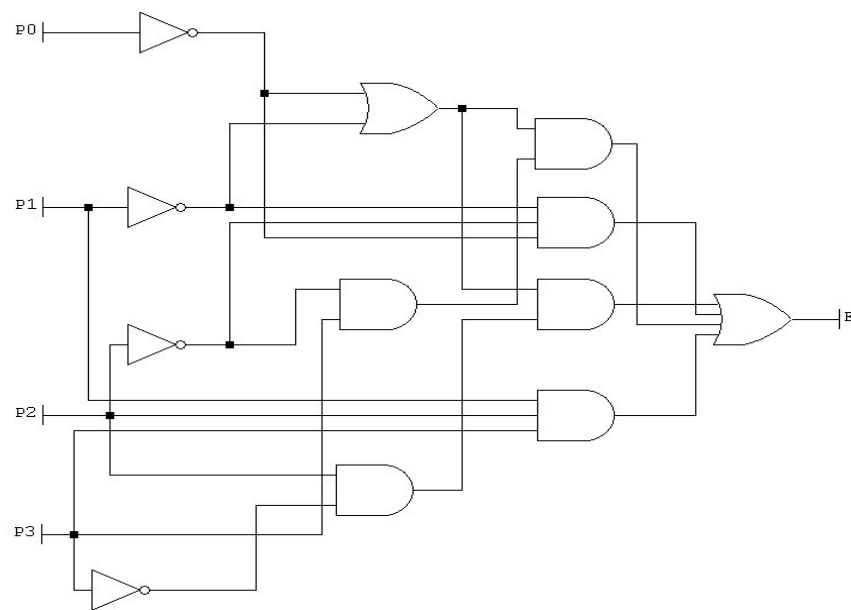


Fig. 10.f: Circuito lógico salida F.

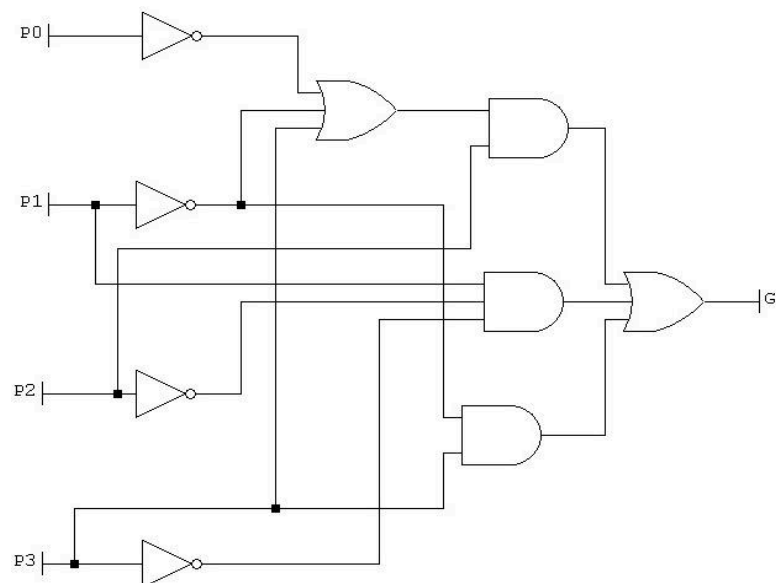


Fig. 10.g: Circuito lógico salida G.

Bloque S:

El bloque S (Signo) lo usará la salida S del bloque P para prender los leds horizontales del medio (es decir, las denominadas como G) si es que ningún sensor está encendido.

S negado está conectado a compuertas AND con las salidas A..F y S estaría conectado con un OR con la salida G.

Si S es 0 entonces S negado es 1, esto significa que hay al menos un sensor prendido entonces todas las ANDs dependen de A..F y el OR se encenderá únicamente cuando G se enciende, esto haría que los leds se prenden como si no existiese el bloque S, en el display se ven los números como deben verse.

Si S es 1 entonces S negado es 0 esto significa que no hay sensores prendidos entonces ninguna AND se encenderá pero el OR se prenderá en 1 por lo que únicamente el led horizontal del medio se prendera.

En esta imagen se puede ver el circuito:

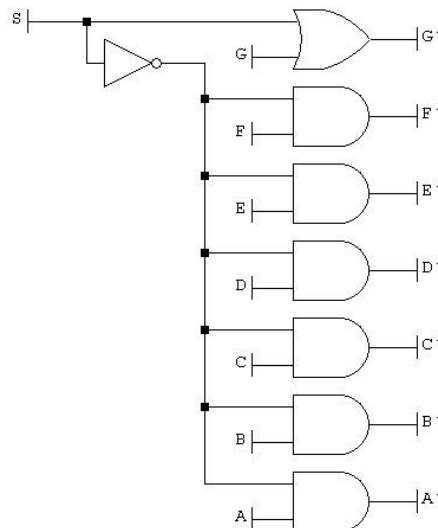


Fig. 11: Circuito lógico bloque S.

