

## Trabajo Informática

### Curso 19/20 - Grupo A109

#### Datos

Inés Portilla de Birazel

55409

i.portilla@alumnos.upm.es

#### Título y resumen

##### Persiana automática con Arduino

Sistema electrónico conformado por un pequeño motor DC que provoca el giro de un carrete cuya cuerda está enganchada a la polea de la persiana, para poder subirla automáticamente. Para tal efecto, el motor se activa con un sensor lumínico cuando este detecta un nivel de iluminación correspondiente al del amanecer. Esta comunicación entre el sensor (entrada) y el motor (salida) se efectúa a través de un microcontrolador de Arduino.

##### Requisitos funcionales

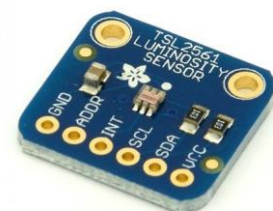
1. El proceso se inicia cuando el sensor detecta una intensidad lumínica equivalente a un amanecer (600 lux aproximadamente).
2. El sensor estará enviando los datos al microcontrolador y cuando se alcanza la intensidad deseada, este acciona el motor.
3. El motor sigue girando hasta que se alcanza un límite de tiempo definido en el código (el tiempo que tarda el motor en subir la persiana).

#### Hardware – Fundamentos técnicos

##### Detección de luz solar - Sensor de luz TSL2592 High Adafruit para Arduino

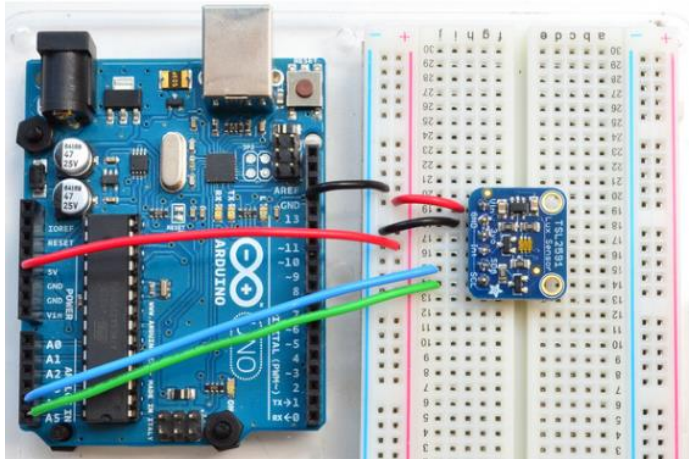
Un sensor de luz permite leer y hacer cálculos con la luz que detecta. Tiene varias aplicaciones, entre ellas en alumbrados públicos automáticos, sistemas de seguridad, como regulador de brillo en las pantallas de aparatos electrónicos (ordenadores y móviles, por ejemplo), e incluso como medidores de luz en las cámaras. Las ventajas de este tipo de sensores son su disponibilidad de formas y tamaños, su bajo coste, son sencillos de programar e incluso necesitan poco voltaje para funcionar (funcionan con 5V de potencia que puede proporcionar el microcontrolador de Arduino Uno). No obstante, los diodos de los sensores lumínicos son sensibles a las temperaturas y son unidireccionales.

En lo que concierne a este sensor en particular, su rango de medición de intensidad lumínica que abarca de 188 uLux a 88,000 Lux y soporta una temperatura de  $-30$  a  $80^{\circ}\text{C}$ . Además, al incluir diodos infrarrojos y de espectro completo, es capaz de hacer mediciones en varios espectros de luz por separado, incluyendo el de luz visible. Funciona con un voltaje entre 3,3 y 5V. En comparación con otras células fotovoltaicas de bajo coste (como las de cadmio) esta realiza mediciones



más precisas, permitiendo así ser configurado para distintos rangos de tiempo y ganancia. Tiene varios pines de funcionamiento, unos de potencia (para encenderlo y para regular su voltaje) y otros de lógica I2C (pines clock y data).

Se conecta al microcontrolador de la siguiente manera:



Y para configurarlo hace falta ingresar el siguiente código:

```
// Configuración de la ganancia y el tiempo de integración para el TSL2591

void configureSensor(void)
{

// El tiempo de integración determina el tiempo durante el cual se detecta la luz. Los tiempos largos de integración son útiles en situaciones
// donde se trabaja con poca luz

tsl.setTiming(TSL2591_INTEGRATIONTIME_600MS);    // Máximo tiempo de integración

//Mostrar la ganancia y el tiempo de integración:

Serial.println("-----");

Serial.print ("Gain: ");

tsl2591Gain_t gain = tsl.getGain();

switch(gain) {

case TSL2591_GAIN_LOW: Serial.println("1x (Low)"); break;

case TSL2591_GAIN_MED: Serial.println("25x (Medium)"); break;

case TSL2591_GAIN_HIGH: Serial.println("428x (High)"); break;

case TSL2591_GAIN_MAX: Serial.println("9876x (Max)"); break;

}

Serial.print ("Timing: ");

Serial.print((tsl.getTiming() + 1) * 100, DEC);

Serial.println(" ms");

Serial.println("-----");

Serial.println("");

}
```

