

Arjun Manikandan

Coppell, TX | 214-604-0861 | axm5507@tamu.edu | [Portfolio](#) | [Linkedin](#) | [Github](#)

EDUCATION

Texas A&M University

August 2025 – May 2028

Bachelor of Science in Honors Computer Science | GPA: 3.75/4.0

College Station, TX

- **Coursework:** Computer Organization, Program Design, Data Structures and Algorithms, Programming Languages, Discrete Mathematics, Linear Algebra, Principles of Statistics, Calculus 3
- **Awards:** Engineering Honors, Dean's Honor Roll

EXPERIENCE

Software Development Intern

May 2026 – August 2026

The REACH Project

College Station, TX

- * Shipped app supporting **2K+** users to iOS App Store using **React Native**, **TypeScript**, and **Firebase** backend
- * Architected efficient **Firestore** data schema and application structure across **4** collections and **30+** screens
- * Reduced backend operations by **50%** by redesigning access patterns and implementing denormalized read models
- * Built **Dockerized Azure-deployed** CRM with **5** custom **PHP** modules, **4-role RBAC**, and KPI dashboard

Data Science Consultant

Jan. 2026 – May 2026

American Airlines

College Station, TX

- * Built ML pipeline predicting sequence spoilage, engineering **31** features from **9k+** structured data/NLP records
- * Tuned **LightGBM**, **XGBoost**, **CatBoost** across **150** trials, reaching **0.823** ROC AUC and **0.747** Macro F1
- * Delivered model interpretability via **SHAP** and isotonic calibration, lifting spoiled class recall to **2.0x** baseline
- * Presented results and operational recommendations to AA executives, winning **1st** place for best project

PROJECTS

Game Boy Emulator | Rust

June 2026 – July 2026

- * Engineered Sharp LR35902 CPU supporting **500+** instructions, interrupt dispatch and per-instruction timing
- * Built graphics pipeline rendering **160x144** frames at **60 FPS**, implementing scanline timing, sprites, and layering
- * Architected modular memory bus routing all reads/writes by address across all the **5** hardware subsystems
- * Implemented **3** Memory Bank Controllers supporting bank-switched ROM/RAM, saves, and RTC emulation

Cache Simulator | C++, Google Test

June 2026 – July 2026

- * Built CPU cache simulator modeling **L1/L2** hierarchies, **3** replacement policies, and **4** write-policy combinations
- * Implemented **64-bit tag/index/offset decomposition**, set-associative lookup, and dirty-bit management
- * Applied **Strategy Design** pattern to support **LRU**, **FIFO**, **Random** replacement without modifying core logic
- * Validated simulator with **50+** automated tests, reproducing **0-98%** cache hit rates across **16k** memory accesses

C Compiler | Rust

June 2026 – July 2026

- * Developed C compiler in **Rust** translating C source into **x86-64 assembly** through **7-stage** compiler pipeline
- * Implemented recursive-descent parser, semantic analyzer, and AST supporting functions, pointers, and structs
- * Generated **System V AMD64** assembly for register allocation, stack frames, calls, and pointer dereferencing
- * Engineered calling convention with **6-register** parameter parsing and **16-byte** stack alignment enforcement

ScratchGPT | Python

April 2026 – July 2026

- * Completed a full GPT model with training loop, text generation, KV-cache, and grouped query attention
- * Built core NLP fundamentals: word embeddings, preprocessing, sentiment analysis, positional encoding
- * Implemented self-attention, multi-head attention, and full transformer blocks, reducing validation loss by **35%**
- * Built BPE/character tokenizer with **5k+** vocab plus batched dataset pipeline processing **2M** tokens

TECHNICAL SKILLS

Programming: , C/C++, Rust, Python, Java, Go, Zig, Verilog, Assembly, JavaScript, TypeScript, SQL, Shell

Frameworks: React, Node.js, Flask, Django, Spring Boot, Angular, Vue, Laravel, TensorFlow, Langchain

Tools: Linux, Amazon Web Services, Azure, Git, Docker, Kubernetes, MySQL, PostgreSQL, MongoDB, Wireshark