

0.1. Introducción

0.2. Conversor ADC0808

El ADC0808 es un conversor analógico-digital de ocho bits con ocho canales multiplexados y salida binaria paralela. En el presente informe se utilizó un único canal analógico. Este conversor está compuesto por una red en escalera 256R conectada a un árbol de llaves, el cual compara los distintos valores proporcionados por la red 256R con la entrada analógica. Es así —mediante el registro de aproximaciones sucesivas, el cual realiza una búsqueda binaria con todos los valores posibles de la red escalera hasta converger a un valor digital óptimo— que logra el ADC0808 realizar la conversión analógica-digital.

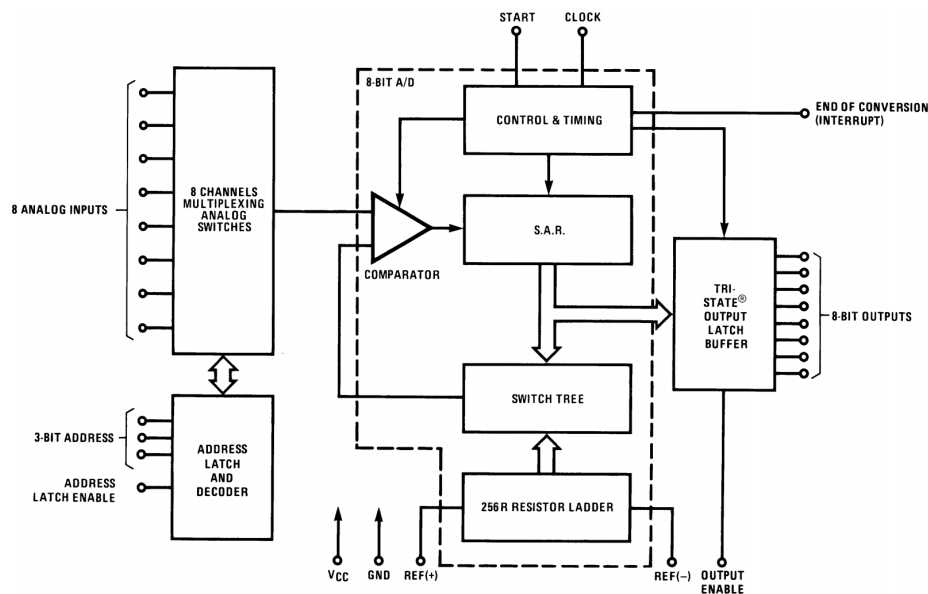


Figura 1: Diagrama en bloques simplificado del conversor ADC0808.

A continuación, se detallará la funcionalidad de cada pin del integrado mostrado en la Figura (1) de manera simplificada donde se puede observar el diagrama temporal en la Figura (2).

- **8 Analog Inputs:** Aquí se colocan las entradas analógicas que se desean convertir. Se utilizó únicamente una sola de estas entradas, mientras que el resto se las conectó a tierra para evitar el ruido electromagnético.
- **3-bit Address:** Esta entrada binaria permite seleccionar qué entrada analógica se utilizará para realizar la conversión. Estas entradas se conectaron permanentemente a tierra, de manera tal que siempre se realice la conversión con la primer entrada analógica.
- **Address Latch Enable:** Para un valor alto, el ADC0808 mantendrá registro del último address ingresado. Se conectó a VCC.
- **VCC:** Tensión positiva de alimentación, se utilizó el valor de $5.12V$ de tal manera que la resolución a la salida sea de $20mV$ por bit.
- **GND:** Tierra del circuito
- **REF(+), REF(-):** Tensiones de referencia para la red escalera 256R. Se conectó la referencia positiva a VCC y la negativa a GND.
- **Clock:** El clock utilizado fue el típico extraído de la datasheet, de $640kHz$.
- **Start:** Esta señal de control inicia un ciclo de conversión.
- **End of Conversion:** Esta señal de interrupción pasa a un estado alto cuando se acaba un ciclo de conversión. Si se conecta esta señal a la señal de **Start** se logra la máxima frecuencia de conversión para una señal de **Clock** fija.
- **8-bit Outputs:** Salida digital paralela de ocho bits.

- **Output Enable:** Permite utilizar la funcionalidad tri-state del buffer de salida. Este pin no fue utilizado.

