



# Control programado con Arduino

## Práctica 8: O relé (Control magnético de potencia)

Ata o de agora, todas as nosas prácticas controlaron pequenos diodos LED que consomen moi pouca corrente. Con todo, no mundo real, os automatismos deben activar motores, calentadores ou lámpadas que funcionan a voltaxes elevadas. Hoxe aprenderemos a utilizar un **relé**, un interruptor mecánico controlado por magnetismo que nos permite gobernar grandes cargas eléctricas de forma segura e illada.

## Obxectivos de Aprendizaxe

---

Ao rematar esta sesión serás capaz de:

- Comprender o principio de funcionamento electromecánico dun relé.
- Diferenciar fisicamente os terminais dun relé: Bobina, Común (COM), Normalmente Aberto (NO) e Normalmente Pechado (NC).
- Diseñar un circuíto con illamento galvánico separando a etapa de control (baixa potencia) da etapa de potencia.
- Programar secuencias temporais para a activación e desactivación dun relé magnético.

## Compoñentes Necesarios

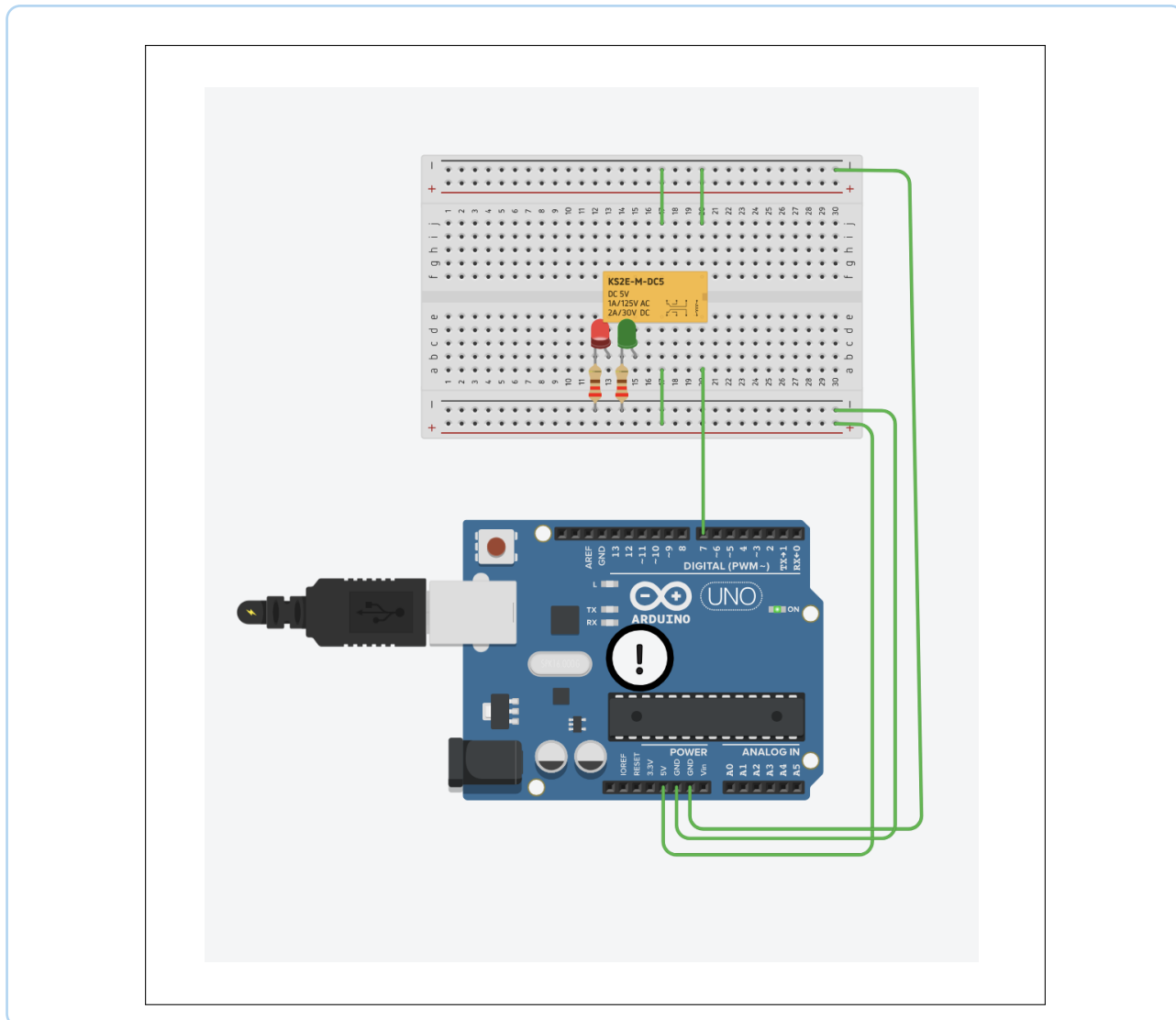
---

Busca e coloca os seguintes compoñentes na túa mesa de traballo de TinkerCad:

