

Timer + PWM Generator

Juan Esteban Ortiz(juane.ortizm@ugvirtual.edu.co),
Mario Fernando Perez Martinez (mariof.perez@ugvirtual.edu.co)
Jaider Ivan Colimba Chiran (jaideri.colimbac@ugvirtual.edu.co)

Diagrama de Bloques

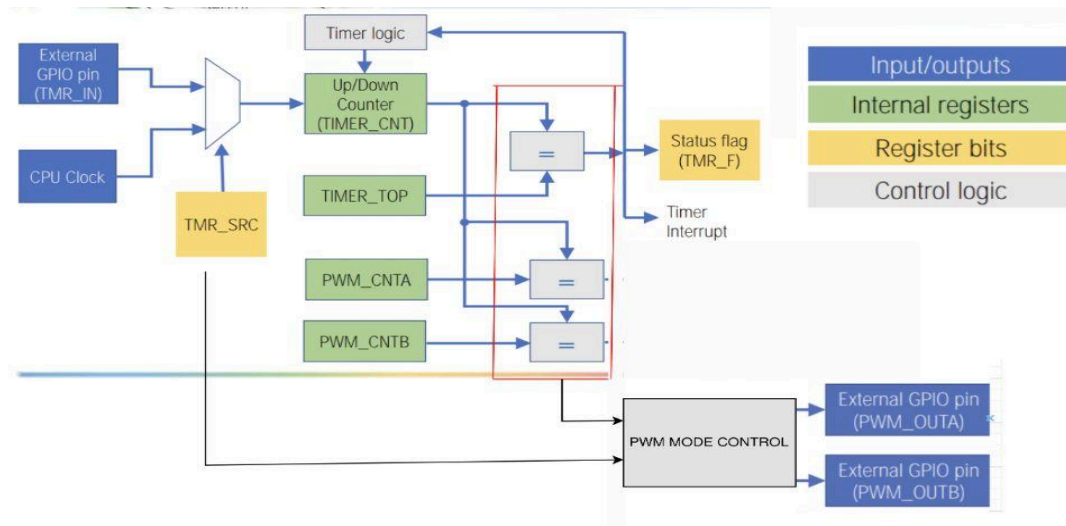


Figura 1. Diagrama de Bloques de Diseño

Timer + PWM Generator

Descripción del Módulo y proposito :

El módulo Timer + PWM Generator está diseñado para generar señales de temporización y modulación por ancho de pulso de manera eficiente en sistemas integrados basados en la arquitectura. Ofrece configuraciones flexibles para adaptar los modos de operación a diferentes necesidades de aplicaciones.

Entradas y Salidas :

Entradas:

- CPU Clock: Fuente de reloj principal utilizada para el conteo y generación de señales PWM.
- TMR_IN: Entrada opcional desde un pin GPIO para habilitar el temporizador con una señal externa.
- Registros de Control: TIMER_CTRL: Configura el modo de operación (Normal, Fast PWM, Phase Correct PWM) y habilita/deshabilita el temporizador.
- PWM_CNTA y PWM_CNTB: Valores de referencia para los comparadores.
- TIMER_TOP: Define el valor máximo del contador.

Salidas:

- PWM_OUTA: Señal PWM generada basada en PWM_CNTA.
- PWM_OUTB: Señal PWM generada basada en PWM_CNTB.
- Interrupt: Señal de interrupción generada cuando TIMER_CNT alcanza TIMER_TOP.

Estructura del Módulo :

El diseño del módulo se organiza en bloques funcionales para facilitar su implementación:

1. Entradas Principales

Multiplexor (MUX): Selecciona entre CPU Clock y TMR_IN como la fuente de reloj para el temporizador, según lo configurado en los registros.

2. Contador y Lógica de Temporización

TIMER_CNT: Contador de 32 bits que incrementa o decrementa los ciclos del reloj seleccionado.

TIMER_TOP: Registro que define el límite superior del contador. Cuando TIMER_CNT alcanza este valor:

Se genera una señal de interrupción.

En los modos PWM, el contador se reinicia o cambia de dirección.

3. Generación de PWM

Comparadores:

Comparador A: Compara TIMER_CNT con PWM_CNTA para controlar la salida PWM_OUTA.

Comparador B: Compara TIMER_CNT con PWM_CNTB para controlar la salida PWM_OUTB.

PWM Mode Control: Lógica de control para los modos Fast PWM y Phase Correct PWM, ajustando las salidas PWM según la configuración.