Bloque KC/AC:

Este bloque cumple con la funcionalidad de invertir la salida dependiendo si el bit de KA está encendido o apagado a partir de compuertas XOR, pues dicha compuerta invierte una entrada cuando la otra es 1 y la deja como está si la otra entrada es 0. En otras palabras, este bloque cuenta únicamente con compuertas XOR en las salidas A..G. El hecho de que el bit de KA esté en 0 o en 1 depende de si el Display al que pertenecen los LEDs es "cátodo común" o "ánodo común". Nosotros trabajamos estableciendo los resultados como si el Display fuera cátodo común, es decir, que el bit de KA será 0 cuando el Display sea cátodo común ya que no queremos inversión en los altos o bajos de las LEDs.

En la siguiente imagen se muestra el circuito lógico del bloque KC/AC:

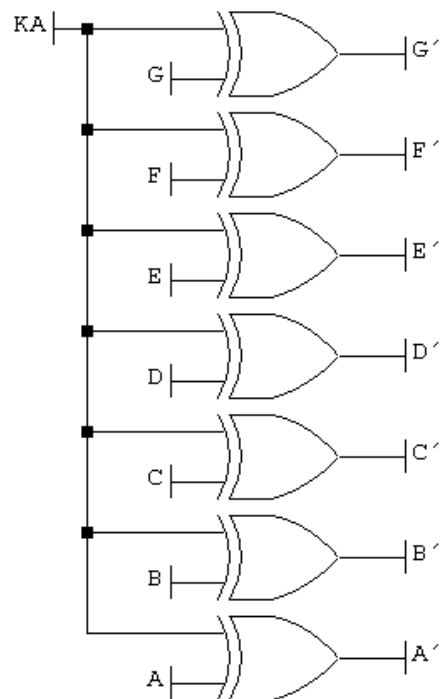


Fig. 12: circuito lógico bloque KC/AC

Ahora bien, el tipo de display de “ánodo común” (AC) es aquel en el que los LED comparten una conexión común a través del ánodo mientras que en los de “cátodo común” (KC) se comparte la conexión común por medio del cátodo. Para los KC se conectan comúnmente todos los cátodos a tierra (GND) y para los AC se conectan todos los ánodos comúnmente a una fuente (VCC).

En esta imagen se puede ver cada caso:

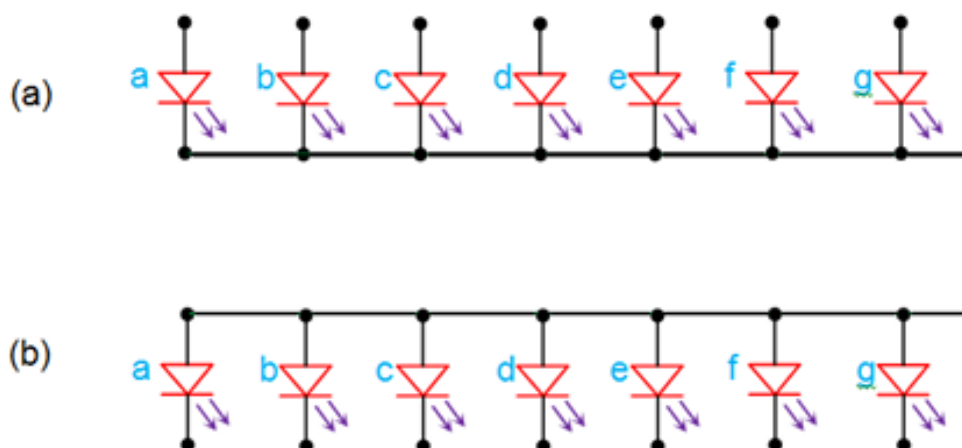


Fig. 13: circuitos para “cátodo común” (a) y “ánodo común” (b).

Cálculo del resistor:

Cada led debe tener su propia resistencia en serie ya que si se tiene una única para todas las leds se corre el riesgo de que si muchas luces están prendidas se ven apaciguadas o directamente no prenden y si solo se prende una sola led se puede llegar a quemar ya que circularía toda la corriente por ahí.

