

PROYECTO

Franz López Arredondo 2420191051@estudiantesunibague.edu.co
Maicol Stiven Prada Aguiar 2420191041@estudiantesunibague.edu.co
Juan Camilo Triana Calvo 2420191025@estudiantesunibague.edu.co

• ENUNCIADO

En el presente proyecto se busca diseñar un diseño óptimo para un panel seguidor de luz, en el cual se planea que el artefacto se dirija ante la mayor fuente de luz solar.

• RESUMEN

Al realizar la elaboración del montaje se presentaron problemas con los materiales usados, ya que el peso de estos perjudicaba a los motores, los dobleces que se proporciona al material generan fricción entre ellos haciendo que los motores funcionen con más esfuerzo, por estos problemas se decidió hacer uso de cartón y madera, ahora bien, las conexiones no están exentas de problemas, ya que la Raspberry pi pico solo cuenta con tres conexiones ADC, pines los cuales en primera instancia servirán para conectar las cuatro fotorresistencias, sin embargo, como se nombro anteriormente, solo existen tres, por lo que se decidió cambiar la cantidad de los sensores al número respectivos de los pines ADC, ubicados estratégicamente en forma de “Y” para así agilizar el monitoreo.

• OBJETIVOS ESPECÍFICOS

1. Elaborar códigos que respondan correctamente a lo esperado del proyecto.
2. Hacer el uso correcto de los sensores de luz para el redireccionamiento del prototipo.
3. Diseño de Hardware óptimo para el correcto funcionamiento.

• MATERIALES

1. Cables de conexión.
2. Raspberry pi.
3. Madera de papelería.
4. Carton industrial
5. Servomotores.
6. Fotorresistencias.

• DESARROLLO

Para el desarrollo del mismo el grupo a cargo implementó el uso de cartón industrial y madera de papelería para asegurar un peso liviano, ahora bien se piensa que a partir de la Raspberry programar los motores, fotorresistencias, etc. Con el objetivo de que los motores giren en dirección al panel, el cual contará con 4 sensores ubicados estratégicamente para que el aparato cumpla el movimiento esperado.

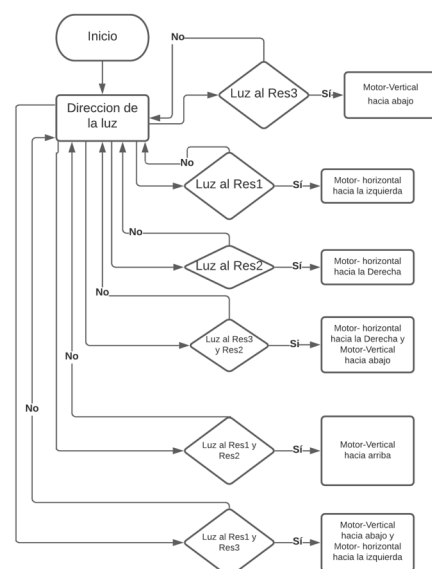


Imagen1. Diagrama de flujo

Como se aprecia en la imagen 1, se muestra el diagrama de flujo que cumplirá el circuito, en

esta se plantean las posibles ocasiones en las que la luz perturbara las resistencias y con esto la acción que tomarán los motores.

Ahora bien, a continuación se presentarán prototipos que se tomaron como ejemplo para la elaboración del artefacto.

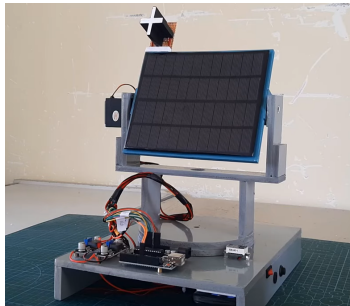


Imagen 2. primer ejemplo (<https://www.youtube.com/watch?v=dP6eO19t2pk&t=1006s>)

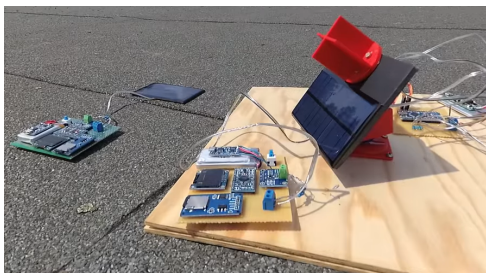


Imagen 3. Segundo ejemplo (<https://www.youtube.com/watch?v=6OJutZfsEs&t=204s>)

Las imágenes anteriormente presentadas son algunas de las que ayudaron a la elaboración del proyecto.