4. Lógica de Control General

Administra señales de habilitación y configura el modo del temporizador, asegurando una operación sincronizada entre todos los bloques.

Modos de Operación

Modo Normal:

TIMER_CNT incrementa hasta TIMER_TOP y genera una interrupción.

Modo Fast PWM:

La señal PWM se genera contando desde 0 hasta TIMER_TOP, y reinicia al llegar al valor máximo.

Modo Phase Correct PWM:

La señal PWM se genera contando hacia arriba y luego hacia abajo, produciendo una forma de onda simétrica.

Características Destacadas

Flexibilidad: Admite múltiples modos de operación.

Precisión: Resolución de 32 bits para la temporización y generación de PWM.

Cobertura de la Especificación

En el diseño propuesto se implementaron las principales funcionalidades del generador de PWM, incluyendo los submódulos correspondientes a los modos Fast PWM, Phase Correct y Normal. La validación de estas funcionalidades se llevó a cabo mediante simulaciones en Icarus Verilog, generando diagramas de tiempo que mostraron el comportamiento esperado del módulo completo y de los submódulos individuales.

Sin embargo, no se logró un análisis detallado a nivel de comportamiento RTL en DigitalJS, especialmente en lo relacionado con:

La interacción interna entre los submódulos.

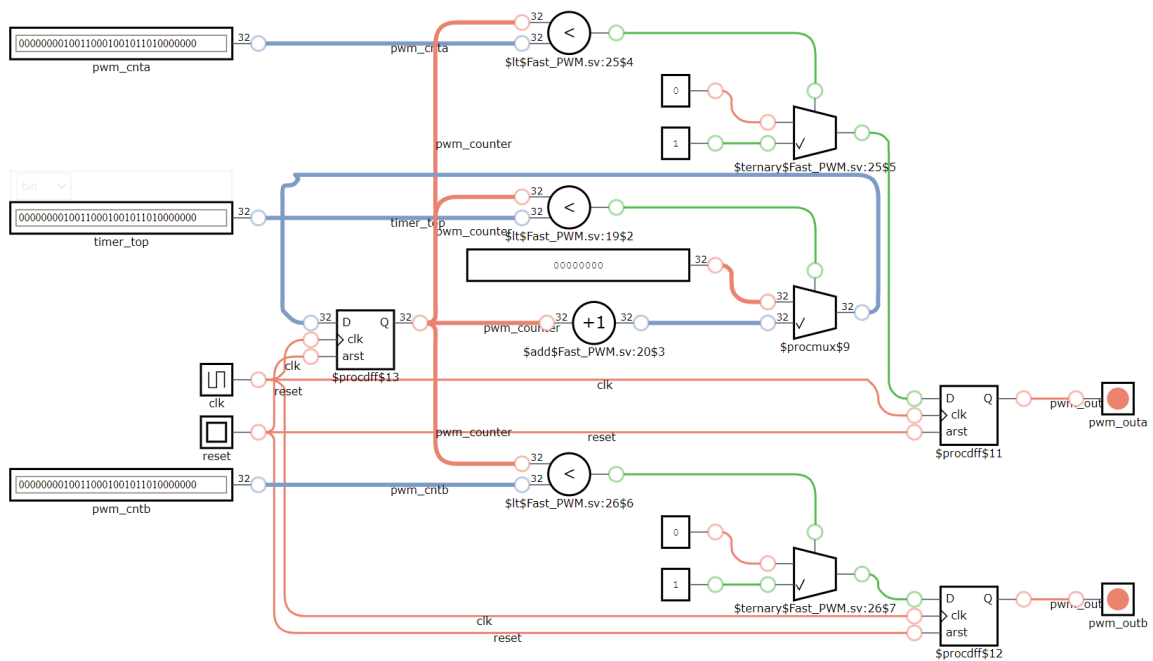


Figura 4. Modo_FAST_PWM:

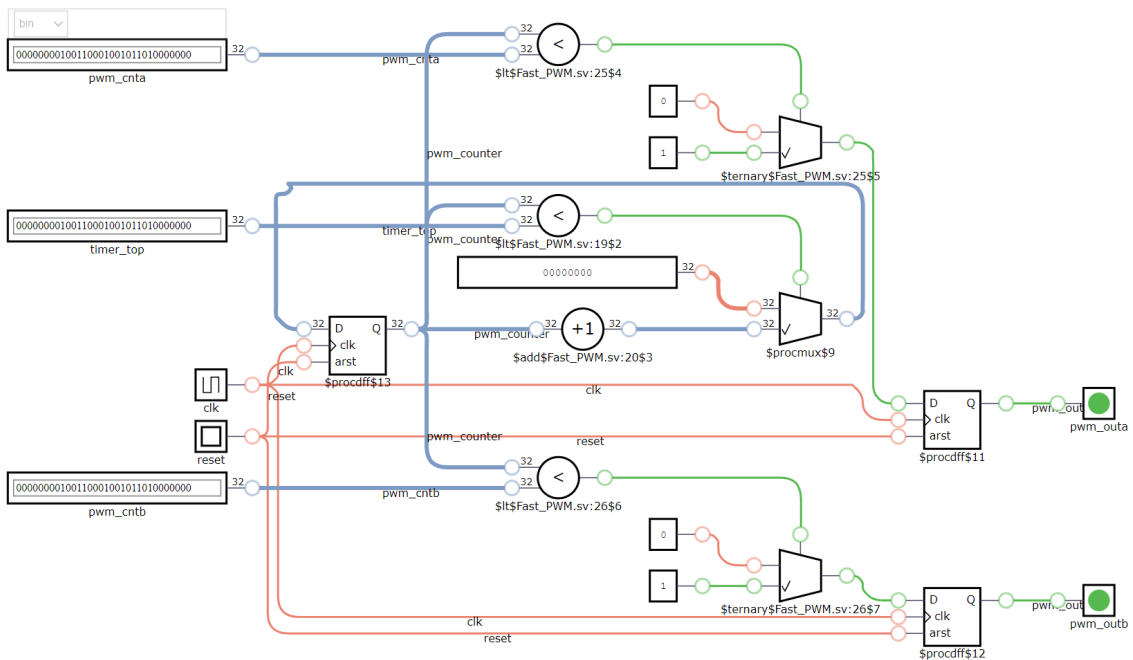
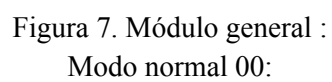


Figura 5. Modo_FAST_PWM cuando llegó al TIMER_TOP



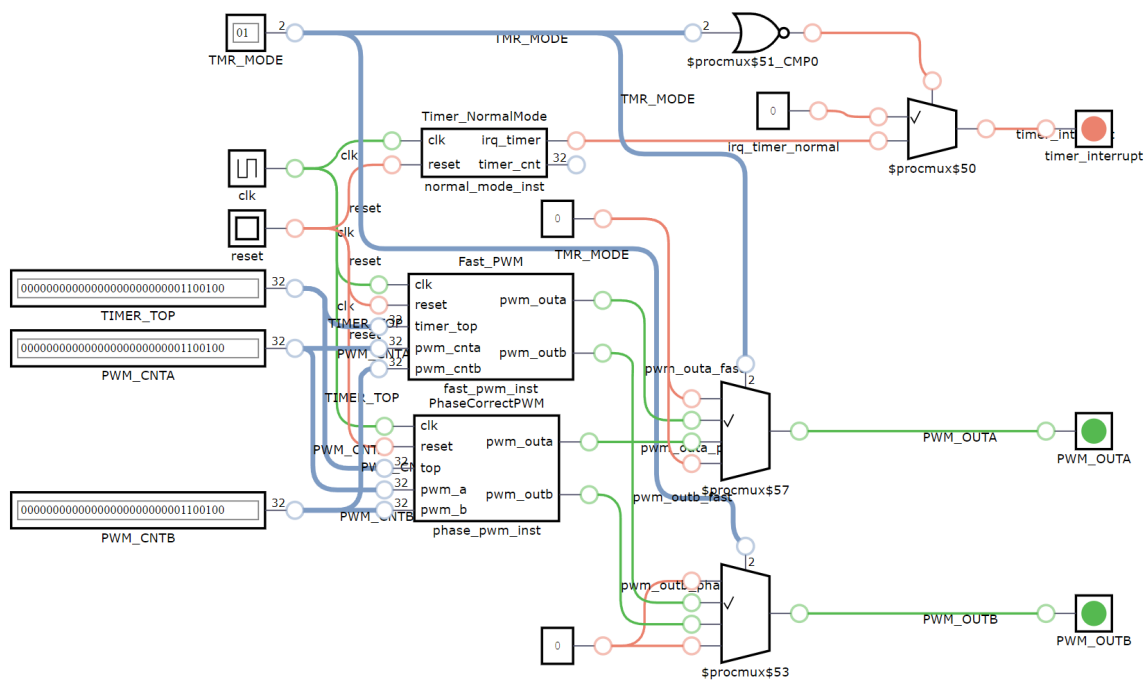


Figura 8. Módulo general :
Modo Fast_PWM 01

Modo phase correct pwm 10:

