The University of Georgia

ELEE 4230/6230E: Sensors and Transducers
Course Project 1 - Idea
Nikhil Mahadevan

Table 1 shows the tasks for this week. The total page limit (aside from if you choose to include this cover page) is **3 pages**. There are no hardware requirements this week. Make sure to look ahead, you'll need to order/obtain parts.

Table 1: *To submit to ELC.*

Weight	***Task***	***Page Limit***
20%	Describe your project idea. Be sure to include why you find it interesting.	½ page
20%	Describe why your project meets the constraints set forth.	½ page
10%	Include a bill of materials (BOM) of your parts list. Include part name, quantity, cost, a link to it, and any notes.	½ page
20%	In lecture there were 100+ sensors. Likely, in doing your research you found more. Explain how you got down to the final choices for your sensors.	½ page
20%	Think to the part about standards in lecture. What standards could apply to your project? Assume you are able to create a testing standard for your overall project goal (i.e., see the perfect cup of coffee example). What would the protocol focus on?	½ page
5%	Include what you learned this week.	1 paragraph
5%	Include how you interacted with AI this week.	1 paragraph

Table 2: Things you should seriously consider doing, but are not to be turned into ELC with CP1.

Weight	***Task***	***Comments***
0%	Write the problem definition of the report to be turned in with CP6.	The last week of class you have Exam 2 and CP6. But if you do CP6 with each checkpoint, life will be much easier.
0%	Start on the sensor selection to be turned in with CP6.	

2-Axis Light Tracker

This project is a solar panel mount that actively tracks the brightest light source in its environment and continuously repositions itself to maximize light exposure. Referring to Figure 1 below, two motors drive independent axes of motion: The turntable motor rotates the entire assembly around the vertical axis, and the panel motor adjusts the tilt angle of the panel. The structure consists of a turntable base with two vertical support pillars, with the solar panel suspended between them. One pillar houses the panel motor which adjusts the pane tilt. A potentiometer housed on the other pillar and also attached to the panel allows positional feedback of the panel. Light tracking works by using an array of photoresistors (Light Dependent Resistors, LDR's) mounted on the panel allowing the microcontroller to determine the direction of maximum light. Hall effect sensors paired with magnets act as limit indicators, allowing the microcontroller to stop panel movement and thus preventing over-rotation and wire tangling. A capacitive touch sensor enables the user to perform in-field calibration and self-test. This project builds on a simple analog light tracker from my Electronics I final project. Rebuilding it with a microcontroller, extra sensors, dual-axis mechanics, filtering, and closed-loop feedback is the next natural step.

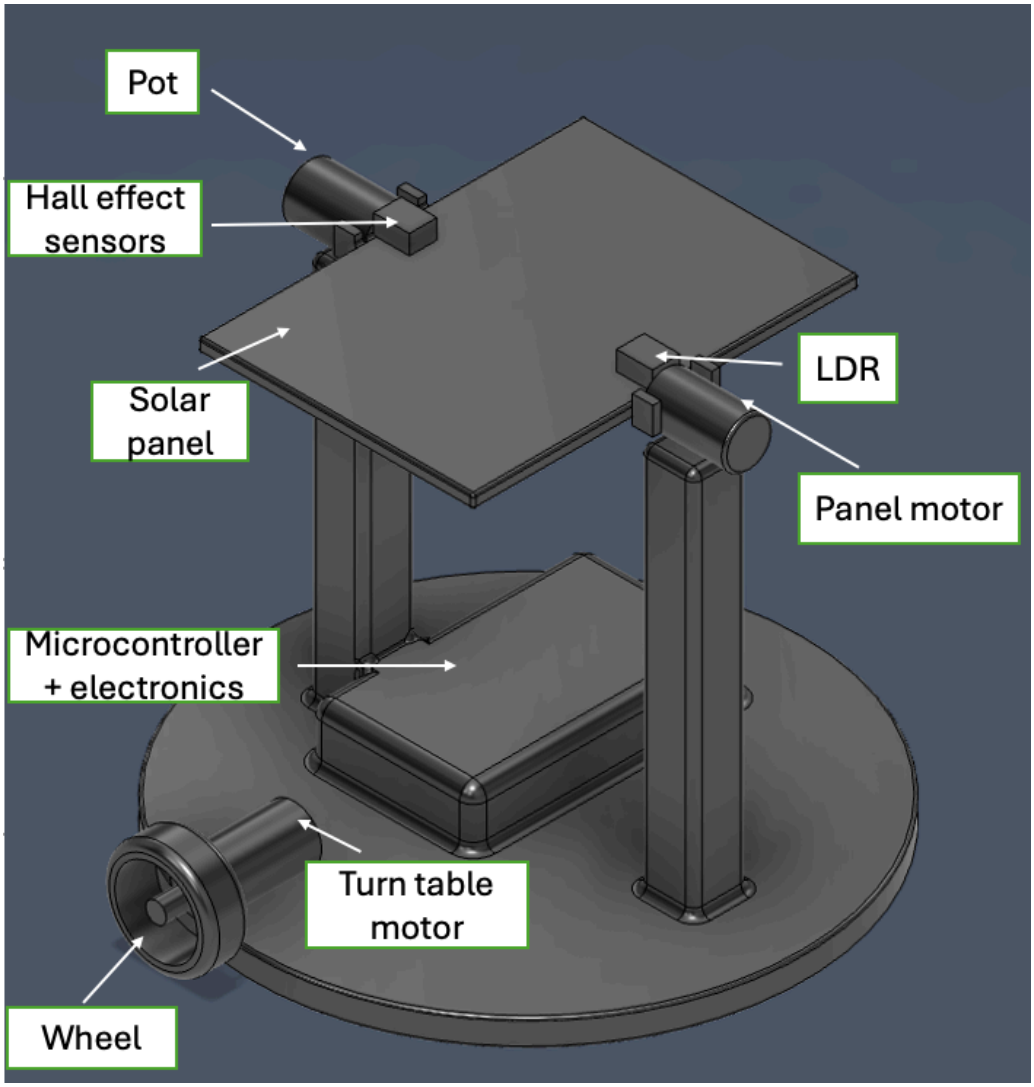


Figure 1. Isometric View of 2-Axis Light Tracker

Constraint Mapping

RAW Sensor: Photoresistors (LDRs) exhibit a nonlinear resistance response to light intensity - resistance increases at low light but becomes less resistive as brightness increases. Signal conditioning along with hardware and software filtering is applied.

