

ELECTRÓNICA DIGITAL

Cómo usar compuertas,
sumadores y decodificadores
para hablar el idioma de las
computadoras.

Teoría y prácticas sobre el módulo #003



EL MUNDO ES DIGITAL

Olviden los voltajes variables.

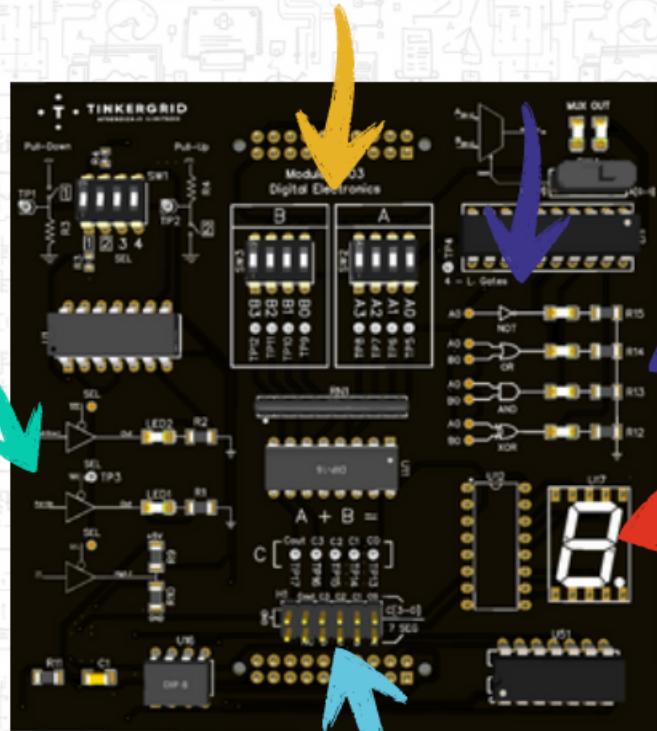
Vivimos rodeados de cosas "Smart": Celulares, IAs y videojuegos.

A diferencia de nosotros que sentimos frío o calor gradualmente (analógico), **las máquinas son radicales:**

Todo es SÍ (1) o NO (0).



CONOCIENDO TU DIGITAL BOARD



Inputs (Centro): Tus teclados binarios. Bancos de interruptores A y B para enviar 0s y 1s.



Logic Gates (Der): El cerebro básico. Compuertas AND, OR, NOT que toman decisiones.



Adder (Centro-Abajo): La calculadora. Un chip que sabe sumar números binarios.



Decoder & Display (Esq-Abajo): La pantalla. Traduce el lenguaje máquina a números que tú puedes leer.



Buffers (Izq): Los semáforos. Controlan el tráfico de datos (High-Z).





PROTEGE TU MÓDULO

La electrónica digital es robusta, pero tiene reglas de oro. Si las rompes, los chips se queman (literalmente).

- **La Muesca Manda:** Todos los chips tienen una marquita o media luna en un extremo. Esta DEBE coincidir con el dibujo en la placa. Si lo pones al revés... ¡Adiós chip!
- **No fuerces nada:** Los pines de los chips son delicados como patas de araña. Si no entra suave, revisa que no estén doblados.
- **Cero Cortos:** Nunca conectes un cable directo de 5V a GND (Tierra). Tu placa se apagará para protegerse, pero es mejor no asustarla.

TEORÍA 1: ESTADOS LÓGICOS Y FANTASMAS

¿Alguna vez has prendido la luz y el foco parpadea? En digitales, eso es fatal. Un cable suelto actúa como antena captando "ruido fantasma". Para evitar que los robots se vuelvan locos, usamos resistencias que fuerzan el estado a Tierra (0) o a Voltaje (1) cuando no presionamos nada.

¿SABÍAS QUE...?

Tu mando de Xbox usa resistencias Pull-Up. Sin ellas, tu personaje saltaría solo o dispararía sin que tú tocaras nada por culpa de la estática de tu ropa.



