

ELECTRICAL ENGINEERING · MINOR IN ROBOTICS · UTEP

Santiago Delgado

I design circuits, write the firmware that runs on them, and test them on the bench. Looking for **embedded systems, automotive, and hardware** engineering roles.

915-494-8460 santdelg5995@gmail.com www.linkedin.com/in/santiago-d santidelgado.com

EDUCATION

B.S. Electrical Engineering, Minor in Robotics

The University of Texas at El Paso · Aug 2024 – Dec 2027 · GPA 3.16/4.0

Study abroad: Spain, Japan

Coursework: Microprocessor Systems, Digital Logic Design, Software Design, Operating System Design

EXPERIENCE

Undergraduate Researcher

Power and Renewable Energy Systems Lab, UTEP · May 2026 – Present

Service Manager

Great American Steakhouse · Aug 2024 – Apr 2026 · Supervised a 20-person staff across 3 shifts, 1,000+ customers weekly

SKILLS

LANGUAGES	C, Embedded C++, MATLAB, Python
EMBEDDED	UART, ADC, PWM, Timer (SysTick)
HARDWARE	Analog circuit design, digital logic design, hardware debugging, oscilloscope
TOOLS	Code Composer Studio (CCS), Xilinx Vivado, Microsoft Office, VSCode

LEADERSHIP

Treasurer, IEEE

Apr 2026 – Present

External Affairs Officer, IEEE RAS

Robotics and Automation Society · Apr 2026 – Present

Operations Chair, SHPE

Society of Hispanic Professional Engineers · Apr 2024 – Present

SELECTED WORK

	PROJECT	DATES	KEY RESULT	
01	Load forecasting and grid research	May 2026 – Present	0.93% MAPE, beating linear regression by 77%; presented to 120 attendees at COURI 2026	2
02	Seizure detection embedded system	Apr – May 2026	2-module MSP432 system with checksummed UART link and 3 ADC sensitivity levels	3
03	Traffic light controller (Verilog FSM)	Oct – Dec 2025	4-state FSM, simulated in ModelSim and run on a Basys-3 FPGA	4
04	Grocery inventory system	Nov – Dec 2025	13-option C app with dynamic memory, sorting, receipts, and binary storage	5
05	ECG signal processing system	Aug – Oct 2025	3-stage op-amp circuit filtering the heart signal to 0.5–100 Hz	6

01 / RESEARCH · POWER SYSTEMS · MAY 2026 – PRESENT

Load forecasting and grid research

At the PRES Lab, I applied K-means clustering and **4 forecasting models** to ERCOT load data, reaching **0.93% MAPE** and beating linear regression by **77%**. I now model a **39-bus grid** to study voltage-collapse risk.

MY ROLE

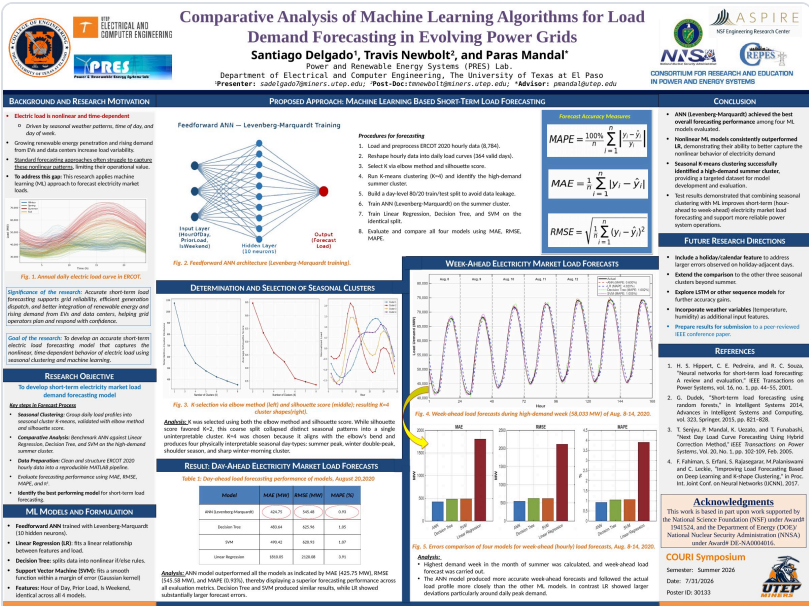
Undergraduate Researcher, Power and Renewable Energy Systems Lab

TEAM

With Travis Newbolt · advisor Dr. Paras Mandal

TOOLS

MATLAB · Python · PyPSA · pandapower · K-means · neural networks



COURI research poster. Full-size version linked below.