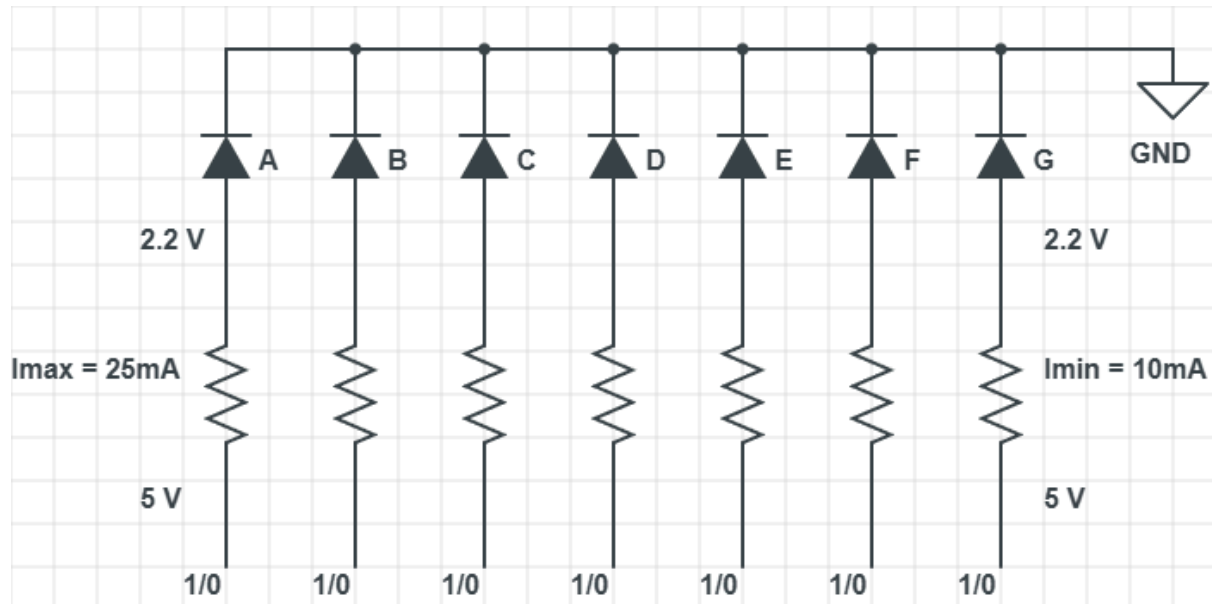


Fig. 14.a: Gráfico cálculo de resistencia en cátodo común.

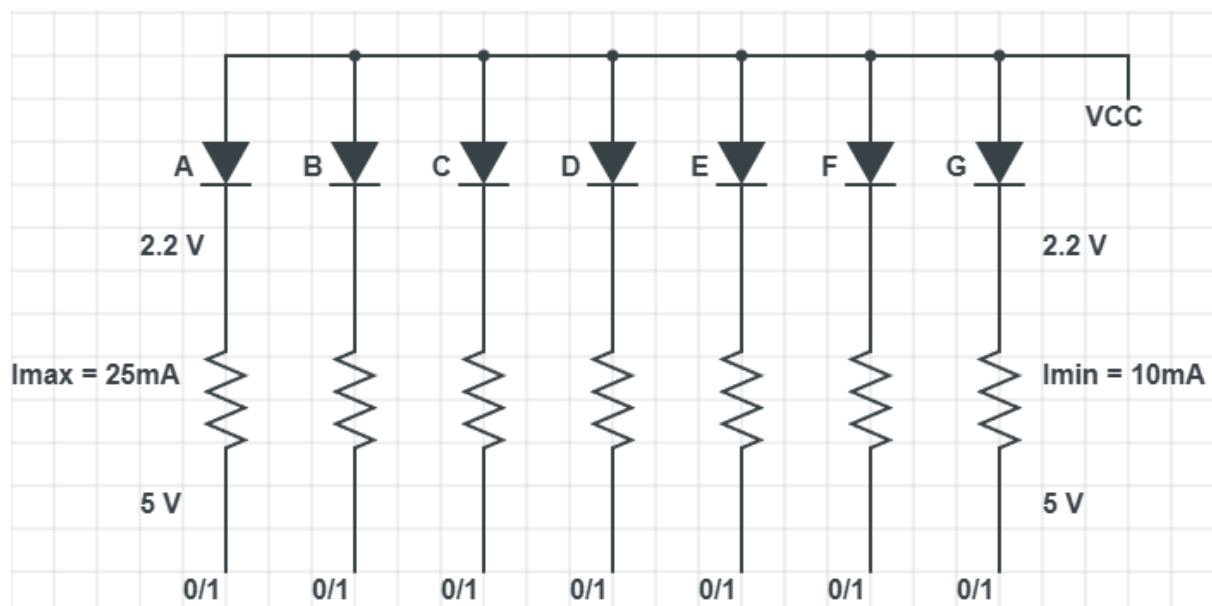


Fig. 14.b: Gráfico cálculo de resistencia en ánodo común.

Para calcular la resistencia se plantea:

- Para la corriente mínima, la resistencia es la máxima:

$$5V - (10A * 0,001 * R_{\text{máx}}) - 2,2V = 0$$

$$\frac{5V - 2,2V}{10A * 0,001} = R_{\text{máx}}$$

$$R_{\text{máx}} = 280\Omega$$

- Para la corriente máxima, la resistencia es la mínima:

$$5V - (25A * 0,001 * R_{\text{min}}) - 2.2V = 0$$

$$\frac{5V - 2,2V}{25A * 0,001} = R_{\text{min}}$$

$$R_{\text{min}} = 112\Omega$$