ESP32 LVGL Stepper Motor Control System

Yatharth Bhalerao, Bhumika Kartar

How to Cite this Article:

Kartar, B. & Bhalerao, Y. (2026). ESP32 LVGL Stepper Motor Control System. International Journal of Creative and Open Research in Engineering and Management, 02(05). https://doi.org/10.55041/ijcope.v2i4.715

https://doi.org/10.55041/ijcope.v2i4.715

Abstract:

This report presents the design, implementation, and evaluation of a two-node embedded system for precision stepper motor positioning controlled via a touchscreen graphical user interface (GUI). The master node — an ESP32-2432S028R (Cheap Yellow Display, CYD) — hosts an LVGL v9.1 UI on a 2.8-inch 320x240 TFT for motion setpoint entry and real-time position feedback display. The slave node is an Arduino Uno mounted with a CNC Shield v3, which provides a dedicated stepper driver socket (A4988 or DRV8825) and end-stop connector for a limit switch. The slave executes trajectory control using the AccelStepper library via the CNC Shield's pre-routed STEP/DIR signals, performs automatic homing at start-up, and streams position data over UART using ArduinoJson. The mechanical stage uses a NEMA17 stepper motor coupled to a T8 lead screw (2 mm pitch) with custom 3D-printed mounts, achieving a theoretical resolution of 0.01 mm/step in full-step mode.

Keywords: ESP32, LVGL, GUI, Arduino Uno, CNC Shield v3, A4988, DRV8825, AccelStepper, UART, JSON, ArduinoJson, Homing, T8 Lead Screw, NEMA17, PlatformIO

Introduction:

Precise linear positioning is fundamental to CNC machining, 3D printing, laboratory automation, and pick-and-place systems. This project implements a master-slave architecture that decouples the human-machine interface (HMI) from real-time motion execution. The master node provides an LVGL-based touchscreen interface for motion parameter entry and live position monitoring. The slave node — an Arduino Uno fitted with a CNC Shield v3 — executes motion profiles with acceleration/deceleration ramps, enforces hardware homing via the shield's end-stop connector, and streams real-time position data back to the master.

Choosing the Arduino Uno with CNC Shield v3 for the slave provides several engineering advantages: the shield pre-routes STEP, DIR, and ENABLE signals to the driver socket, provides dedicated end-stop connectors, and is electrically isolated from the motion supply through the shield's power connector. This simplifies wiring and improves reliability in a motion-control context.

The mechanical stage consists of a NEMA17 stepper motor coupled to a T8 lead screw (2 mm pitch) with custom 3D-printed mounts. Communication between master and slave uses a JSON protocol over UART, serialized with the ArduinoJson library on both nodes.

Project details:

The system consists of two microcontroller boards communicating over a dedicated UART link at 115,200 bps using JSON-encoded messages. The architecture separates UI rendering from motion execution, allowing each node to be optimized independently.

Key subsystems:

