ECEN 345/ECEN 3450-001

Mobile Robotics I

Prof. A. Gilmore (Instructor)

Sajjad Alhassan (TA)

Laboratory Assignment 4

Lab 4 - Light Sensing with Photo-resistors

Written by

David Perez

Department of Electrical and

Computer Engineering (ECE)

University of Nebraska

Lincoln/Omaha

Due: 3/10/2024

Introduction

The objective of lab four was to enable the CEENBoT robot the ability to home in on a target. The target in the context of this lab was a light source or lack thereof. The sensor used to provide light intensity measurements was a photo-resistor. This sensor was soldered together and connected to the robots ADC's read the voltage across the photo-resistor. To accomplish this task the ceenbot-api documentation, and lab handout proved to be helpful resources to configure the robot. This lab took about 7 hours to complete and was completed by David Perez.

Background

To start off the lab, the first thing that needed to be implemented was the photo-resistor circuit for the robot. The photo-resistor is a semiconductor that changes its resistance based on light intensity. These attributes can be configured in a voltage divider circuit where an output voltage will be a function of the variable resistance of the photo-resitor as seen in **Figure 1** below.

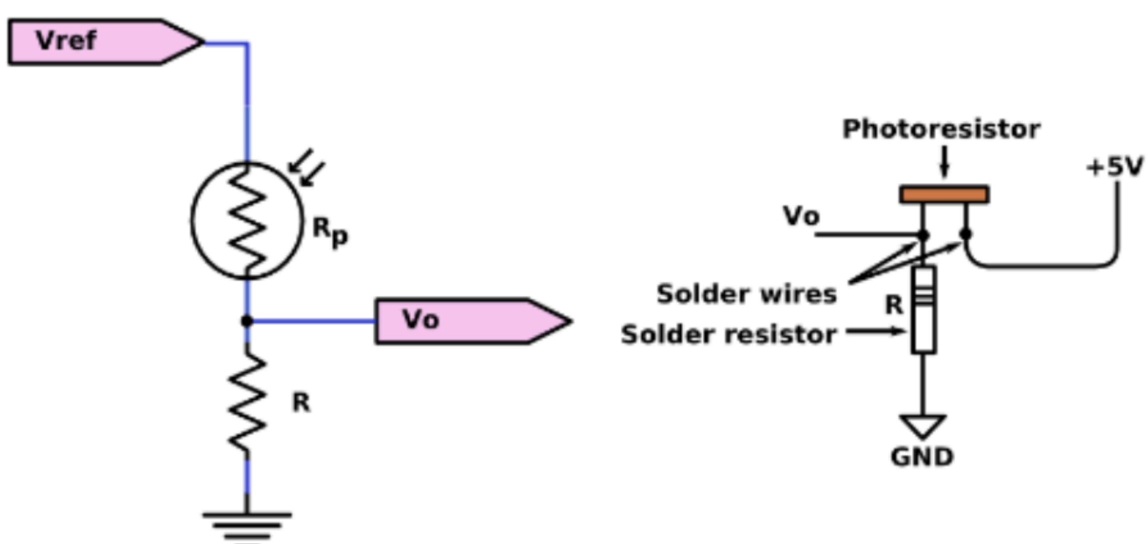

Figure 1: Photo-Resistor Voltage Divider Circuit

Obtaining the 5v and ground supplied by the CEEBoT robot the circuit can be constructed and attached to the robot's ADC's (see datasheet for unreserved ADC's). Since there will be a photo-resistor sensor on the left and right front of the robot this will require the same assembly of two circuits.

Observing the circuit in **Figure 1**, the output voltage of the provided circuit is given by:

$$V_o = \left(\frac{R}{R + R_p} \right) \cdot V_{ref}$$

Equation 1: Photo-resistor circuit output voltage

The photo-resistor sensor is manufactured by Jameco with part number CDS001-8001. This documentation states the resistance of can fluctuate between 3kΩ to 200kΩ. The additional resistor provides additional tuning to the voltage reading. A good choice is between 3-10kΩ. Doing so gives the upper and lower bounds of the output voltage (**Equation 2**). Then **Equation 3** results when the +5v reference voltage is used.

$$0.0476 \cdot V_{ref} \;\leq\; V_o \;\leq\; 0.7692 \cdot V_{ref}$$

Equation 2: Output Voltage bounds

$$0.238\text{V} \;\leq\; V_o \;\leq\; 3.846\text{V}$$

Equation 3: Output Voltage bounds with +5v Vref

This output voltage was sampled as instructed by the documentation in the lab handout and ceenbot-api documentation. Keep in mind that the configuration of the ADC is outlined, a conversion is also required to convert the sample into a voltage reading using **Equation 4.** This is required due to the fact that the ADC produces a 10 bit value stored in a 16 bit number which then needs to be converted to a voltage.

$$V_{\mathrm{ADC}} = \left(\frac{\mathrm{ADC}_{\mathrm{code}}}{1024} \right) \cdot V_{\mathrm{ref}}$$

Equation 4: ADC Sample Conversion

Using this hardware the robot now has a data stream of the light intensity. To achieve a homing effect these sensor values were used to determine how much speed should be applied to the robots motors. Using different configurations of right and left motor and right and left sensor determines if light homing or avoidance behavior occurs.