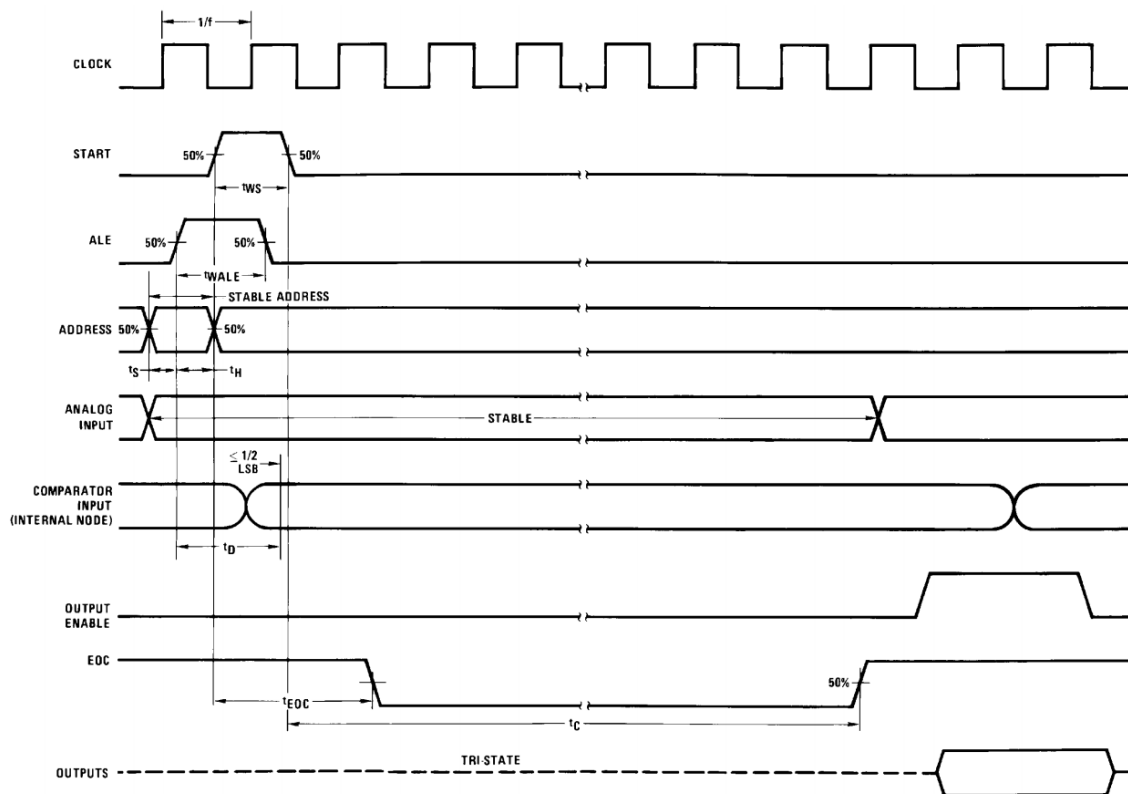


Figura 2: Diagrama temporal de las señales de entrada, salida y control del conversor ADC0808.

0.3. Acondicionamiento de la señal de entrada

0.3.1. Offset y enclavamiento

Dado que la señal que ingresa al ADC0808 debe estar contenida dentro del rango $0V-5.12V$ con un margen de $100mV$, se montó a la señal de entrada sobre un nivel de continua igual a $\frac{5.12V-0V}{2} = 2.56V$ para luego limitar con un circuito enclavador a esta resultante entre los rangos permisibles del conversor. El circuito utilizado se detalla en la Figura (3).

