ELEC 327 (Spring 2025)
Micro-Climate Sensor Node

Tomi Kuye Bryce Gibson

19 May 2025

1 Project Concept

Houston's live-oak canopy is anything but uniform: sun flecks, trapped humidity, and wind tunnels create distinct *micro-climates* even on neighbouring branches. Our node is designed to clip onto a branch, measure three ambient variables, and back-haul the data in real time.

Key functional requirements

- We initially set out to capture **temperature**, **relative humidity**, and **light** at *5 s cadence*.
- Transmit each sample set over LTE Cat-M1 so no campus Wi-Fi is needed.
- Provide a local read-out for installers (no laptop while up the ladder).

2 Hardware Design

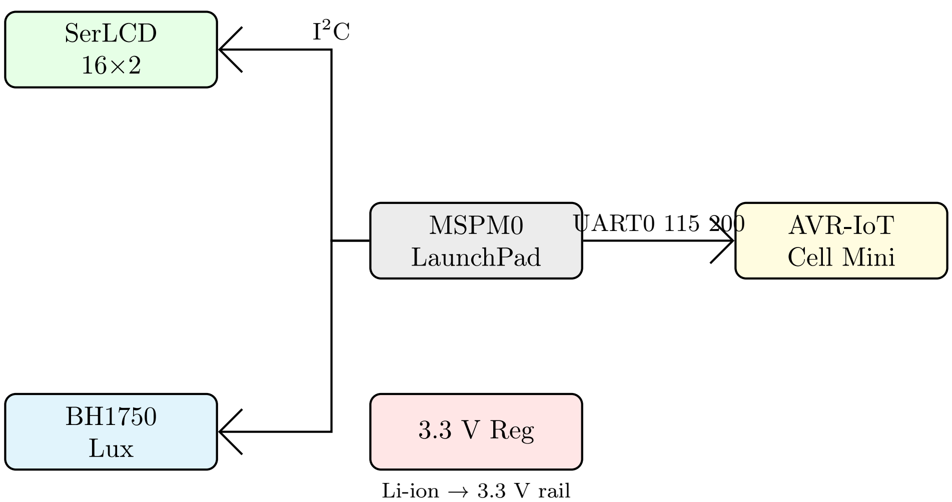

Figure 1: High-level block diagram. Solid=data

Pin map (LaunchPad)

- I^{2}C bus: `PA28` (SCL), `PA31` (SDA)
- UART0 to LTE: `PA10` (TX), `PA11` (RX)
- Optional RTS/CTS: `PA12`/`PA13` (left open in this build)

Mechanical notes. We initially designed a PCB so the parts could sit on a single board, but an issue in our design resulted in us using the MSPM0 Launchpad instead.

3 Pedagogical Notes on the Devices

SHT40 CMOS band-gap thermometer + capacitive RH sensor; communicates with one 2-byte command, great for teaching CRC and fixed-point conversion (not yet installed due to back-order).

BH1750 Spectral-split photodiode plus on-chip ADC; illustrates gain staging and logarithmic light levels. One I^2C word → lux after a single multiply.

SerLCD HD44780 LCD fronted by an AVR, exposing the legacy bus as I^2C 0×72—students can update text with a two-byte write.

AVR-IoT Mini Combines a Sequans GM02S modem with an AVR128DB48 that presents a single AT command port; convenient for demonstrating stateful serial protocols and MQTT.

MSPM0 LaunchPad 48 MHz ARM Cortex-M0+ with full driver library; chosen because CCS tooling is already familiar to ELEC 327 labs.

4 Firmware Overview

Key behaviour:

1. **System bring-up.** SYSCFG_DL_init() loads the TI-SysConfig pin-mux generated by CCS. An on-board LED is set high so field staff know power is good even before I^2C traffic starts.
2. **BH1750 initialisation.** The firmware writes the single byte 0x10 ("continuous high-resolution mode") to sensor address 0x23.
3. **First-measurement latency.** A coarse delay_ms(180) (180 ms) ensures the photodiode has completed its first 120 ms integration plus datasheet margin.
4. **Main loop—sample & heartbeat.**
   - A 2-byte I^2C read is started; the code spins in a fixed timeout loop waiting for each byte to appear in the RX FIFO.
   - If both bytes arrive before timeout, the LED toggles once each iteration—giving a visual ≈120 ms "heartbeat" that confirms the sensor link is alive.
   - The raw 16-bit luminance value is not yet converted to lux; that scaling step could be added once the humidity/temperature device is added in and the UART payload is formalised.
5. **Timing primitives.** Two delay helpers exist: delay_ms() is a calibrated busy-wait at 16 MHz; delay_cycles(1600000) is used only for the LED blink and equates to 100 ms at that clock rate.

At this stage the binary is just 3.4 kB. When the SHT40 arrives we will fold its driver into the same loop, replace the blocking delays with a one-shot timer interrupt, and format the lux/temperature/RH triplet into the CSV string sent to the cellular modem.

5 Results and Outlook

Though we were unable to design and integrate a complete PCB design, we were successful using the launchpad to send data from the light sensor through the AVR, as well as using the LCD to display. Because we did not receive the humidity and temperature sensor that we planned on using, we were not able to collect all of the data that we hoped to. We believe that we could spin this project off into a larger project if we were able to integrate all of our parts properly within the design.