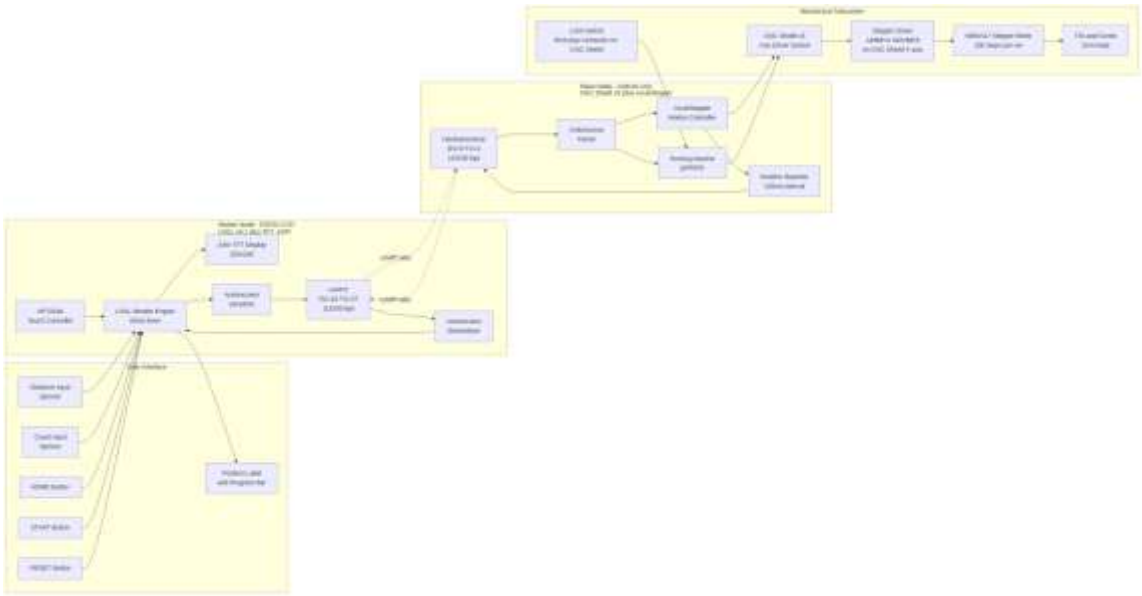
- Master Node (ESP32-CYD): Renders LVGL UI on a 320x240 TFT, processes XPT2046 touch input, serializes commands to JSON over UART1, and updates position/progress widgets from slave feedback.
- Slave Node (Arduino Uno + CNC Shield v3): Receives JSON commands over Hardware Serial (RX=0, TX=1), drives the stepper driver via the shield's X-axis STEP=2/DIR=5 pins, reads the end-stop pin for homing, and sends position telemetry back to the master.
- CNC Shield v3: Plugs directly onto the Arduino Uno headers. Provides stepper driver sockets (X/Y/Z/A axes), end-stop connectors (X+/X-/Y+/Y-/Z+/Z-), spindle/fan headers, and a power input terminal for the motor supply.
- Transport: UART at 115,200 bps. Master uses UART1 (RX=22, TX=27 on ESP32). Slave uses Hardware Serial (RX=0, TX=1 on Arduino Uno).
- Mechanics: NEMA17 + T8 lead screw (2 mm/rev) with 3D-printed mounts and backplate.

Conceptual Signal Chain:

```
Touchscreen -> XPT2046 -> LVGL UI -> ArduinoJson -> UART1 (ESP32)
  --[UART wire]--> Hardware Serial (Arduino Uno)
  -> ArduinoJson -> AccelStepper -> CNC Shield -> A4988/DRV8825 -> NEMA17 -> T8 Screw
  <- (position feedback) <- UART <- AccelStepper position query
  -> LVGL Label / Progress Bar on master TFT
```

Block Diagram:

The block diagram illustrates the full system architecture: user interface widgets on the master CYD, JSON-over-UART transport, the Arduino Uno + CNC Shield slave node, A4988/DRV8825 driver, NEMA17 motor, T8 lead screw, and limit switch end-stop.



Conclusion:

This project demonstrates a practical, low-cost embedded motion control system using commercial-off-the-shelf hardware. The ESP32-CYD master provides a professional-quality touchscreen HMI via LVGL, while the Arduino Uno with CNC Shield v3 delivers a clean, well-routed motion node with minimal wiring effort.

The CNC Shield v3 eliminates per-project STEP/DIR wiring and provides industry-standard stepper driver socket compatibility with A4988 and DRV8825 modules. The JSON-over-UART protocol is lightweight, human-readable, and easily extensible for additional commands or axes.

Potential future enhancements include: multi-axis motion (using additional CNC Shield driver sockets Y/Z/A), closed-loop feedback with a rotary encoder, G-code command interpreter on the Arduino Uno, OTA firmware updates on the ESP32 over Wi-Fi, and persistent motion profiles stored in EEPROM on the Uno.

Reference:

[1] LVGL Documentation. https://docs.lvgl.io/ (accessed April 2026).

[2] Airspace Systems. AccelStepper Library. http://www.airspayce.com/mikem/arduino/AccelStepper/
[3] Benoit Blanchon. ArduinoJson Documentation. https://arduinojson.org/
[4] Bodmer. TFT_eSPI Library. https://github.com/Bodmer/TFT_eSPI
[5] Paul Stoffregen. XPT2046_Touchscreen. https://github.com/PaulStoffregen/XPT2046_Touchscreen
[6] Espressif Systems. ESP32 Technical Reference Manual. https://www.espressif.com/
[7] SquareLine Studio. LVGL UI Editor. https://squareline.io/
[8] CNC Shield v3 Schematics. https://blog.protoneer.co.nz/arduino-cnc-shield/