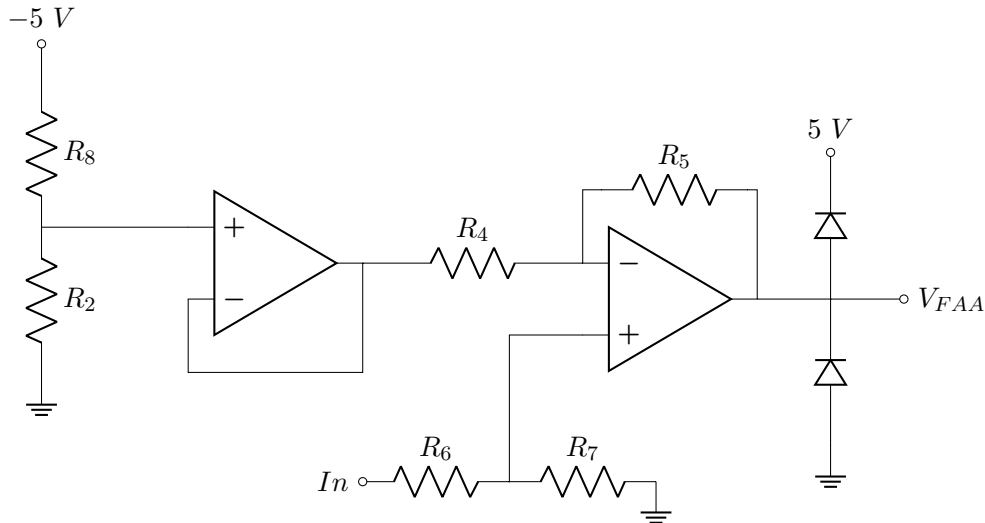


Figura 3: Circuito de acondicionamiento de la señal de entrada.

0.3.2. Sample & Hold

A la hora de realizar el proceso de conversión, el comparador del ADC necesita que la señal de entrada se mantenga estable. Como se vio en la sección anterior, si no se utiliza un Sample & Hold, es necesario que la frecuencia de entrada sea lo suficientemente baja como para que el comparador logre hacer su trabajo sin comprometer la precisión de este.

El S&H seleccionado es el [LF398N](#). Sabiendo que se posee una tasa de conversión máxima de 8.62 kHz , impuesta por el tiempo máximo de conversión del ADC0808 de $116 \mu\text{s}$, lo que implica que

$$T_{acq} = \frac{1}{2 \cdot 8.62 \text{ kHz}} \approx 58 \mu\text{s}$$

De esta forma, observando el gráfico de la hoja de datos mostrada en la Figura (4) del S&H, se requiere un capacitor de 20 nF .

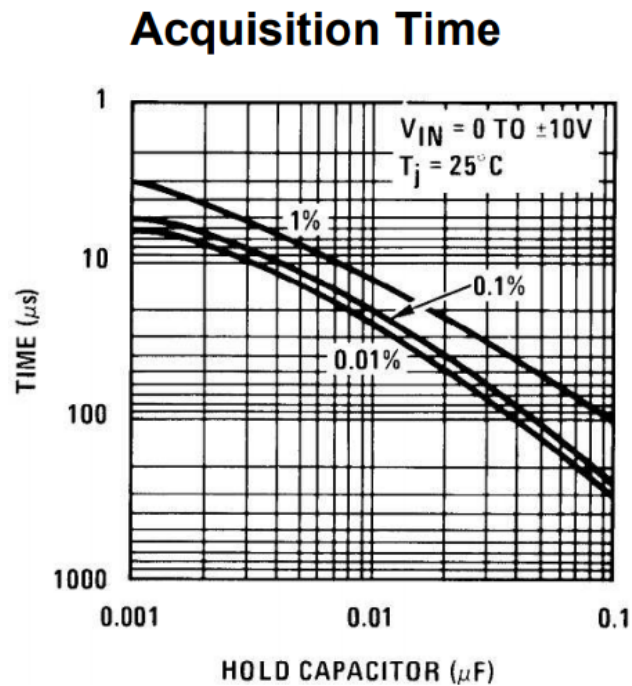


Figura 4: Tiempo de adquisición del S&H en función del capacitor de hold.

0.4. Máxima frecuencia de entrada sin Sample & Hold

De la datasheet del ADC0808, se tiene que si $V_{CC} = V_{REF+} = 5.12V$ y $V_{REF-} = 0V$, la resolución será de $20 \frac{mV}{bit}$. Si se utiliza la frecuencia de clock f_{CLK} típica utilizada en la datasheet de $640kHz$, el tiempo de conversión t_C máximo será de $116\mu s$. Esto implica que la entrada no deberá de tener una pendiente mayor a $\frac{20mV}{116\mu s}$ para no introducir error en la cuantización de la señal.