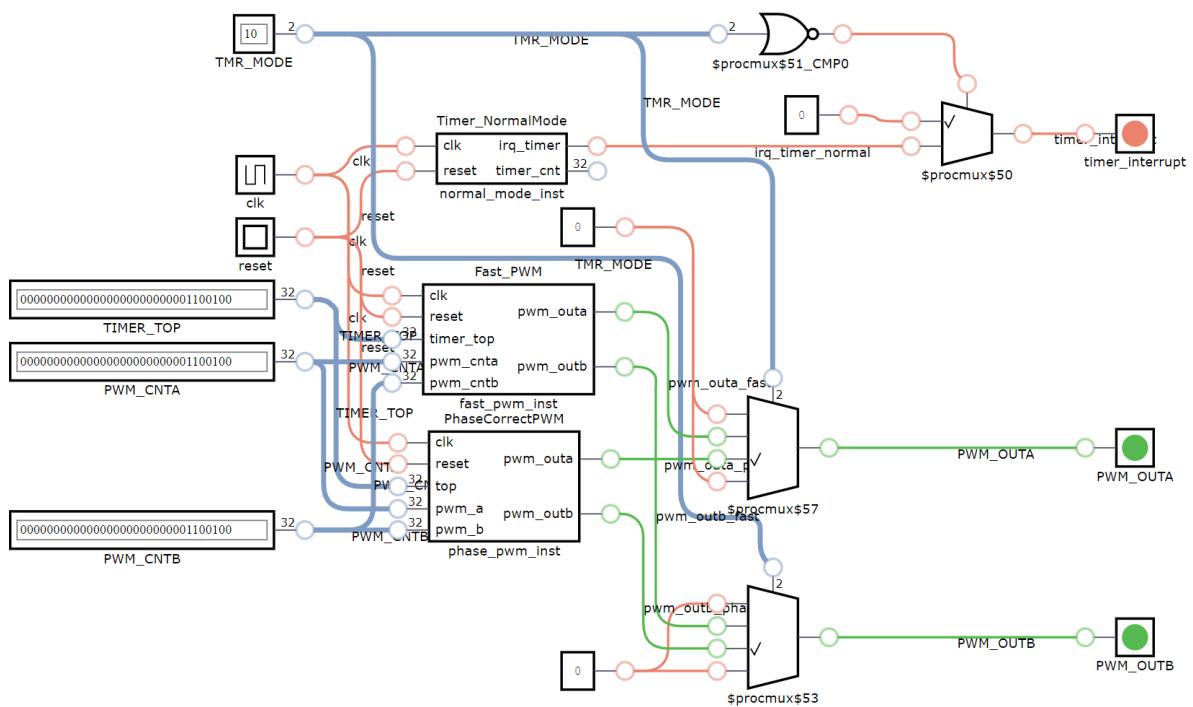


Figura 9. Modo PHASECORRECT 10:

Diagramas de Tiempo

Se obtuvieron diagramas de tiempo en Icarus Verilog que verificaron los siguientes puntos:

- La correcta generación de señales PWM en ambos modos.
- La sincronización entre el temporizador base y los submódulos.
- El cambio de duty cycle y su impacto en la salida.