GOAL

Forecast grid demand accurately, then test how large new loads, like data centers, stress the grid.

WHAT I DID

- Used **seasonal K-means clustering** to isolate a high-demand summer group for focused testing.
- Trained a **10-neuron feedforward neural network** with Levenberg-Marquardt.
- Benchmarked it against linear regression, decision trees, and an SVM using hour of day, prior load, and weekend status.
- Presented the research to **120 attendees at COURI 2026**, which led to 2 project proposals.
- Built and validated a **39-bus model in PyPSA and pandapower** as a baseline for a voltage-collapse risk study.
- Generated a year of **data-center load profiles**, sized at 25–30% of system load, to show voltage-stress scenarios.

RESULT

0.93%

MAPE

77%

better than linear regression

120

attendees at COURI 2026

39-bus

grid model for voltage study

Neural network MAE: 424.75 MW vs. 1,810.05 MW for linear regression. Research led to 2 project proposals.

SKILLS SHOWN

Time-series data

Model benchmarking

Power-flow modeling

Python

MATLAB

Technical presenting

FILES [Full-size poster](#) [MATLAB scripts and results](#)

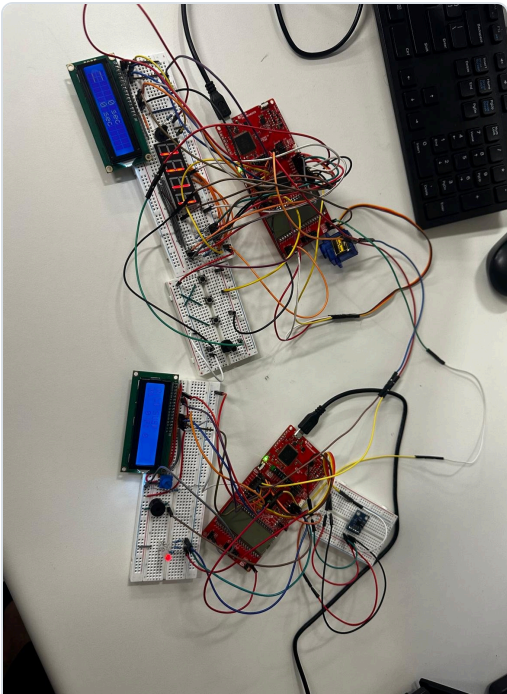
Forecasting results as reported on the COURI poster, not rerun for this portfolio. The scripts need the original dataset and saved MATLAB workspaces to run. The poster is a presentation, not a peer-reviewed paper.