En el primer avance se realizó el montaje de Hardware del seguidor solar, en el cual se implementó uso de cartón industrial y algunas piezas de madera para diseño de las bases del proyecto, a su vez, se hace uso de dos servomotores, los cuales se encargaran individualmente de manejar ya sea el eje "X" o el eje "Y". Se tuvo en cuenta el diagrama mostrado en la parte inferior para realizar las conexiones correspondientes entre los servomotores y la Raspberry Pi. Luego de realizar algunos ajustes en el Software para ejecutar el funcionamiento del seguimiento, se realizaron las primeras pruebas parciales del proyecto, dichas pruebas se presentan en video,

el video se encuentra en los anexos del documento.

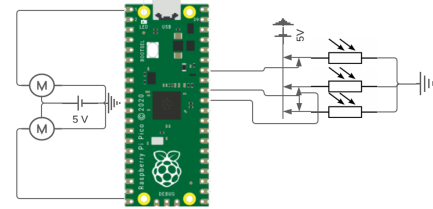


Imagen 4. esquemático

Como se aprecia en la imagen.4 se hace uso de la raspberry pi pico, en la cual se realizó el uso de las 3 entradas Análogas/Digitales, ubicando en estas las fotorresistencias para dar orientación a los motores.

```
# Asignar las fotorresistencias En los pines De ADC
ldr1 = ADC(28) # Fotorresistencia superior verde
ldr2 = ADC(27) # Fotorresistencia derecha naranja
ldr3 = ADC(26) # Fotorresistencia izquierda cafe

#Asignar los pines y limites conectados a el servomotor Horizo
servoh = PWM(Pin(13)) # Asignacion del pin PWM
servoh.freq(50) #Establecer frecuencia a trabajar
servohLimitHigh = 2500000
servohLimitLow = 27777
servoh_contador=1250000 # iniciar en 2
#Asignar los pines y limites conectados a el servomotor Vertical
# servov = 145
servov = PWM(Pin(14))
servov.freq(50)
servovLimitHigh = 2500000 # limite superior del angulo
servovLimitLow = 1666666 # limite inferior del angulo
servov_contador=1944444 # iniciar en 1455
```

Imagen 5. Primera parte del código

En el primer bloque del código se asignaron los pines para los servomotores y los sensores usados, se inicializa la frecuencia de trabajo para los servomotores y como último se declararon las variables con sus límites límites de orientación.

```
while True:
    leer1 = ldr1.read_u16() #leer arri
    leer2 = ldr2.read_u16() #leer arri
    leer3 = ldr3.read_u16() #leer arri
    sleep_ms(70)
    #print(leer1)
    #servov.duty_ns(servov_contador) #se mueve
    servov_contador=servov_contador+27777 # 1 grado hacia arriba
    #print('servov_contador'+servov_contador)
    if (servov_contador >= servovLimitHigh):
        servov_contador = servovLimitHigh
    elif (promedio_arri < leer3):
        servov.duty_ns(servov_contador)
        servov_contador = servov_contador-27777 # 2
        #print('servov_contador'+servov_contador)
        if (servov_contador <= servovLimitLow):
            servov_contador = servovLimitLow
    else:
        servov.duty_ns(servov_contador)
    if (promedio_de > promedio_iz):
        servov.duty_ns(servov_contador) #se mueve
        servov_contador=servov_contador+27777 # 1 grado hacia arriba
        #print('servov_contador'+servov_contador)
        if (servov_contador >= servovLimitHigh):
            servov_contador = servovLimitHigh
    elif (promedio_de < promedio_iz):
        servov.duty_ns(servov_contador)
        servov_contador = servov_contador-27777 # 2
        #print('servov_contador'+servov_contador)
        if (servov_contador <= servovLimitLow):
            servov_contador = servovLimitLow
    else:
        servov.duty_ns(servov_contador)
```

Imagen 6. Segunda parte del código

Ya por último, en la imagen.6, la cual corresponde al segundo bloque del código se lleva a cabo el algoritmo realizado para dar las instrucciones a los servomotores según los valores analógicos que marcan las fotorresistencias y así mismo obtener una dirección donde se presente más luz. En los anexos se comparte el código funcional. Las siguientes imágenes corresponden al prototipo terminado.

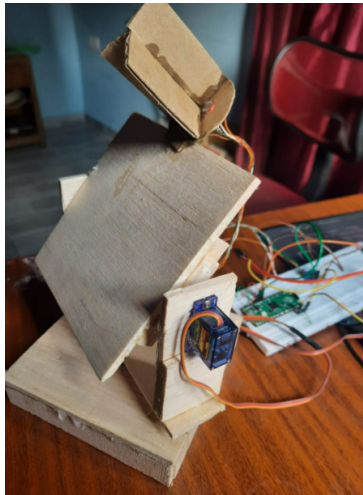


Imagen 7. Finalización



Imagen 8. Acabado

● Anexos

1. [FinalVideo_1636873829.254001.mov](#)
2. <https://github.com/franz-b11/Seguidor-solar/blob/main/CodigoFuncional.py>
3. [Funcionamiento del proyecto](#)