Si la señal de entrada se encuentra en el peor caso, es decir, con una excursión de tensión de $-0.1V + V_{REF-}$ a $5.12V + 0.1V$; esta se encuentra montada sobre un nivel de continua igual a $(5.22V - (-0.1V))/2 = 2.66V$; y esta se puede considerar senoidal gracias a la teoría desarrollada por Fourier; se tiene que la amplitud pico máxima de la senoidal podrá ser $2.66V$. Luego, asumiendo el peor caso de la pendiente de la senoidal, para un ángulo igual a cero radianes, lo que permite utilizar la aproximación paraxial, se tiene que

$$\left. \frac{d(2.66V \cdot \sin(2\pi f_{in_{max}} t))}{dt} \right|_{t=0} = 2.66V \cdot 2\pi f_{in_{max}} = \frac{20mV}{116\mu s} \quad (1)$$

Finalmente, se obtiene que la componente de mayor frecuencia de la señal de entrada para no comprometer la precisión del ADC0808 deberá ser como máximo

$$f_{in_{max}} = 10.3Hz$$

0.5. Conversor DAC0800

Se utilizó el integrad DAC0800, un conversor D/A de 8 bits con salida diferencial de corriente. Para convertir esta corriente en un nivel de tensión se utilizó el circuito propuesto por la hoja de datos que se muestra a continuación:

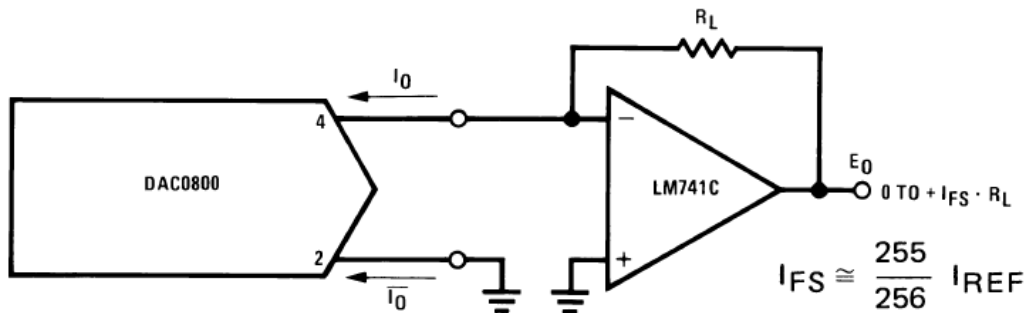


Figura 5: Configuración salida DAC.

La salida va de 0 a $V_{fs} = I_{fs} \cdot R_L$.
En cuanto al pinout del DAC es el siguiente:

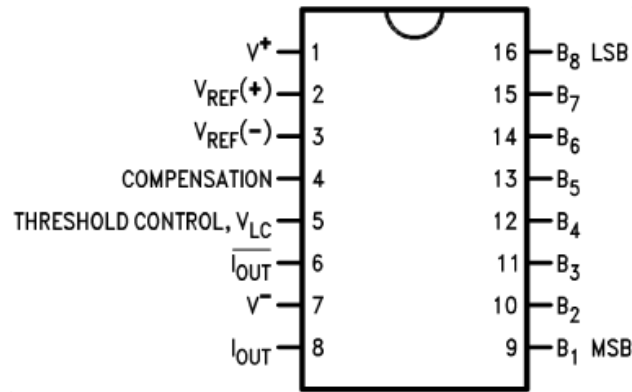


Figura 6: Pinout DAC0800.

Donde los pines desde B1 a B8 son las entradas digitales, siendo B1 el bit mas significativo. En cuanto a la salida, es por corriente, y corresponde a los pines 6 y 8, cabe mencionar que dichas corrientes son complementaria, osea su suma da 0. Los pines 2 y 3 son las tensiones V^+ Y V^- fueron conectadas a 5V y a GND respectivamente, mientras que las tensiones de referencia, si bien dice tensiones, uno debe proveer corrientes de referencia, esto se hace utilizando una resistencia entre vcc y V_{ref+} , al igual que una entre GND y V_{ref-} . En cuanto al pin de threshold fue conectado a masa, finalmente en el pin de comp se conectó un capacitor de 100nF.

0.6. Señal de sincronización de conversión

0.6.1. Restricciones temporales

Si los componentes utilizados tuviesen un tiempo de operación ideal, se podría utilizar un clock tan rápido como se quiera. Sin embargo, esto no sucede. Por parte del integrado ADC0808, se tiene que el tiempo de conversión con una señal de sincronización de $640kHz$ es como máximo $116\mu s$. Luego, el DAC0800 posee un tiempo de estabilización de $100ns$. Por parte del sample & hold utilizado, el LF398N, se tiene que el tiempo de adquisición máximo con un capacitor de hold de $80nF$ es de $20\mu s$. Finalmente, se obtienen dos limitantes en tiempo:

- El ADC0808 tardará como máximo $116\mu s$ en convertir el valor holdeado de la señal analógica en digital.
- El LF398N tardará como máximo $20\mu s$ en cargar el capacitor de hold con el valor de la señal analógica en la fase de sampleo.