02 / EMBEDDED SYSTEMS · APR – MAY 2026

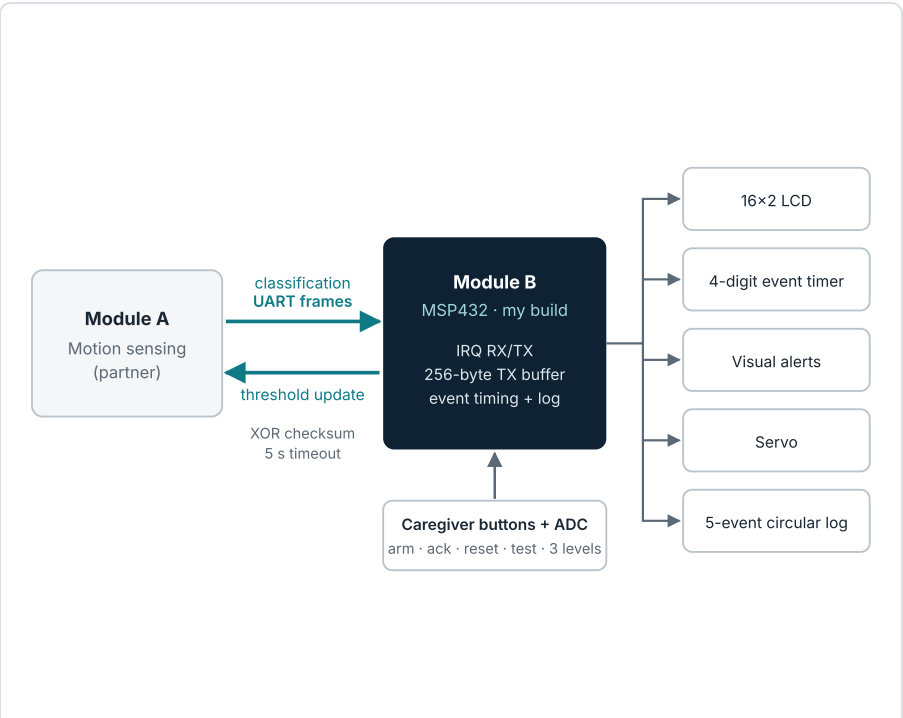
Seizure detection embedded system

A **2-module seizure-alert system** on MSP432 microcontrollers that coordinates sensing, alarm, and display. I built **Module B**, which takes motion classifications over a **checksummed UART link** and turns them into timed alerts and caregiver controls.

MY ROLE	TEAM	TOOLS
Module B, the alarm and caregiver interface	2 people · Daniel Mendoza built Module A	Embedded C · TI MSP432 · UART · 14-bit ADC · SysTick



Bench build. Module A (top) and Module B (bottom) on MSP432 boards.



System view. My module (dark) sits between the sensing board and the caregiver.

GOAL

Alert a caregiver quickly and reliably when the sensing board reports a seizure-like event, and let the caregiver tune sensitivity.

WHAT I DID

- Wrote a framed UART protocol with **XOR checksums** and interrupt-driven receive and transmit, using a 256-byte transmit buffer.
- Added a **5-second timeout** so Module B flags a lost link instead of failing silently.
- Read a 14-bit ADC to select **3 sensitivity levels** and sent new thresholds back to Module A.
- Drove a 16×2 LCD, a multiplexed 4-digit timer, software-generated servo pulses, and a 5-event circular log.
- Built arm, disarm, acknowledge, reset, and self-test controls for the caregiver.

RESULT

2 boards
talking over one checked serial link

3
sensitivity levels set by caregiver

5 s
link-loss timeout

5
caregiver controls incl. self-test

Working two-board classroom prototype, tested with simulated events.

SKILLS SHOWN

Serial protocols

Interrupts

Fault detection

ADC

HW/SW integration

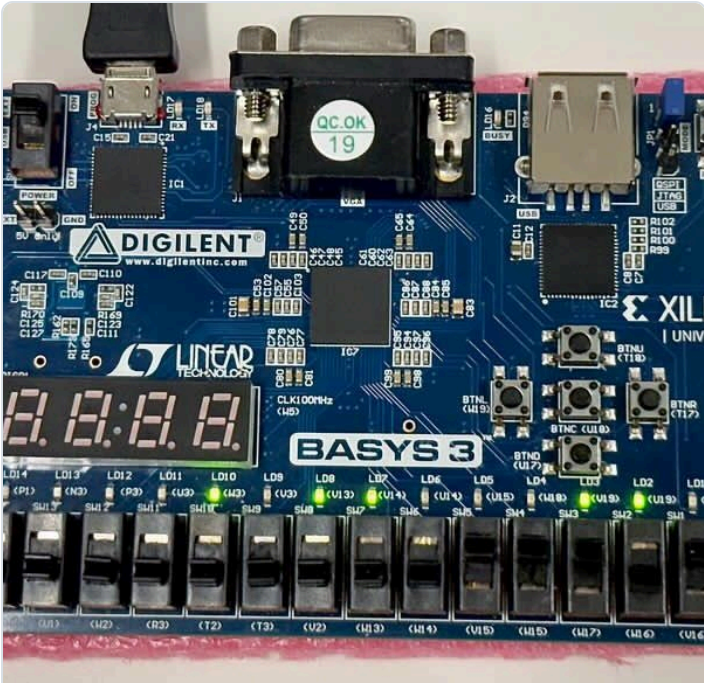
FILES [Module B source code](#) [main.c](#)

Classroom prototype; no clinical validation claimed. UART and SysTick use interrupts; buttons are polled and servo timing uses software delays.