PRÁCTICA 1 – PULL UP Y DOWN

EL VERDADERO UNO O CERO

INSTRUCCIONES

Las resistencias pull-up y pull-down se usan para **fijar un nivel lógico estable** (alto o bajo) en una entrada digital cuando no está conectada, evitando estados flotantes.

- Ubica la sección Pull-Up / Pull-Down (Esq. Superior Izq.).
- Conecta el TP1 y TP2 a un LED de prueba (TP).
- Modifica la posición de los switches 1 o 2 del DIP SW1.

Resultado: El LED prende y apaga "en seco". No hay parpadeos ni luz tenue. ¡Eso es una señal digital pura!

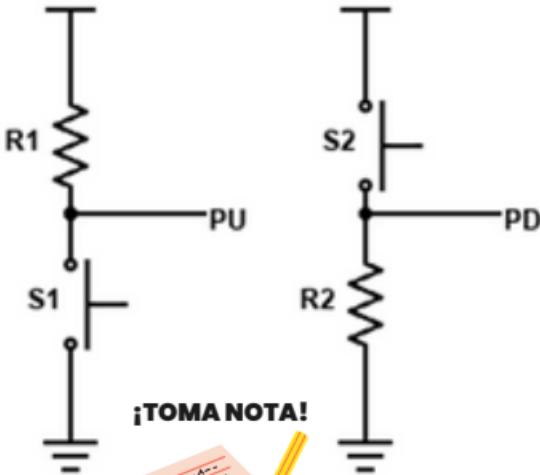
SECCIÓN DE LA TARJETA



PRÁCTICA 1 - PULL UP Y DOWN

EL VERDADERO UNO O CERO

CIRCUITO



Tu reto: Conecta el mismo LED a ambas PU y PD. Observa el comportamiento del mismo. Verifica que marca tu punta logica.

TEORÍA 2: TOMANDO DECISIONES

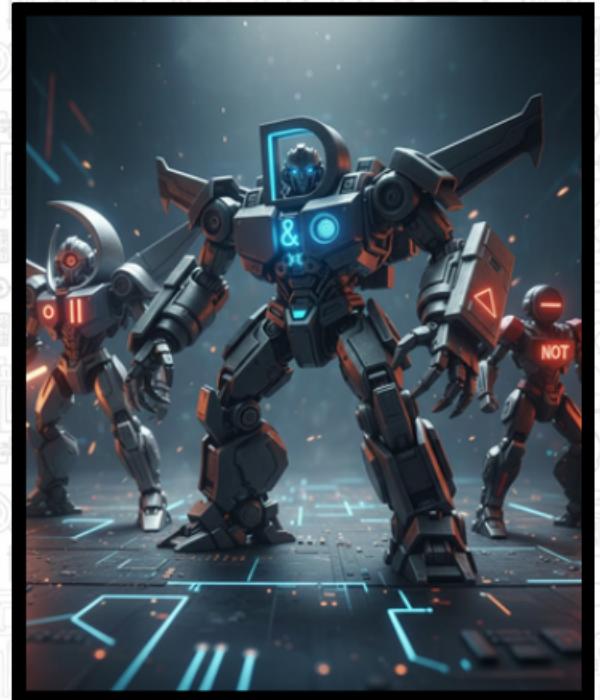
COMPUERTAS LÓGICAS (AND / OR / NOT)

Las máquinas no piensan, solo siguen reglas. Las compuertas son esas reglas:

AND (Y): La estricta. Necesita que A Y B sean verdad. (Ej: Tarjeta Y NIP para sacar dinero).

OR (O): La relajada. Basta con A O B. (Ej: Detectar incendio con humo O calor).

NOT (NO): La opuesta. Si A es verdad, la salida será falsa, y si A es falsa, la salida será verdad. (Ej: Interruptor que apaga la luz cuando está encendido).



TEORÍA 2: TOMANDO DECISIONES

COMPUERTAS LÓGICAS (EXCLUSIVAS)



XOR (O Exclusiva): La selectiva. Solo es verdad si A y B son diferentes; si son iguales, es falsa. (Ej: Interruptor de escalera: luz encendida si uno está arriba y otro abajo).