- 1 Placa Arduino Uno.
- 1 Placa de prototipado pequena (Protoboard).
- 1 Relé SPDT de 5 Voltios (en TinkerCad figura como "Relé SPDT").
- 1 Diodo LED (para simular a carga que imos acender).
- 1 Resistencia de  $220\ \Omega$  (protección do LED).
- Cables de conexión virtuais.

## Esquema de montaxe na Protoboard

O relé actúa como un interruptor físico. A corrente que sae de Arduino só se usa para activar un electroimán interno (bobina). O circuío que acende o LED é completamente independente.



## Código Base: Activación Intermitente

Escribe o seguinte programa no editor de TinkerCad. Verás que programar un relé é idéntico a programar un LED: só temos que enviar un sinal dixital HIGH ou LOW polo pin correspondente.

```
1 // Asignamos o Pin Dixital 7 para controlar a bobina do rele
2 int pinRele = 7;
3
4 void setup() {
5     // O pin do rele funciona como SAIDA (subministra enerxía á bobina)
6     pinMode(pinRele, OUTPUT);
7 }
8
9 void loop() {
10    // ESTADO 1: Activamos o rele (atrae o contacto mecanico cara a 'NO')
11    digitalWrite(pinRele, HIGH);
12    delay(3000); // Mantemos o rele activado durante 3 segundos
13
14    // ESTADO 2: Desactivamos o rele (o muelle interno devolve o contacto a 'NC')
15    digitalWrite(pinRele, LOW);
16    delay(3000); // Mantemos o rele desactivado durante 3 segundos
17 }
```

Listing 1: Código para activar e desactivar un relé ciclicamente.

## Como funciona o código?

Cando inicias a simulación en TinkerCad, escoitarás un tremelique visual e verás como a agulla interna do relé cambia de posición:

- **Lóxica do Relé:** Ao executar `digitalWrite` `HIGH`, o pin 7 envía 5V á bobina do relé. Isto xera un campo magnético que atrae a lámina metálica do terminal **Común (COM)**, conectándoa mecanicamente co terminal **Normalmente Aberto (NO)**. Nese instante, a corrente flúe cara ao LED e este acéndese.
- **O peirao de retorno:** Ao facer `digitalWrite` `LOW`, a bobina queda sen electricidade, o magnetismo desaparece e un peirao interno empuxa a lámina metálica de volta ao terminal **Normalmente Pechado (NC)**, apagando o LED de forma instantánea.

### Protección Eléctrica na Realidade

Aínda que no simulador TinkerCad conectamos o relé directamente ao pin de Arduino, en circuitos reais **nunca** se debe alimentar a bobina dun relé directamente desde un pin do microcontrolador. As bobinas consomen máis corrente da que Arduino pode proporcionar de forma segura e xeran picos de alta tensión ao apagarse. Nunha montaxe real utilízase un transistor (como interruptor intermedio) e un diodo de protección en paralelo (diodo volante ou flyback).

## O Reto da Práctica 8

Domina o uso de conmutadores electromecánicos resolvendo os seguintes retos prácticos:

1. **Indicador de Estado Dobre:** Engade un segundo LED (por exemplo, de cor Vermella) ao circuíto. Fai as conexións necesarias para que o LED Verde estea no terminal **NO** e o LED Vermello no terminal **NC**. Sen tocar o código base, observa que sucede ao iniciar a simulación. Como se comportan os LEDs?
2. **Interruptor de Potencia Remoto:** Integra un pulsador (como o da Práctica 3) no teu circuíto de TinkerCad. Modifica o código para que o relé permanezca inactivo (apagado) e **só se active de forma sostida mentres manteñas premido o pulsador**.
3. *Pregunta para reflexionar:* Por que cres que a este compoñente se lle asocia co concepto de "illamento galvánico"? Que ocorrería se a carga do circuíto de potencia sufrise un cortocircuíto grave? Veríase afectada a placa de Arduino?