Traffic light controller (Verilog FSM)

I designed and simulated a **4-state finite-state machine** in Verilog for a 2-direction traffic intersection, then ran it on a **Basys-3 FPGA** using only the board's LEDs, button, and 4-digit display.

MY ROLE Solo: design and implementation	TEAM Individual project	TOOLS Verilog · Xilinx Vivado · ModelSim · Basys-3 FPGA · XDC constraints
--	----------------------------	--



Basys-3 board. No external parts; all lights and the countdown use on-board hardware.

GOAL

Run a safe, timed crossing sequence from a single button press using only the FPGA board's built-in hardware.

WHAT I DID

- Designed a **4-state FSM** in Verilog with timed transitions, and simulated it in **ModelSim**.
- On a button request, moved vehicles to **yellow** while the pedestrian STOP light flashes.
- Switched to red and gave pedestrians a **10-second countdown** on the 4-digit display.
- Returned STOP and turned vehicle lights off at the end of the countdown.
- Debugged synthesis errors** and fixed timing constraints in the **XDC file** for reliable FPGA implementation in Vivado.

RESULT

4

FSM states

10_s

walk countdown

1

button input

0

external components

Coordinates user input, timing, and display output in one design.

SKILLS SHOWN

Verilog / HDLFSM designTimingFPGA I/O

ModelSimVivado / XDC

Light sequence after a button press

Button press

Pedestrian asks to cross

Vehicles: yellow

Pedestrian STOP flashes · ~3 s

Vehicles: red · Pedestrians: walk

10 s countdown on 4-digit display

Countdown ends

STOP returns · vehicle lights off

Behavior.

One button and a counter drive the 4-state FSM.

ONLINE [Interactive demo of the light sequence](#)

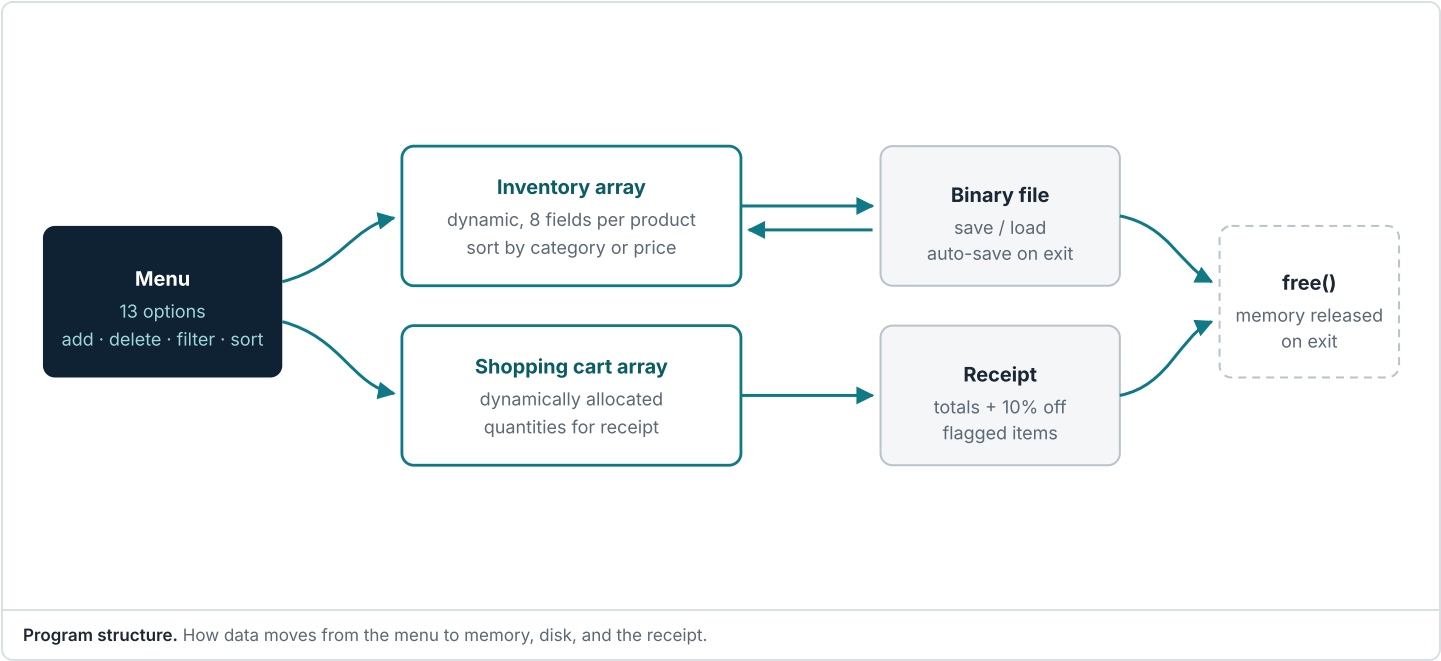
The original Verilog source was lost in a computer change, so code is not linked. The website has an interactive model of the light sequence.