NAND (NO Y): La desconfiada. Es falsa solo cuando A Y B son verdad; en cualquier otro caso, es verdad. (Ej: Alarma que se activa salvo que dos condiciones críticas se cumplan).

NOR (NO O): La estricta negativa. Solo es verdad cuando A y B son falsas; si alguna es verdad, la salida será falsa. (Ej: Sistema que solo arranca si no hay ninguna alerta activa).

TEORÍA 2: TOMANDO DECISIONES

COMPUERTAS LÓGICAS (RESUMEN)

NOMBRE	AND - Y	OR - O	XOR O-exclusiva	NOT Inversor	NAND	NOR																																																																																	
SÍMBOLO																																																																																							
SÍMBOLO																																																																																							
TABLA DE VERDAD	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	a	b	z	0	0	0	0	1	0	1	0	0	1	1	1	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>	a	b	z	0	0	0	0	1	1	1	0	1	1	1	1	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	b	z	0	0	0	0	1	1	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>a</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	z	0	1	1	0	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	b	z	0	0	1	0	1	0	1	0	1	1	1	0	<table border="1"> <thead> <tr> <th>a</th> <th>b</th> <th>z</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> </tr> </tbody> </table>	a	b	z	0	0	1	0	1	0	1	0	0	1	1	0
a	b	z																																																																																					
0	0	0																																																																																					
0	1	0																																																																																					
1	0	0																																																																																					
1	1	1																																																																																					
a	b	z																																																																																					
0	0	0																																																																																					
0	1	1																																																																																					
1	0	1																																																																																					
1	1	1																																																																																					
a	b	z																																																																																					
0	0	0																																																																																					
0	1	1																																																																																					
1	0	1																																																																																					
1	1	0																																																																																					
a	z																																																																																						
0	1																																																																																						
1	0																																																																																						
a	b	z																																																																																					
0	0	1																																																																																					
0	1	0																																																																																					
1	0	1																																																																																					
1	1	0																																																																																					
a	b	z																																																																																					
0	0	1																																																																																					
0	1	0																																																																																					
1	0	0																																																																																					
1	1	0																																																																																					
EQUIVALENTE EN CONTACTOS																																																																																							
AXIOMA	$z = a \cdot b$	$z = a + b$	$z = \bar{a} \cdot b + a \cdot \bar{b}$	$z = \bar{a}$	$z = \bar{a} \cdot \bar{b}$	$z = \bar{a} + \bar{b}$																																																																																	

PRÁCTICA 2 – COMPUERTAS LÓGICAS

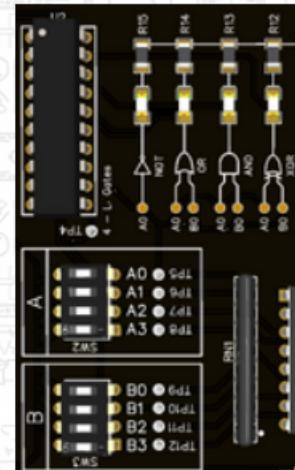
EL PRINCIPIO DE TODO PROCESADOR

INSTRUCCIONES

Las **compuertas lógicas** son circuitos digitales que aplican reglas booleanas para procesar señales binarias y tomar decisiones en sistemas electrónicos.

- Vamos a la sección de Gates.
- Usa los Dip Switch A0 y B0 como entradas.
- Conéctalos a la compuerta AND.
- Prueba todas las combinaciones que puedas en 5 minutos.