Es por esto, que la rapidez máxima del clock que gobierna la frecuencia de conversión del circuito será el doble del mínimo entre la inversa de los dos limitantes temporales, es decir, $\frac{1}{2 \cdot 116\mu s} = 4.31kHz$. Sin embargo, si que quisiese aumentar aún más las frecuencia de conversión del circuito, se podría implementar un clock cuyo duty cycle no sea del 50 %, debido a que mientras el ADC se encuentra convirtiendo, el s&h no posee ninguna restricción temporal más que la de fuga, mientras que si el s&h está sampleando la señal, el ADC no posee ninguna restricción temporal.

Basta con que el duty cycle sea tal que el tiempo de holdeo sea igual al tiempo de conversión del ADC y el tiempo de sampleo se igual al tiempo máximo que tarda el s&h en cargar su capacitor de hold. Se tiene entonces que

$$DT_{optimo} = \frac{20\mu s}{116\mu s} \cdot 100 = 17.24 \% \quad (2)$$

para simplificar el circuito, se tomó $DT = 25 \%$ y la frecuencia de esta señal de sincronización deberá ser como máximo $\frac{1}{116\mu s + 20\mu s} = 7.352kHz$.

0.6.2. Circuito generador de la señal de sincronización

Se partió de un generador de onda cuadrada de $640kHz$, el cual será la señal de clock del ADC0808. Esta señal pasará además por un divisor de frecuencia que divide por 88 veces para obtener una frecuencia menor pero cercana a $7.352kHz$, de $7.272kHz$. Luego, esta señal ingresa a dos divisores de frecuencia en cascada que dividen por dos. Se detalla en la Figura (??) esta operación. Luego, se realiza la operación AND entre la señal antes de pasar por los divisores en frecuencia, y las señales a la salida de cada divisor. Así se obtiene una señal de DT del 25 % independientemente de la frecuencia de la señal original.

Esta señal posterior a la operación AND es la que se utilizará como señal de control del S&H. Finalmente, se quiere enviar un pulso de una duración de al menos $t_{ws} = 200ns$ al pin de start del ADC0808 cuando el S&H comience a holdear, por lo que se utilizará un negador seguido de un detector de flancos para generar esta señal de start. El resultado se detalla en la Figura (8)

