

Comedero automático inteligente para gatos (ESP8266)

adelgadocriado@gmail.com • [Linkedin](#) • [GitHub](#)

Objetivo

Diseñar desde cero una PCB funcional que controle un comedero automático mediante un ESP8266, usando sensores IR, un motor paso a paso y buzzer. Proyecto realizado como puente entre automatización industrial y diseño electrónico embebido.

Especificaciones

El sistema se basa en un **ESP8266** con conectividad WiFi, un **sensor infrarrojo** para detectar al animal y un **motor paso a paso** que acciona un **tornillo sin fin** para dispensar el alimento, acompañado de un **buzzer** como alerta sonora. La PCB, de **dos capas** y diseñada en **EasyEDA**, recibe **12V de entrada**, que se convierten a **5 V** para los periféricos y **3.3 V** para el microcontrolador..

Para facilitar la programación, se incorporaron dos pulsadores físicos: uno para **BOOT (GPIO0)** y otro para **RESET**, además de un **interruptor (switch)** para seleccionar modos de funcionamiento o activar funciones específicas del sistema.

Diseño técnico

El esquemático se basó en un diseño existente, pero fue redimensionado y adaptado en función de la disponibilidad de componentes. Utilicé **EasyEDA** como herramienta principal, ya que permite diseñar directamente con el stock de **LCSC**, lo que me ayuda a reducir costes y asegurar la disponibilidad de los componentes durante la fabricación.

En cuanto al diseño físico de la placa, implementé técnicas como **copper pour** y el uso de **polígonos de cobre** para mejorar la disipación térmica y reducir el ruido EMI. El plano inferior está completamente conectado a **GND**, proporcionando un retorno limpio para las señales. Además, se recortó la zona de la antena del ESP8266 para optimizar la señal WiFi.

También presté atención al serigrafiado de la **capa superior**, añadiendo referencias claras para cada componente, aunque quedó pendiente marcar adecuadamente los pines **GND** y **3.3 V** del conector de programación TTL, algo que mejorará en la próxima iteración.

Errores cometidos (y solucionados)

- **GPIO15 mal configurado:** se conectó erróneamente a 3.3 V cuando debía ir a GND (es un strapping pin).
→ Corregido desoldando y redirigiendo la conexión mediante la resistencia **R7**.
- **Falta de serigrafía en pines de programación:** no se marcaron los pines **GND** y **3.3 V** en el conector TTL, dificultando la identificación durante las pruebas.
→ A corregir en la próxima versión del diseño.
- **Error en el circuito del buzzer y del LED indicador:** se conectaron incorrectamente, lo que impidió que el LED señalizara correctamente la actividad del microcontrolador.
→ Detectado durante las pruebas funcionales; se ajustará la lógica en la próxima iteración.

Montaje y programación

- Montaje manual de la placa y programación del ESP8266 mediante conversor USB–TTL.
- El siguiente paso es desarrollar una **aplicación móvil** para conectarse al sistema por WiFi.
- Programar la lógica que controle la **detección y la alimentación del gato** desde el móvil.
- Rehacer el esquema con una **alimentación única a 5 V** para simplificar el diseño.
- Añadir **test points** para facilitar la depuración.
- Considerar el uso de un **ESP32** para futuras ampliaciones de funcionalidad.
- Modularizar el diseño por **bloques funcionales** para hacerlo más escalable y reutilizable.

Perfil profesional

Estoy en proceso de reconversión profesional de la automatización industrial al diseño de hardware, sistemas embebidos y electrónica digital. Este proyecto es el primero de muchos.