SECCIÓN DE LA TARJETA

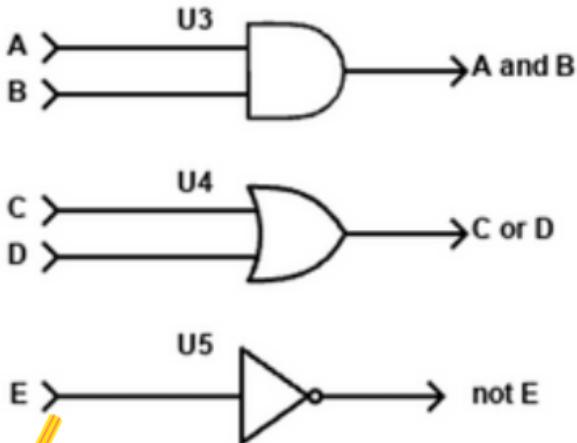


Resultado: Si compruebas la tabla de verdad anterior, podrás confirmar que el LED indica la salida. ¡Sin magia, solo lógica booleana!

PRÁCTICA 2 - COMPUERTAS LÓGICAS

EL PRINCIPIO DE TODO PROCESADOR

CIRCUITO



¡TOMA NOTA!

Tu reto: ¿Cuál es la única combinación que enciende el LED?

Respondesta:

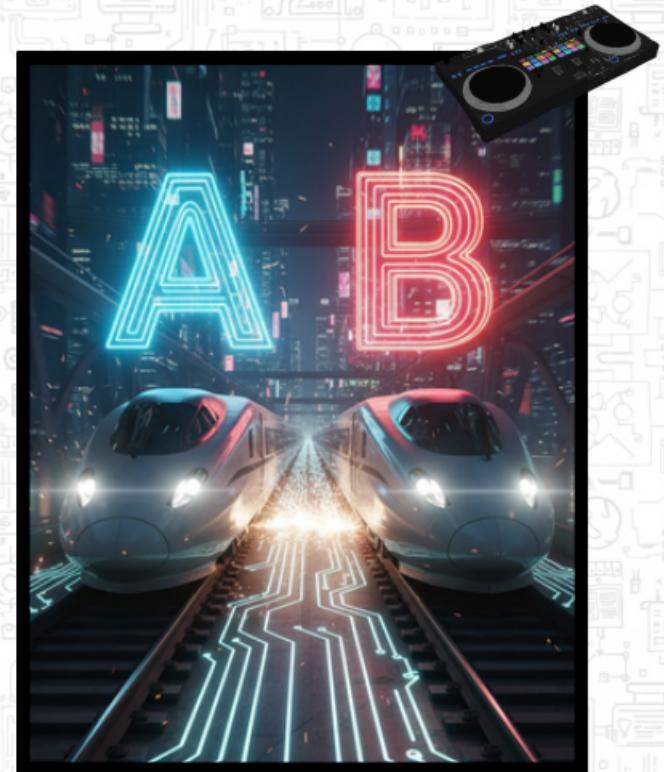
Solo cuando ambos son 1.

TEORÍA 3: EL DJ DEL CIRCUITO

Imagina dos trenes (Datos A y Datos B) queriendo entrar a una sola vía. **Si chocan, se pierden los datos.** El **Multiplexor** es como el **cambio de vía para salvarlos.**

Un pin de "Selección" decide quién pasa. Nos ayuda a decidir hacia dónde fluye la información.

En otro contexto, un multiplexor es vital para enviar internet por un solo cable de fibra óptica.



PRÁCTICA 3 – MUX

SELECCIONANDO LA INFORMACIÓN

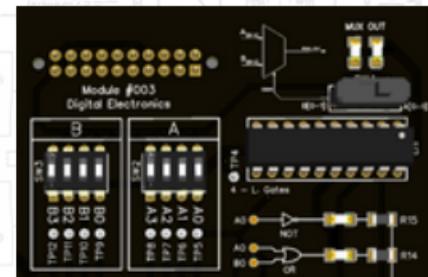
INSTRUCCIONES

Un MUX (Multiplexor) es un dispositivo digital que selecciona una de varias señales de entrada y la dirige a una única salida según las líneas de control.

- Pon un "1" en la entrada A y un "0" en la B.
- Mueve el interruptor SW4 (SEL).

Resultado: El LED de salida cambia entre copiar a A o copiar a B. ¡Tú controlas el tráfico de información!, de pronto nos volvimos un agente vial.