Procedure

As mentioned the first step was creating two photo-resitor circuits and soldering wires to connect them to the robot. Testing was done with each sensor to get sensor resistance and voltage reading testing was done to calibrate the sensors (see *Results* section). Additionally, the sensors were mounted facing forward on the left and right front side of the robot. The reference voltages were placed on ADC 3 and 4 (which is pin 1 and 2 on the J3 header).

With the assembly and confirmation of the sensor complete, the second part of the lab was creating the homing behavior for the robot. The behavior that was mimicked was inspired by Valentino Braitenberg’s vehicles. The first program used the inspired reactive control architecture. This architecture directly maps the voltage reading to some motor input voltage (along with some scaling). An excitatory and an inhibitive behavior were implemented based on light intensity. Meaning the robot would speed up or slow down given light intensity readings.

Source Code Discussion

Before any behaviors were implemented a scaling of the sensor readings was done. That is, the right sensor would generally have .250 mv reading higher than the left side so this was reduced from the right sensor reading first. Then, the 4 behaviors implemented for this lab are encompassed by 4 different scenarios. The first being the same side inhibitory (*SS_INHIBITORY* in source code) which inversely maps the left sensor voltage directly to the left motor velocity and right sensor to the right motor. The left sensor voltage inversely mapped to the right motor is another state referred to “opposite side”.This inhibitory behavior is a result of the max sensor voltage being subtracted by the sensor voltage reading before the multiplicative constant, then being fed into the motor controller. The ADC sample conversion of the sensor readings was done by the *read_right_pr_sensor_voltage()* and *read_left_pr_sensor_voltage()* functions. The display of these readings was done by the *display_pr_readings()* function.

The behaviors mentioned above are implemented in the *app_main()* function in **app.c** of the source code and dependent on the *ReactiveBehavior* enum input. This function is what holds the main driving loop for the program. This function first reads the photo-resistor’s, displays them, then scales the right sensor voltage reading. A switch statement then determines which mapping of sensor to motors occurs and also if the motor should speed up or slow down to higher lighter intensities. These switch blocks call the function *set_motor_speeds()* to set both motor velocities and is the only function used by the **motor_control.c** file. The remainder of the source code is the main which configures the LCD, STEPPER, and ADC subsystems.

Results

The first part of the lab was to measure the min and max of the left and right sensor. The corresponding values mins were 277Ω and 599Ω with their maxes being 158kΩ and 161kΩ respectively. Using **Equation 1,2, and 3**, their ranges are shown in **Table 1**:

Sensor Location	Theoretical Min Output Voltage	Theoretical Max Output Voltage
Left	0.066	3.911
Right	0.067	4.525

Table 1: Output Voltage Calculations

Part 4 of the lab handout consists of testing the light sensor readings in different light intensity environments. The sensor voltage for each condition for each sensor was marking and is shown in **Table 2** below:

Light Conditions	Left Sensor Output Voltage (V)	Right Sensor Output Voltage (V)
Lab Table	1.079	1.074
Lab Floor	0.703	0.659
Dark (covered sensor)	0.132	0.034
Bright artificial Light	2.766	2.556
Sunlight	3.331	3.44

Table 2: Photo-resistor Reading’s in Varying Light Conditions

Part 5 of the handout was testing of the “same side” excitatory behavior. THe robot performed as expected and would approach a light source at a speed proportional to the light intensity. Next was the “opposite side” excitatory where we would expect the robot's left motor to turn forward proportional to the light intensity of the right sensor. Again, the robot exhibited this behavior which allowed the robot to get closer to the light source since it would swing its entire body to face directly at the light.

The inhibitory behavior was to decrease the motor speed in proportion to the light intensity. It is therefore expected that the robot would slow down with a greater light source intensity. The robot successfully decreased its speed with a bright enough light source pointed at it. The robot's turns mimicked that of the excitatory behavior with the exception that it would slow down with greater light intensities.

Conclusion

The purpose of this lab was to investigate how photo-resistors could provide a voltage reading of light intensity for robot design. This sensor paired with a reactive control code architecture allowed the robot to home in or slow down based on light intensity. This was looked at analytically then implemented directly on the CEENBoT robot. The implementation of this functionality was successful and the robot was able to steer towards and also avoid varying light intensities.