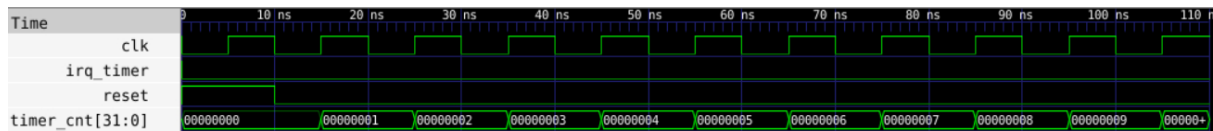


Figura 10. Señales de modo Normal 00

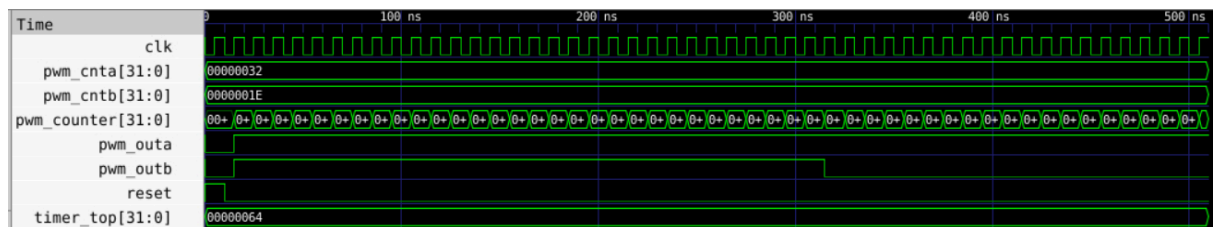


Figura 11. Señales de modo FAST_PWM 01

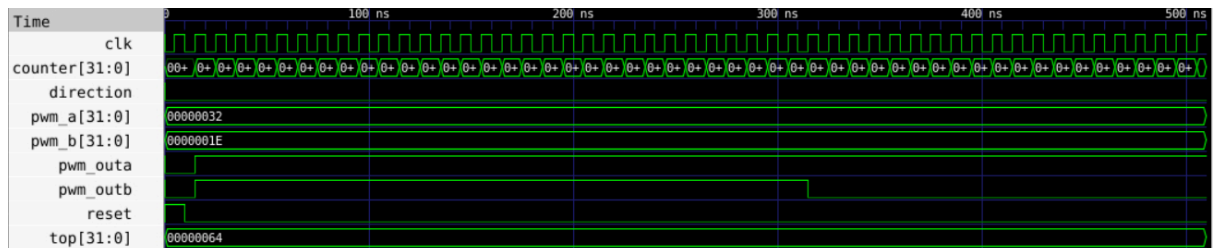


Figura 12. señales Modo PHASE_CORRECTPWM 10

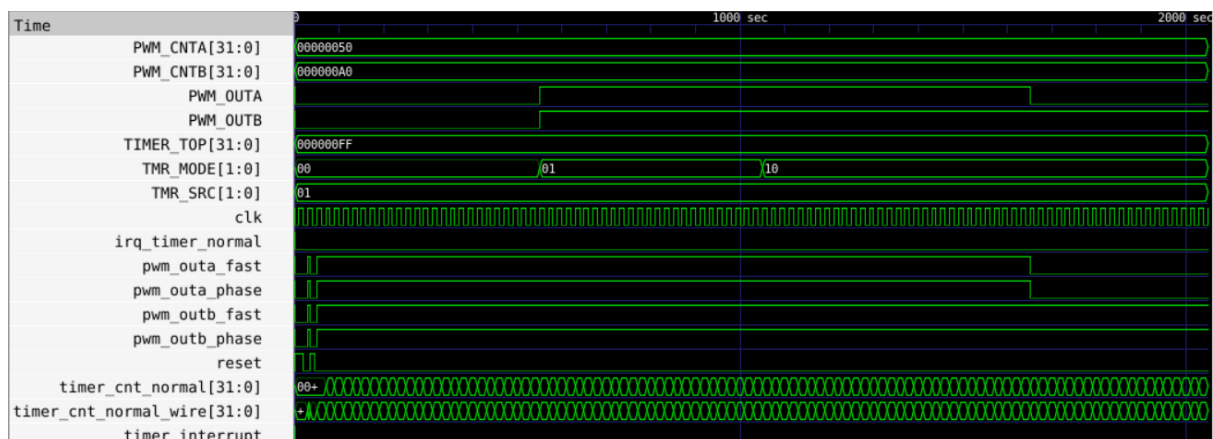


Figura 13. Diagrama de tiempo general .

Caracterización de área y temporización del chip

Se realizó una caracterización de área y temporización, los resultados obtenidos se presentan en la siguiente tabla, donde se evaluaron diferentes estrategias de síntesis.

SYNTH_STRATEGY	Gates	Area (μm²)	Worst Setup Slack (ns)	Total -ve Setup Slack (ns)
AREA 0	1110	10652.716800	-0.409882805964...	-0.6572724772706...
AREA 1	1069	10397.472000	0.9163505769107...	0.0
AREA 2	1047	10292.371200	1.0756364992046...	0.0
AREA 3	1638	13753.190400	4.2823314995038...	0.0
DELAY 0	1208	12087.843200	1.3007328915459...	0.0
DELAY 1	1160	11637.411200	2.7941445503757...	0.0
DELAY 2	1122	11257.046400	2.6969947039029...	0.0
DELAY 3	1172	11796.313600	0.4669784691112...	0.0
DELAY 4	1184	11926.438400	3.0130667702578...	0.0

Figura 14. Caracterización de área y temporización del chip

Con base en los resultados, se seleccionó la estrategia **AREA 2**, ya que presentó el mejor compromiso entre área y temporización. Se realizó la configuración correspondiente en el archivo config.json para optimizar el diseño a partir de esta selección.