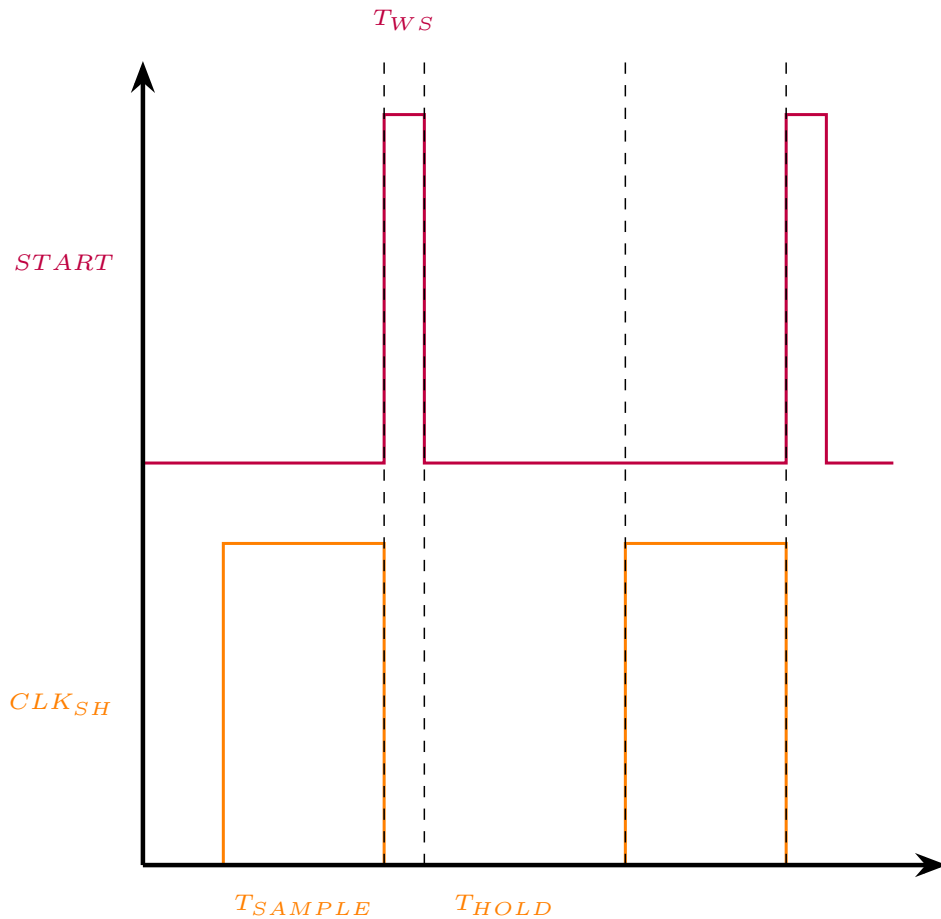


Figura 7: Diagrama temporal de las señales de control del circuito.

Así se obtienen todas las señales de control del circuito de una misma señal de clock original, de tal manera que todas estén sincronizadas entre sí. Si se desea disminuir la frecuencia de conversión basta con disminuir la frecuencia

0.6.3. Circuito oscilador

Se utilizó el circuito mostrado en la Figura (??) para generar la señal de clock de $640kHz$ la cual no solo será ingresada al ADC0808 sino de la cual se generarán las señales de sincronización del S&H y de start del conversor.

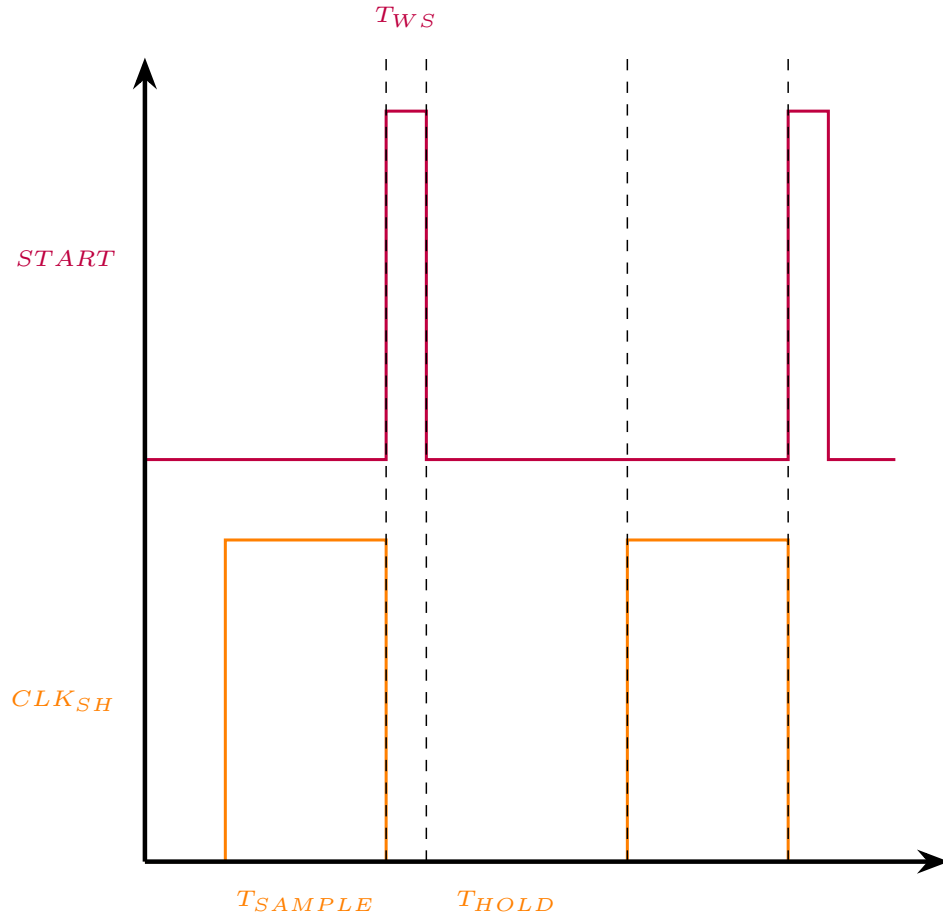


Figura 8: Circuito oscilador.

Utilizando $R_T = 620 + 110\Omega$, $R_{POT} = 1k\Omega$ y $C_T = 1nF$ se logra obtener

$$49kHz < f_{CLK} < 632kHz$$

por lo que la frecuencia de conversión del circuito total será de

$$557Hz < f_{CONV} < 7.2kHz$$