04 / SOFTWARE · NOV – DEC 2025

Grocery inventory system

I wrote a command-line inventory and checkout app in **C** that manages its own memory, sorts and filters products, and saves everything to a **binary file**.

MY ROLE Solo: design and all code	TEAM Individual project	TOOLS C · structs · malloc / free · binary file I/O · sorting
--------------------------------------	----------------------------	--



GOAL

Build a store inventory and checkout tool that keeps data between runs and handles a product list of any size.

WHAT I DID

- Defined a product **struct with 8 fields**, including category, source, price, stock, serial number, and discount flag.
- Stored inventory and cart in **dynamically allocated arrays** instead of fixed-size tables.
- Wrote add, delete, category filter, and **sort by category or price**.
- Added **binary save and load** so inventory persists between runs.
- Generated receipts with quantities, totals, and a 10% discount on flagged products.

RESULT

13
menu options

10%
auto discount on flagged items

8
fields per product

2
dynamic arrays: inventory and cart

Saves inventory on exit and frees all allocated memory.

SKILLS SHOWN

C

Memory management

File I/O

Data structures

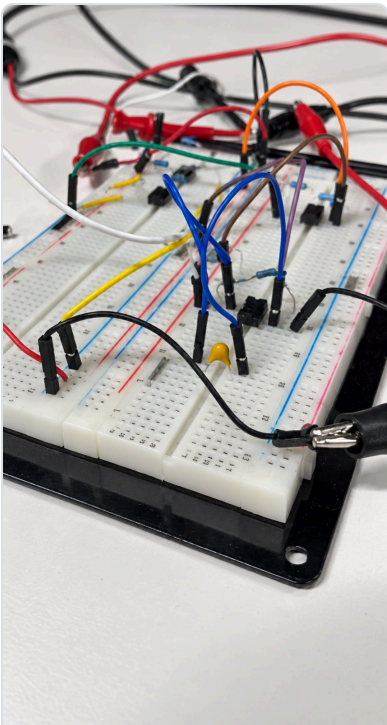
Sorting

05 / ANALOG ELECTRONICS · AUG – OCT 2025

ECG signal processing system

I designed a **3-stage analog circuit** that picks up a heart signal with an instrumentation amplifier and uses op-amp filtering to keep the **0.5–100 Hz** band, then converted the output to digital for processing.

MY ROLE Design calculations, part selection, assembly, and filtering	TEAM 2 people · Leila Armstrong led testing and troubleshooting	TOOLS Op-amps · instrumentation amplifier · RC filters · breadboard · oscilloscope
--	---	--



Breadboard build. AI-restored photo; originals linked.

Signal chain

I owned the stage calculations, part values, and filter integration.

Design. Filter, amplify, then filter again, keeping 0.5–100 Hz.

GOAL

Pick up a heart signal that is tiny and noisy at the skin and make it large and clean enough to digitize and process.

- WHAT I DID**
- Calculated the amplifier stages and **selected component values**.
 - Assembled the **instrumentation amplifier** stages on a breadboard.
 - Added **high-pass input** and **low-pass output** filtering to keep the **0.5–100 Hz** range.
 - Converted the analog output to **digital** for processing.

RESULT

3

stages: HPF, in-amp, LPF

0.5–100 Hz

signal band kept

Working breadboard circuit with the output trace captured on the oscilloscope (scope photo linked). Built for Electronics II.

- SKILLS SHOWN**
- Op-amp design

Filter design

Biosignals

Bench work

Oscilloscope

FILES [Original circuit photo](#) [Oscilloscope photo](#)

The featured photo is an AI-restored version of the Electronics II photo. Gain and CMRR are not listed because measured values were not recorded.