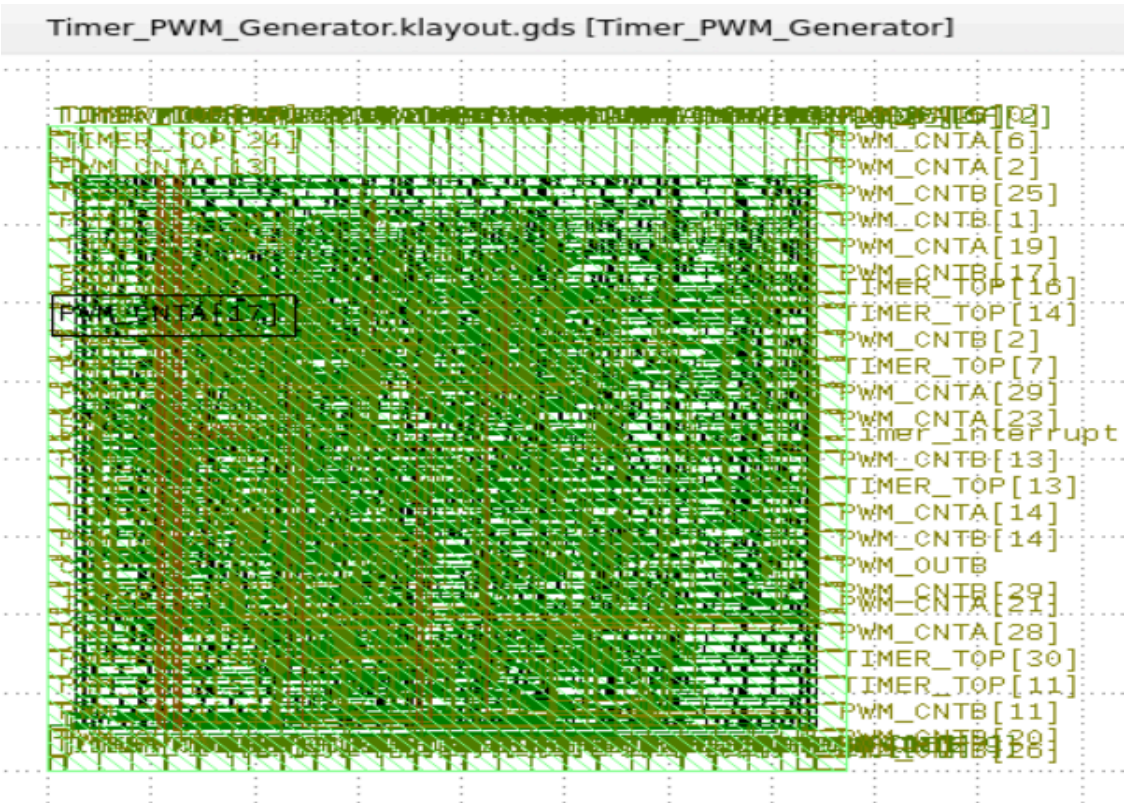


Figura 15. Layout de Klayout

La imagen mostrada corresponde al layout final del módulo **Timer_PWM_Generator** generado

mediante **KLayout**. Este layout representa las máscaras necesarias para la fabricación del circuito integrado, donde se pueden observar las capas asociadas al diseño físico del sistema.

Observaciones sobre la imagen:

1. **Visibilidad de capas:** La imagen presenta un diseño denso con múltiples conexiones y etiquetas correspondientes a las señales principales del módulo, como PWM_CNTB, y TIMER_TOP. Estas etiquetas permiten identificar las interconexiones y las estructuras lógicas que componen el sistema.
2. **Dificultades en la visualización:** No se logró observar claramente las distintas capas del diseño debido a que no se recordaron los pasos exactos para configurar la visualización de las capas en **KLayout**. Esto afecta la claridad del análisis de las diferentes regiones funcionales como las conexiones metálicas, difusiones y contactos.
3. **Detalles destacados:**
 - Las etiquetas principales están distribuidas alrededor del perímetro del diseño lo que facilita identificar las entradas y salidas del sistema.
 - Aunque no se tienen separadas visualmente las capas de metalización difusiones o vías el layout refleja la integración final del diseño lógico y físico.