David H. **Silver**

ACADEMY → INDUSTRY → COMMERCIALIZATION
NY, USA

+1-929-782-2060 | ✉ david@embino.com | 🏠 embino.com | 💼 davidsilveril

Current Venture: Embino
embino.com — Founder & Inventor   *Stealth mode*

Natural-language programming for embedded devices. A grammar-constrained small language model generates verified bytecode that runs on ESP32, Arduino, RP2040, and any $1 microcontroller. No hallucinations, no cloud dependency, deterministic execution.

Status: Patent filed, website live, seeking seed investment to build production toolchain.

Embedded ML & Real-Time Experience

ML RESEARCHER — APPLE INC.   *2016 – 2019*
Developed embedded ML algorithms for depth sensing and signal processing in hardware products. Focused on real-time inference, power-constrained optimization, and ASIC-compatible implementations. Cross-disciplinary work spanning sensors, ML, and hardware/software co-design.

ALGORITHM ENGINEER — INTEL CORPORATION (RealSense)   *2013 – 2016*
Computer vision algorithm engineer in the RealSense depth sensing group. Developed solutions from theoretical R&D to ASIC-compatible implementations with strict real-time constraints. Work on coded light systems, edge detection, and range reconstruction for consumer hardware.

Startup Leadership

CO-FOUNDER & CTO — EMBRYONICS (ACQUIRED BY Rhea Labs 2023)   *2019 – 2023*
Led fertility-tech startup. Built ML pipeline from vision to geometric networks, from research to clinical deployment.

HEAD OF AI/ML — AKA Foods   *2021 – 2023*
Led AI R&D for plant-based food development. Built generative AI system for recipe optimization integrated with food lab processes.

Patents (15 granted/pending)

Embino: US 63/927,859 — Grammar-Constrained Code Generation for Embedded Devices (2025). **Intel:** US09800795B2 — Active Illumination Depth Camera; US09792671B2 — Coded Light Depth; US20170178305A1 — Edge Filters; US10775501B2 — Range Reconstruction; US10540784B2 — Texture Camera Calibration. **Apple:** US11914078B2 — Depth Calibration; US12153140B2 — Visual Inertial Odometry. **Embryonics:** WO2022009186A1 — Embryo Implantation Prediction; US20220284542A1 — Semantic Medical Imaging; US20220375069A1 — Oocyte Quality Estimation. **AKA Foods:** US20230288392A1 — Food Processing, Mixture Modeling, Molecular Embedding, Virtual Tasting.

Publications (12 peer-reviewed)

Nature (2016): Mid-Developmental Transition and Animal Body Plans. *PNAS* (2012), *PNAS* (2018): Cyanobacteria Viruses; Gene Regulatory Programs. *IEEE TPAMI* (2019): Intel RealSense SR300 Coded Light Depth Camera. *MIDL* (2020, first author): Data-Driven Prediction of Embryo Implantation. *Human Reproduction* (2022): Embryologist Agreement on Blastocyst Implantation. *LNCS/AIiH* (2024): Bonna Algorithm for Embryo Implantation. *PLOS ONE* (2025, sole author): Taskmaster UK Format Analysis. *Molecular BioSystems* (2012, first author): Evolutionary Transcriptomics. *Bioinformatics* (2013, first author): ELOPER Genome Assembly.

Education & Academic Foundation

Microsoft Research PhD Fellowship — Cambridge, UK (2011–2013). Computational biology and high-throughput data analysis in collaboration with Technion and MSR Cambridge.

Technion — Israel Institute of Technology — Rothschild Scholars Program for Excellence. Research assistantships: Computational Evolution & Development Lab under Prof. Itai Yanai (2010–2013, led to Nature and PNAS publications); Evoked Potentials Laboratory under Prof. Hillel Pratt (2008–2010, neural signal processing); VR & Neurocognition Lab under Prof. Miriam Reiner (2009–2010, brain-computer interfaces); Intelligent Systems Lab under Prof. Michael Lindenbaum (2007–2009, computer vision and ML).

Current Roles

Head of AI/ML — Rhea Labs (2023–present): AI-powered diagnostics & patient journey management following Embryonics acquisition.