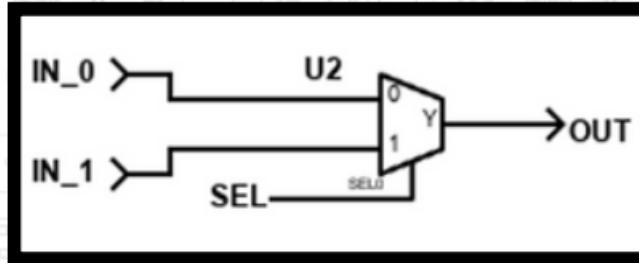
SECCIÓN DE LA TARJETA



PRÁCTICA 3 – MUX

SELECCIONANDO LA INFORMACIÓN

CIRCUITO



¡TOMA NOTA!

Tu reto: Conecta la salida de alguna compuerta lógica al pin de SEL, observa cómo puedes configurar a que el MUX siga la lógica de las compuertas que vimos. ¡Prueba varias!

TEORÍA 4: MATEMÁTICAS ROBÓTICAS

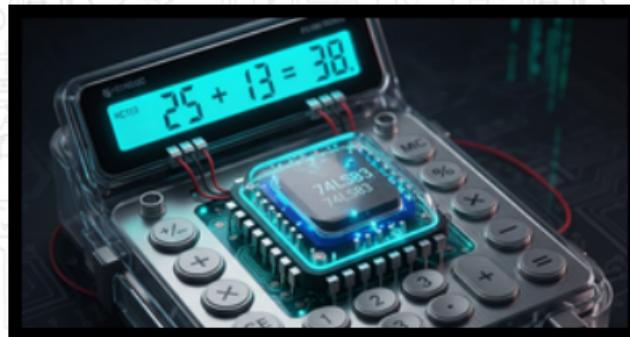
¿Cómo suma una calculadora? Usando lógica.

$$0 + 0 = 0$$

$$1 + 0 = 1$$

$1 + 1 = 0$ (y llevamos 1, lo que llamamos **carry**).

Tu placa tiene un chip **74LS283**, un cerebro matemático de 4 bits.



PRÁCTICA 4 - 74LS283

TODO FRIAMENTE CALCULADO

INSTRUCCIONES

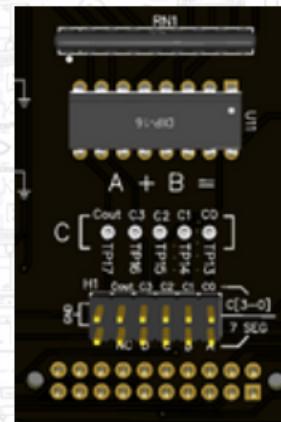
El 74LS283 se utiliza como sumador binario de 4 bits, realizando la suma de dos números de 4 bits y generando una salida con acarreo para operaciones aritméticas más grandes.

- Identifica el chip sumador en U11.
- Pon el número 3 en binario (0011) en el banco A.
- Pon el número 2 en binario (0010) en el banco B.



Resultado: Los LEDs de salida "C" deben mostrar 0101 (que es 5 en binario). ¡Tu placa acaba de hacer su primera tarea de matemáticas! Sé cómo tu placa, haz tu tarea.

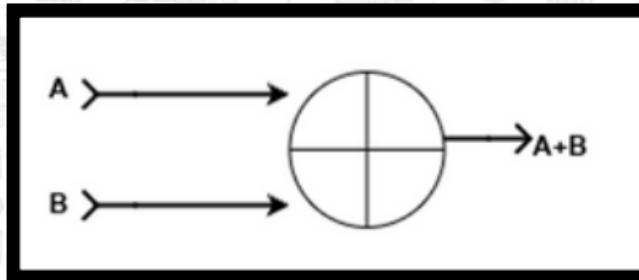
SECCIÓN DE LA TARJETA



PRÁCTICA 4 - 74LS283

TODO FRIAMENTE CALCULADO

CIRCUITO



¡TOMA NOTA!