En este proyecto, la tarea del sensor será medir la luminosidad que llega a la ventana de la habitación durante el amanecer. Para medirla, se emplea el siguiente código:

```

void
{
    simpleRead(void)

    uint16_t x = tsl.getLuminosity(TSL2591_VISIBLE);

    Serial.print("[ ");
    Serial.print(millis());
    Serial.print(" ms ");
    Serial.print("Luminosity: ");
    Serial.println(x, DEC);
}

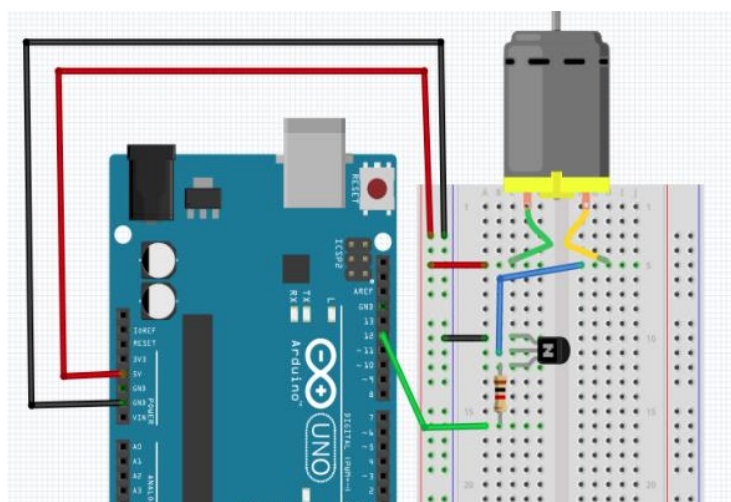
```

## Funcionamiento del carrito – Mini motor DC

Un motor de corriente continua es un aparato que puede convertir energía eléctrica en mecánica, realizando un movimiento rotatorio. Consta de dos partes: una parte fija compuesta por un electroimán, que es puesto en marcha por dos campos magnéticos de polaridades opuestas situados en el extremo al cual se conecta la corriente. A su vez, estos dos campos inducen la fuerza necesaria sobre la parte móvil para hacerle girar, aplicando un torque. Esta otra parte está compuesta por varios espirales o bobinas, que al girar hacen rotar una protuberancia metálica. Gracias a su versatilidad (por su fácil control, paro y automatización) resulta útil en contextos industriales y domésticos. Por ejemplo, se encuentran motores DC en maquinillas eléctricas, en los limpiaparabrisas e incluso en motores de coches.

El motor que se usará en concreto para este proyecto es un mini motor reductor 10:1 de alta potencia. Tiene un voltaje nominal de 6 voltios (aunque funciona bien con los 5V de voltaje que le proporciona el microcontrolador Arduino Uno), su velocidad de salida es de 3000 rpm y tienen un consumo sin carga de 120mA. Las cualidades de este motor son su resistencia a las temperaturas y a la abrasión y su gran capacidad de carga. En particular, la protuberancia (mediante la cual se hará girar el carrito) está hecha con acero de alta calidad, alta dureza y resistencia.

Para poder manipular este motor mediante Arduino es necesario el uso (aparte de cables, por supuesto) de un resistor y un transistor. Un resistor impide parcialmente el flujo de electrones dentro de la corriente eléctrica, y un transistor es un dispositivo semiconductor que permite entregar una señal de salida en respuesta a una de entrada. Para combinar todos estos aparatos se monta el circuito como se muestra en el diagrama:



Y se activa con este código:

```

const int base=12;

void setup() {
  pinMode(base,OUTPUT);
}

void loop() {

  digitalWrite(base,HIGH); // Se enciende el transistor

  delay(30000); // Tiempo que gira

  digitalWrite(base,LOW); //Apagado del transistor

  delay(100000000); //Tiempo que para

}

```

## Referencias

Adafruit Industries. (2018). *Adafruit TSL2591 High Dynamic Range Digital Light Sensor*[Ebook (1st ed., pp. 3-18). Retrieved from

[https://www.cetronic.es/sqlcommerce/disenos/plantilla1/seccion/producto/DetalleProducto.jspx?idIdioma=&idTienda=93&codProducto=999334022&cPath=1343&gclid=Cj0KCQiA-biyBRCCcARIsAFboWg2jEmebQvp0diffuzgg05AEinYSC8SN61KRFypkhhpYK0qUIXu4z-MaAphUEALw\\_wcB](https://www.cetronic.es/sqlcommerce/disenos/plantilla1/seccion/producto/DetalleProducto.jspx?idIdioma=&idTienda=93&codProducto=999334022&cPath=1343&gclid=Cj0KCQiA-biyBRCCcARIsAFboWg2jEmebQvp0diffuzgg05AEinYSC8SN61KRFypkhhpYK0qUIXu4z-MaAphUEALw_wcB)

CREA Club de robótica ETSIDI. (2019). *Transistores Puente H*. Presentation, Madrid.

El mejor Reductor para motor eléctrico de 2020 | Las20Mejores.com. (2020). Retrieved 21 March 2020, from <https://las20mejores.com/mejor-reductor-para-motor-electrico/>

Fritz, C. (2017). Definition of a DC Motor. Retrieved 21 March 2020, from <https://sciencing.com/definition-of-a-dc-motor-13409319.html>

Mini Motor Reductor Metalico 10:1 Alta Potencia. Retrieved 21 March 2020, from <https://www.electan.com/mini-motor-reductor-metalico-101-alta-potencia-p-3092.html>

Motor de corriente continúa; tipos y partes | Tercesa. (2019). Retrieved 21 March 2020, from <https://tercesa.com/noticias/motor-de-corriente-continua-tipos-y-partes/>