

# embino\_

## Tiny intelligence for tiny devices

A middle-level programming stack that brings safe, deterministic logic to ESP32, Arduino, and any \$1 microcontroller.

Patent Pending (US 63/927,859)

// what we do

# A miniature GPT-style language model that runs on **\$1 microcontrollers**

We fill the gap between **massive cloud-based LLMs** that can't run on devices, and  
**tiny C++ code** that's painful to write.

100 GB

Cloud LLMs

16 KB

Embino

512 B

Raw C++

Natural language in → Verified bytecode out → Deterministic execution on-device

// the problem

# Programming microcontrollers is stuck in 1985

## Current Reality

- Hand-written C/C++
- Low-level memory management
- Device-specific registers
- Manual GPIO configuration
- Painful debugging
- Zero abstraction for intent

## Existing "Solutions"

- **Python/MicroPython:** Heavy runtime, unpredictable
- **LLMs:** Can't run on MCUs, nondeterministic
- **Arduino libraries:** Still low-level C++

**Gap:** No middle-level abstraction between natural language and C code that is **safe, deterministic, compact**, and runs on **\$1 microcontrollers**

// the abstraction gap

# LLMs broke the pattern

Every abstraction gain was **proportional** — until LLMs required **6 million times more RAM** than C++

Layer	RAM Usage	Runs on MCU?
Machine Code	64 B	✓ Yes
Assembly	96 B	✓ Yes
C / C++	2 KB	✓ Yes
MicroPython	96 KB	⚠ Barely
<b>LLM (GPT-4)</b>	<b>100 GB</b>	<b>✗ Cloud Only</b>
<b>Embino</b>	<b>16 KB</b>	<b>✓ Yes</b>

// the solution

# Express intent, not implementation

Embino is a **middle-level language** for embedded systems. Write rules in structured English. Get deterministic bytecode that runs on any microcontroller.

```
When motion is detected
  after 22:00
  and brightness < 30%
then turn on lamp
  for 120 seconds.
```



Human rules



DSL + Validation



Bytecode



MCU  
execution

# Three components, one stack

## 01 — The Language

A tight, rule-oriented DSL for events, conditions, timers, and actions.  
Expressive enough for real behavior.  
Constrained enough to verify.

- Events & conditions
- Thresholds & timers
- Actions & state

## 02 — The Translator

An offline model that converts your rules to DSL. It *rejects* ambiguous inputs — never guesses, never hallucinates.

- Ambiguity detection
- Conflict checking
- Static verification

## 03 — The Runtime

A tiny interpreter in C that runs on ESP32, Arduino, RP2040. Deterministic. Bounded. No surprises.

- <64 KB Flash
- <16 KB RAM
- <20 ms latency

// specifications

# Datasheet

## Performance Targets

INTERPRETER\_FLASH  
 $\leq 64 \text{ KB}$

RUNTIME\_RAM  
 $\leq 16 \text{ KB}$

EVENT\_LATENCY  
 $\leq 20 \text{ ms}$

TRANSLATION\_ACCURACY  
 $\geq 80\%$

EXECUTION\_MODEL  
Deterministic

## Target Hardware

- **ESP32** — Primary platform
- **Arduino Uno/Nano** — ATmega328P
- **RP2040** — Raspberry Pi Pico
- **Any MCU** with 16KB+ RAM

## Use Cases

- Smart lighting & automation
- Sensor alerts & monitoring
- Robot behavior logic
- Edge autonomy

// market opportunity

# Massive underserved market

**30-40B**

MCU units shipped / year

**\$20-30B**

IoT tooling market

**\$1.2T**

Semiconductor market by 2030

## Beachhead: Makers & Education

- 10M+ Arduino/ESP32/Pico boards sold yearly
- Hobbyists frustrated with C++ complexity
- STEM educators need accessible tools
- Robotics clubs, hackathons, workshops

## Scale: Industrial & Automotive

- Appliances: HVAC, washers, coffee machines
- Industrial: PLCs, sensors, factory automation
- Automotive: subsystems, ECUs, ADAS
- Every vertical still codes in C/C++

Even **0.1%** of MCU deployments = **30-40 million units/year** addressable



# Three-stage hardware roadmap

## Stage 1 — Add-On Module (12-24 months)

**Small external board** attaching via UART/I<sup>2</sup>C/SPI. Houses Embino runtime. Bridges old workflows with new system. Zero hardware changes to existing boards.

**Revenue:** Dev boards, makers, robotics startups, schools

## Stage 2 — System-on-PCB (24-36 months)

**Embino integrated on the same board** as MCU. Integrated interpreter. Standard pinout (Arduino/Pico style). Ready for industrial prototyping.

**Revenue:** Volume dev-board sales, robotics kits, automation integrators

## Stage 3 — On-Board Chip (36+ months)

**Custom MCU with Embino baked in.** ROM-resident interpreter. Bootloader integrated with toolchain. Custom SoB variants for OEMs.

**Revenue:** Licensing to appliance manufacturers, industrial OEMs

# 12-month technical feasibility

**Goal:** Prove technical feasibility and commercial interest (\$300k, 12 months)

## Objectives

- **DSL + Safety Model** — Define language, grammar, static semantics
- **LLM Translator** — Build pipeline with ambiguity detection
- **Micro-VM** — Implement interpreter on ESP32/Arduino
- **User Validation** — Study with 8-10 participants

## Deliverables

- DSL spec document
- Reference interpreter (Python)
- LLM-assisted translator prototype
- Micro-VM firmware (ESP32, Arduino)
- 3 demo applications with metrics
- User study report
- 2+ letters of support

**Success:**  $\geq 80\%$  translation accuracy • All demos within resource limits • Users complete tasks faster

# The next abstraction layer for 30 billion chips

## Why This Matters

Every appliance, vehicle, and industrial machine runs MCUs programmed in 50-year-old languages. As AI transforms software development, embedded systems remain untouched — **the largest category of compute still waiting for its abstraction revolution.**

## End State

- Embino runtime in every MCU
- Natural language → verified bytecode
- Standard for edge device behavior
- OEM licensing across verticals

## Protected Innovation

- Grammar-constrained code generation
- Formal safety model with bounded resources
- Deterministic execution guarantees
- NL→DSL→bytecode→MCU pipeline

Patent Pending (US 63/927,859) — 35 claims

## No Direct Competition

MicroPython: too heavy. Arduino: still C++. LLMs: can't run on-device. No one combines NL input with verified, deterministic MCU execution.

# Leadership & Organization



**David H. Silver**

CEO / Founder

→ Transitioning to CTO when CEO hired

T

**CEO**

HIRING

Business strategy, fundraising

**CTO**

David H. Silver

Technical vision, architecture

**COO**

HIRING

Finance, operations

## Founder Background

- 6 years embedded ML at Apple & Intel RealSense
- 15 patents (depth sensing, ML, medical imaging)
- 12 publications: Nature, IEEE TPAMI, PNAS
- Co-founder/CTO Embryonics (acquired 2023)

## Experience Logos



intel

Nature



Kernel Keys LLC — Founded 2024, NY

# Ready to bring intelligence to tiny devices

**Seeking:** SBIR Phase I funding (\$300k)

**12**

months to POC

**3**

working demos

**10+**

user validation

**David H. Silver**

david@embino.com

embino.com • kernel-keys.com • dhsilver.me