

PWM y ADC con micropython en la Raspberry Pi Pico

Juan Dominguez - 20221005039
Mariana Obando Gutierrez - 20192005009
Julián Esteban Pacheco - 20202015125

Septiembre 2023

1 Introducción

Para la práctica que se planteó para esta ocasión se hace uso del PWM (Pulse Width Modulation) y el ADC (Analog to Digital Converter) con micropython en la Raspberry Pi Pico, con un ejemplo en concreto para controlar la intensidad de un led, el ADC se utilizara para capturar señales analógicas y convertirlas en digitales de 16 bits, con una precisión de 12 bits que pueden ser procesados por la Raspberry Pi Pico.

2 Desarrollo

2.1 Resultados esperados

- Determinar la frecuencia de la señal triangular
- Calcule el Tau del filtro RC y compararlo con la frecuencia del PWM

2.2 Procedimientos

2.2.1 Código

Para hacer uso del PWM y ADC utilizamos el código brindado por el profesor el cual se encuentra en github en PWM y ADC con micropython en la Raspberry Pi Pico

Este código básicamente crea una modulación PWM simple en la que el ciclo de trabajo de la señal PWM se modifica periódicamente según las lecturas del ADC. El ciclo de trabajo PWM varía entre 0 y Dutymax, con un paso de 4000, y vuelve a empezar cuando alcanza el valor máximo.

2.2.2 Desarrollo

1. Para determinar la frecuencia de la señal triangular que entrega el ADC hay que tener en cuenta el ciclo útil del PWM, para eso dividimos 2^{16} entre 4000 y multiplicamos por el periodo de la señal 500 ms

$$\frac{2^{16}}{4000} = 8.192ms$$

Este es el periodo de la señal, para la frecuencia lo dividimos por 1 de tal manera que:

$$\frac{1}{8.192ms} = 122mHz$$

2. Para calcular la constante de tiempo, , en el circuito, debemos multiplicar la resistencia por la capacitancia utilizados en el filtro, las cuales tienen valores de 1 k y $0.47\mu F$ Para la frecuencia lo dividimos por 1 de tal manera que :

$$\frac{1}{0.47ms} = 2.1kHz$$

2.3 Conclusion

Debido a que la frecuencia en el ciclo PWM es de 100kHz y la del filtro es de 2.1kHz evidencia que al momento de hacer una lectura de datos al ser mas grande la frecuencia del PWM que la del filtro hay un error ya que no se alcanzan a tomar todos los datos al momento de graficar