Pre-Packaged Sensors: Hall effect sensors act as limit switches to prevent motor overrotation. A capacitive touch sensor enables the user to initiate calibration and self test procedures.

Battery Power: The entire project is battery powered without external or AC mains power.

Algorithms: Filtering in hardware and software, including Kalman filtering of the LDR signals is implemented in this project. The go-beyond algorithm could be implemented using a solar panel to recharge its own battery. Telemetry streams over UART from the microcontroller will be logged to a connected laptop.

Physics Redundancy & Measurands: Tilt angle is redundantly measured by two independent physical mechanisms - potentiometer (electromechanical) and hall effect sensors (magnetic) - satisfying the requirement for two branches of physics measuring the same quantity. The full measurand set includes illuminance via LDR (optical/raw), tilt angle via potentiometer (electromechanical), tilt limits via hall effect sensors (magnetic/pre-packaged), and touch input via capacitive sensor (capacitive/pre-packaged).

PCB: A custom PCB consolidates the MCU, signal conditioning, motor drivers, sensors, and UART output into a single self-contained board.

Name	Qty	Cost ($)	Link	Notes
Arduino Uno	1	27.60	Arduino Uno	Datasheet
LDR	1	5.75	LDR	30 pack Datasheet
Hall Effect	2	2.62	Hall Effect	Datasheet
Turntable	1	9.99	Turntable	Rotating base
DC Motor	1	15.99	DC Motor	2 pack
MOSFETs	1	8.99	NMOS/PMOS	12 pack Datasheet/10 pack Datasheet
Cap. Touch Sensor	1	8.49	Cap. Sensor	Datasheet
Battery	1	28.00	Battery	4 pack Datasheet
10 Turn Pot	1	11.99	10 Turn Pot	
Solar Panel	1	14.50	Solar Panel	
Shop items	NA	NA	NA	General electronic components, mechanical hardware, etc.

Table 3. Bill of Materials

Sensor Selection Rationale

Light Sensing (RAW): Cameras, photodiodes, and phototransistors add unnecessary complexity. Thermopiles measure infrared rather than visible light. LDRs were selected for their simple resistance-based output, nonlinear response, low cost, and suitability for array deployment.

Tilt Limits (Pre-Packaged): Mechanical limit switches wear over time and require accurate positioning. Optical interrupters add wiring complexity. Hall effect sensors were selected for their contactless operation, simple mounting, no moving parts, and direct digital MCU interface.

Tilt Angle (Redundancy): IMUs introduce drift and are power hungry. A potentiometer was selected for its simplicity and ease of interfacing, direct mechanical coupling, and energy efficiency.

Homemade Capacitive Sensor: A capacitive touch sensor was selected for its solid-state reliability and to satisfy the course requirement.

Standards Applied to the Project

While no single formal standard directly governs light tracking systems, this project incorporates metrics from IEEE 2700 (Standard for Sensor Performance Parameter Definitions) to characterize sensor sensitivity and a custom testing protocol was established across three critical dimensions: tracking accuracy, response latency, and fail-safe operation. This testing protocol ensures that the system is not only functional, but also consistent and robust enough for reliable operation. The primary goal for the dual-axis mechanism is to track a moving light source, ensuring the panel stays precisely oriented toward the brightest point in its environment. This necessitates calibration of the LDR arrays, verified by sweeping a handheld flashlight across the panel's range of motion confirming it follows without losing track. Additionally, the potentiometer ADC readings should stabilize when the light source is held stationary. Success is also defined by its responsiveness, requiring the system to detect dynamic changes in light positioning and initiate motor adjustments with negligible delay with no perceptible lag to an observer. Finally, the testing protocol confirms success by verifying that the device achieves mechanical reliability. It tests boundary conditions to prove that the software logic and limit switches can seamlessly halt motion at structural boundaries. More specifically, the hall effect sensors must stop the panel motor before any mechanical contact with the frame occurs. By simulating various movement extremes and sensor failures, the protocol validates that the hardware components work in tandem with the code to prevent damage to the components.

What I Learned This Week

This week, I started bare-metal C programming for the ATmega328P microcontroller (Datasheet) used in the Arduino Uno. I had no prior C experience and learned how to set up software tools, manipulate registers, and perform bitmasking particularly regarding the configuration of the integrated ADC and I/O ports. I learned how to implement a prescaler to divide the clock signal down to a viable sampling rate and how to perform bit manipulation in I/O ports used for motor control.

AI Interactions

I utilized AI to learn how to set up a Visual Studio Code environment to program the ATmega328P microcontroller. I also used AI to help me navigate the learning curve of the C

programming language and strengthen my conceptual understanding rather than generating copy-paste code. My interactions were primarily about bitmasking techniques and understanding execution procedures, such as the multi-register sequence required simply to initialize the ADC. By explicitly restricting the AI to structural explanations, I ensured that I wrote every line of code myself, which led to a deeper understanding of interfacing the microcontroller to hardware.