Comedero automático inteligente para gatos (ESP8266)

adelgadocriado@gmail.com •Linkedin • GitHub

🧭 Objetivo

Design a fully functional PCB from scratch to control an automatic pet feeder using an ESP8266, infrared sensors, a stepper motor, and a buzzer. This project serves as a personal bridge between industrial automation and embedded electronics design.

⚙️ Specifications

The system is based on an **ESP8266** with integrated WiFi connectivity, an **infrared sensor** to detect the animal, and a **stepper motor** that drives a **worm screw** to dispense food, along with a **buzzer** for audio alerts. The **two-layer PCB**, designed in **EasyEDA**, takes a **12 V input**, which is regulated down to **5 V** for peripherals and **3.3 V** for the microcontroller.

To facilitate programming, two physical push buttons were added: one for **BOOT (GPIO0)** and another for **RESET**, along with a **switch** to select operating modes or activate specific system functions.

🧩 Technical Design

El The schematic was based on an existing design but was resized and adapted according to component availability. I used **EasyEDA** as the main tool since it integrates directly with **LCSC's** component stock, helping reduce costs and ensuring availability during manufacturing.

For the physical layout, I applied techniques such as **copper pour** and **polygon fills** to improve heat dissipation and reduce EMI. The bottom layer of the PCB is a full **GND plane**, which provides clean signal return paths. Additionally, the **ESP8266 antenna zone** was partially cut out to improve wireless performance.

II also paid close attention to the **top-layer silkscreen**, clearly labeling each component. However, one improvement for the next revision is to properly mark the **GND and 3.3 V pins** on the TTL programming connector.

🛠️ Issues Encountered (and Resolved)

- **Misconfigured GPIO15**: This strapping pin was mistakenly connected to **3.3 V** instead of **GND**.
  → Resolved by desoldering and rerouting via **resistor R7**.

- **Missing silkscreen for programming pins**: The **GND and 3.3 V** pins were not labeled on the TTL header, making setup harder during testing.
  → To be fixed in the next PCB iteration.

- **Buzzer and LED miswiring**: The **buzzer** and the **activity LED** were not connected properly, preventing the LED from indicating the microcontroller's status.
  → Identified during testing; logic will be corrected in the next version.

🔧 Assembly, Programming & Next Steps

- Manual PCB assembly and ESP8266 programming using a **USB–TTL converter**.
- Next step: develop a **mobile app** to connect to the system via WiFi.
- Implement logic to **detect the animal** and **trigger feeding** from the mobile app.
- Redesign the circuit to use a **single 5 V power input** for simplicity.
- Add **test points** to assist with debugging.
- Evaluate upgrading to an **ESP32** for extended functionality.
- Refactor the system into **modular blocks** for better scalability and reusability.

💼 Perfil profesional

I am currently transitioning from **industrial automation** to **hardware design**, **embedded systems**, and **digital electronics**.
This project marks the **first of many** on my path toward a more electronics-focused career.