

High-Volume Memory Device Programmer

Custom Hardware Design

Embedded Firmware

Desktop Application Development

Manufacturing Tooling

CLIENT

A US-based company.

THE PROBLEM

Programming small memory devices one at a time during manufacturing didn't scale — if the client needed dozens programmed per batch and the process took even a minute each, that's an unworkable bottleneck on a production line. They needed something that could program many devices essentially at once, controlled from a PC, and — just as importantly — catch faulty or already-used parts automatically, because a bad part that slips through becomes a much more expensive problem downstream.

WHAT WE BUILT

We designed a programmer built around a removable carrier board with sockets for many memory devices at once, connected to a PC over a standard serial interface. The PC application is where the actual workflow lives: it can program devices individually or drive the whole carrier board at once, and after programming, it automatically verifies every single device — checking for parts that are blank, already used, or simply faulty, and flagging them rather than letting them continue down the line. Each batch gets its own configuration file, so there's a record of exactly what was programmed, when, and with what result.

WHAT IT DOES

- ✓ Programs dozens of memory devices at once using a removable, socketed carrier board
- ✓ Entire programming workflow — individual or batch — controlled from a PC application
- ✓ Programs a full batch of devices in under a minute, removing the one-at-a-time bottleneck
- ✓ Automatically verifies every device after programming and flags blank, already-used, or faulty parts
- ✓ Generates a configuration file per batch, giving manufacturing a record of what was programmed and the result for each part

OUTCOME

The client's manufacturing line went from a one-at-a-time bottleneck to programming and verifying dozens of devices per minute, with automatic fault detection catching bad parts before they moved further down the line.