Head of AI/ML — Canotera (2023–present): Lead AI initiatives for legal dispute outcome prediction.

embino_

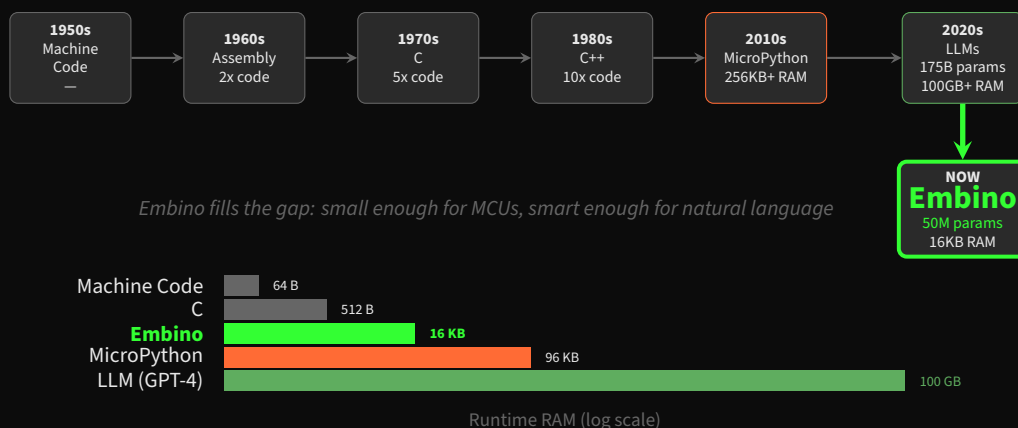Tiny intelligence for tiny devices

Problem → Solution

Problem: 30B+ MCUs ship annually, all programmed in C/C++. LLMs can generate code but can't run on MCUs (100GB+ RAM, hallucinations, cloud dependency). No middle abstraction exists.

Solution: Embino is a natural-language programming stack for microcontrollers. Write rules in structured English → get deterministic byte-code on any $1 chip. Three components: (1) rule-oriented DSL, (2) grammar-constrained LM (<50M params, 100% syntactic validity), (3) micro-interpreter (<16KB RAM, <20ms latency).

The Abstraction Gap

Programming abstractions evolved proportionally—until LLMs broke the pattern, jumping to 175B+ parameters for tasks that run on 16KB chips.

```
1950s          1960s          1970s          1980s          2010s          2020s
Machine        Assembly       C              C++            MicroPython    LLMs
Code           2x code        5x code        10x code       256KB+ RAM     175B params
—                                                                          100GB+ RAM
                                                                              │
                                                                              ▼
                                                                           NOW
                                                                           Embino
                                                                           50M params
                                                                           16KB RAM
```

Embino fills the gap: small enough for MCUs, smart enough for natural language

```
Machine Code    ▮ 64 B
C               ▮▮ 512 B
Embino          ▮▮▮▮ 16 KB
MicroPython     ▮▮▮▮▮ 96 KB
LLM (GPT-4)     ▮▮▮▮▮▮▮▮▮▮ 100 GB
```
Runtime RAM (log scale)

Why Now / Competitive Moat

Why now: Grammar-constrained decoding and efficient LoRA adaptation only became viable in 2023-24. Small language models (<100M params) now match GPT-3 on structured tasks. The timing is right.

Competition: MicroPython (too heavy, 96KB+), block-based tools (no AI, fixed templates), cloud LLMs (can't run on MCU). No one combines NL→DSL→verified bytecode→deterministic MCU execution.

Moat: 35-claim patent covering the full stack. First-mover in grammar-constrained embedded code generation.

Market & Business Plan

Market: 30+ billion MCUs ship annually—all programmed in C/C++. Target: makers/education → robotics/automation → industrial OEMs. Even 0.1% = 30M+ units/year.

Business model: Toolchain licenses (freemium → pro), dev board sales, OEM licensing fees for embedded runtime.

Three-Stage Roadmap: **Stage 1** (12-24mo): Add-on module via UART/I²C for existing boards. Dev boards, makers, schools. **Stage 2** (24-36mo): System-on-PCB with integrated interpreter. Volume sales, robotics kits. **Stage 3** (36mo+): Custom SoC with ROM-resident runtime. OEM licensing.

Investment Opportunity

Seeking $500K seed round to build production toolchain and Stage 1 hardware prototype.

Use of funds: DSL spec + reference interpreter, GC-SLM translator, micro-VM firmware (ESP32/Arduino/RP2040), first add-on module design, partner outreach.

- **Product-first:** Real toolchain for real hardware—not a research benchmark
- **Strong IP:** 35-claim provisional patent covering the full stack
- **Massive market:** 30B+ MCUs/year still using 50-year-old programming
- **Founder fit:** 6 years embedded ML at Apple/Intel, 15 patents (depth sensing, ML, medical imaging), startup exit, Nature publication