Tu reto: Intenta sumar el numero más grande de un bus de datos + el numero más grande del otro bus, ¿qué observas?

TEORÍA 5: HABLANDO HUMANO

Ver luces binarias (0101) es confuso. Para que la tecnología sea útil, debe hablarnos en decimal.

El **Decodificador** es el traductor. **Convierte el lenguaje máquina en señales para encender los segmentos del display y formar cualquier numero, en gigante.**

¿SABÍAS QUE...?

Los marcadores de los estadios de fútbol antiguos usaban exactamente esta tecnología para mostrar los goles.



PRÁCTICA 5 – DECODIFICADOR BCD

TRADUCIENDO CÓDIGOS A NÚMEROS

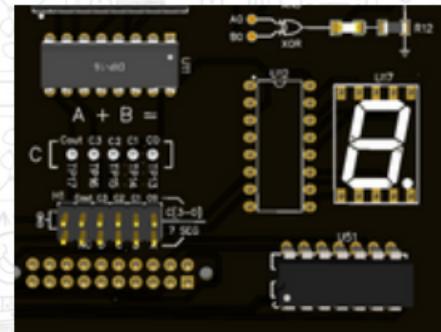
INSTRUCCIONES

Un decodificador **BCD a 7 segmentos** convierte un número en código BCD (4 bits) en señales que activan los segmentos de un display para mostrar el dígito decimal correspondiente.

- Conecta la salida de tu suma ($C_0 - C_3$) a la entrada del Decodificador (U12).
- Cambia los valores de las entradas para que la suma muestre valores distintos.

Resultado: ¡Listo! El resultado de tu suma ya no son puntitos de luz, sino un número legible en el display. ¿Ahorramos varios pasos verdad?

SECCIÓN DE LA TARJETA

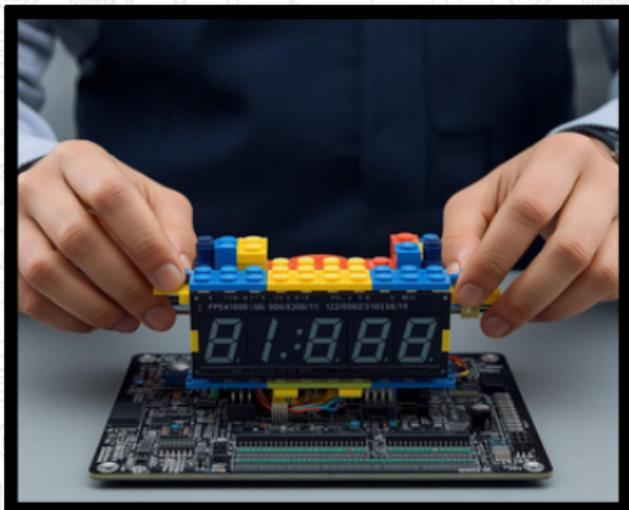


TU SIGUIENTE RETO

Ya dominas el flujo:
Entrada → Proceso → Salida.

Tu Misión: Configura el sistema para que sume dos números, pero usa el Multiplexor antes del display para decidir si quieres ver el Resultado de la Suma O si quieres ver solo el valor de la Entrada A.

¡La electrónica digital es el LEGO del futuro, y tú ya sabes armarlo!



TU CARRERA HA DESPEGADO

¡Felicitaciones! Lo que acabas de practicar en tu placa #003 es el principio fundamental de:

- **Ingeniería en Computación:** Diseñar los procesadores del mañana (Intel, AMD, Apple Silicon).
- **Robótica e IA:** Crear cerebros artificiales que toman decisiones lógicas.
- **Telecomunicaciones:** Enviar datos a la velocidad de la luz.

Esto no fue un juego; fue entrenamiento de ingeniería real. Los conceptos de Lógica, Aritmética Binaria y Decodificación **son las herramientas que usarás para inventar el futuro.**





¡MUCHAS GRACIAS!



tinkergridmx