Understanding the Light Sensor Through Data Logging

Kathryn della Porta
Basic Engineering I
November 1, 2010
The LEGO Light Sensor

• The LEGO light sensor will respond to both ambient light and reflected light from the sensor’s built-in red LED.

• The light sensor will respond to radiation in the visible spectrum as well as the infrared spectrum.

• The light sensor’s default mode is percent mode.
Percent Mode vs. Raw Mode

- A high raw value corresponds to a low percentage value.
- In percent mode:
  - Values range between 0 and 100
  - Low Value = Less Light
  - High Value = More Light
- In raw mode:
  - Values range between 0 and 1023
  - Low Value = More Light
  - High Value = Less Light
The LEGO Light Sensor vs. The Human Eye

• The human eye has two structures, called photoreceptors, that sense light: rods and cones.
  – Rods are more sensitive to light and are responsible for night vision and peripheral vision.
  – Cones are less sensitive to light and detect color, fine detail, and rapid changes in images.
• The human eye has millions of rods and cones.
• The LEGO light sensor has the equivalent of a single rod.
What We See
What the Light Sensor “Sees”
The Myth of Sharp Edges

• What, to the human eye, appears to be a “sharp” or well-defined edge becomes irregular as it is magnified.

• When the LEGO light sensor is pointed downwards at a black line, it “sees” only a very small section of the edge of the line.

• The light sensor senses the average brightness over this small section of the edge of the line.
Why is This Useful?

- Knowing how the light sensor “sees” the edge of the black line allows us to write better line-following algorithms.
- Knowing how the robot senses its environment allows us to understand and more effectively debug unexpected robot behaviors.
Data Logging

• The datalog allows a program to record data, such as a sensor’s output, that can be uploaded to a computer for analysis.
• The datalog shares memory space with the programs uploaded to the RCX, and requires 3 bytes of memory per entry.
• The maximum size of the datalog is 2033 entries, but this would leave no room for a program to collect the data.
Using the Datalog

• Step 1: Initialize the datalog
  – `CreateDatalog(# of entries);`
  – Creating a datalog with 0 entries will eliminate the datalog.

• Step 2: Fill the datalog
  – `AddToDatalog(target);`
  – To add data to a datalog at regular intervals, use a for loop.
Converting to Excel
Experimental Setup
Datalogging Program

```c
#define MOTOR OUT_B
#define EYE SENSOR_1
#define DATA 500

task main()
{
    // SetSensor(EYE, SENSOR_LIGHT);
    SetSensorType(EYE, SENSOR_TYPE_LIGHT);
    SetSensorMode(EYE, SENSOR_MODE_RAW);

    CreateDatalog(0);   // clear datalog
    CreateDatalog(DATA);

    OnFwd(MOTOR);

    int i;
    for(i=0; i<DATA; i++)
    {
        AddToDatalog(EYE);
    }

    Off(MOTOR);
}
```
Data Collected

Light Sensor Values Crossing Black Line in Percent Mode

Light Sensor Values Crossing Black Line in Raw Mode
Interpreting the Data

Light Sensor Values Crossing Black Line in Percent Mode

- Sensor value over white mat
- Sensor value over black line
- Crossing the edge of the black line

Notice the ‘slope’ of the data as the light sensor crosses the edge of the black line. The steep slope indicates that the transition between dark and light happens very quickly, in a narrow area. This data will inform how hysteresis is implemented in a line-following program.
Logging a T-Track Navigation Program

Light Sensor Values in Percent Mode Following Black Line During T-Track-Following-Program Execution

- Robot follows black line
- Robot turns around after detecting obstacle
- Robot misses green square and turns around
- User picks up robot
- User holds robot over green square
- Robot follows black line around corner, turning right