ENGR290

Winter 2023

Technical assignment #1

Objectives:

- Get familiar with Arduino Integrated Development Environment (IDE).
- Gain hands-on experience in basic microcontroller's features:
    - General Purpose Input-Output (GPIO);
    - Timers and Pulse-Width Modulation (PWM);
    - Analog to Digital Converter (ADC);
    - Serial communication: Universal Asynchronous Receiver-Transmitter (UART).
- Gain hands-on experience with Infrared (IR) and Ultrasonic (US) rangefinders and learn about their features.
- Introduction to datasheets. (Hint: watch the recording of my "datasheets" lecture available on Moodle.)

Tasks to perform:

- Find and download the datasheets for both sensors and ATMega328P.
- Connect to the controller:
    - US rangefinder (HC-SR04);
    - IR analog rangefinder (GP2Y0A21 or GP2Y0A02. Make your choice based on the sensors' specifications and the assignment requirements).
- Get readings from them.
- Control D3's brightness depending on the distance between the each (one at the time) sensor and an obstacle:
    - 15 cm or less – 100%;
    - 45 cm or more – 0%;
    - linearly increases from 45 cm to 15 cm.
- Turn yellow LED ("L", connected to PB5) on the Arduino board on when the obstacle is outside of [15; 45] cm range.
- Output the actual distance and ADC readings on the screen via UART (You can use Hyperterminal or any other terminal software of your choice, e.g PuTTY or Arduino serial watcher).
- Once you finish writing the code that performs the aforementioned tasks, continue to the experiment.

Experiments' description:

1. Choose the **<u>optimal</u>** ADC reference voltage <u>for your project scope</u> for the IR sensor and set it with RV1. A voltmeter (DMM) is available in H-941.
2. Connect IR sensor to the board to any ADC channel (3-pin white header) of your choice.
3. Run three trials and fill out the table (Table 1) for the report:
    a. move an obstacle farther that 45 cm;
    b. move it to 45 cm mark and record the ADC reading;
    c. move it to 15 cm mark and record the ADC reading.
4. Move the obstacle from 15 cm distance **<u>towards</u>** the sensor till the obstacle touches the sensor. Observe the sensor's output and LED L's behaviour.
5. Lift the sensor ~5 cm above the table. Point it ~30 degrees up and away from any obstacle within its range. Slowly tilt it down to ~45 degrees and observe how the sensor's output changes.
6. Lift the sensor ~10 cm above the table, point it horizontally away from any obstacle within its range. Keeping it pointed horizontally, slowly bring it down until it touches the table and observe how the sensor's output changes.

7. Connect the US sensor to a digital port (4-pin white header), change the code accordingly and repeat steps 3-6.

Report:

- **Fill out table 1:**

Table 1 is available in a separate doc-file for your convenience. Get it from the Moodle and copy/paste it in your reports. Make sure you preserve the formatting.
"Average": the average values for the actual distance and ADC readings (IR) or pulse length (US) for three trials.
"Standard deviation" – the same as above.
"Expected reading (ADC value (IR) or pulse length (US))" – the ADC value(IR) or pulse length (US) you expect to read according to the data available in the sensor's and microcontroller's datasheets. **Hint:** pay attention to the ADC Vref.
"Error, %" – ("Expected"-"Average")/"Expected"*100%.

- **Answer the questions. All answers should be justified, when necessary:**

1) Explain how you calculated the Vref.
2) Did both sensors have the same performance **<u>for the distance range in this assignment</u>**?
3) How did each sensor behave when the obstacle was in its close proximity? Did you see something strange? Explain what went wrong and why.
4) Where is in the datasheet the sensor's output vs. distance values can be found (for each sensor)? (Give the datasheet you used revision number, page and paragraph.)
5) How did you convert ADC readings (IR) or pulse length (US) to distance?
6) Given that you do not have to indicate or use actual distance expressed in cm in any way, how you would write your code that controls the LEDs **<u>without</u>** converting the ADC readings into distance (in cm) in your code? (**Rationale:** the distance in cm uses floating point variables, which are not "native" for the microcontroller used for this assignment and the project. It results in huge code inflation and decrease of overall system's performance. Keep this in mind for your final project code.)
7) Did you observe something strange in the LED L's behavior when it is connected to one sensor, but not to another? If so, explain, what happened.
8) Compare your observations in steps 5&6 for the sensors. Which parameter of the sensors you tested?
9) Write a few sentences (no limit) about what you have learned in this assignment, what surprised, shocked and/or enlightened you, what difficulties you experienced, what kind of help you could have profited from, etc.

*Keep your answers short - **three sentences max** per answer. I will NOT read and grade anything that exceeds either three sentences or five lines of text per answer (Arial 10 font, regular spacing, no undeclared abbreviations, no "shortening" of the words). It is not cumulative, e.g. you cannot use two sentences for the answer #1 and four sentences for the answer #2.*
***<u>CRUCIAL</u>:** A copy of the **expectations of originality form** signed **<u>in ink by hand</u>** by all team members, who **PARTICIPATED** in the assignment and **contributed** to the report, must be attached to the report.*
<u>Reports without properly filled in expectations of originality form will not be graded.</u>