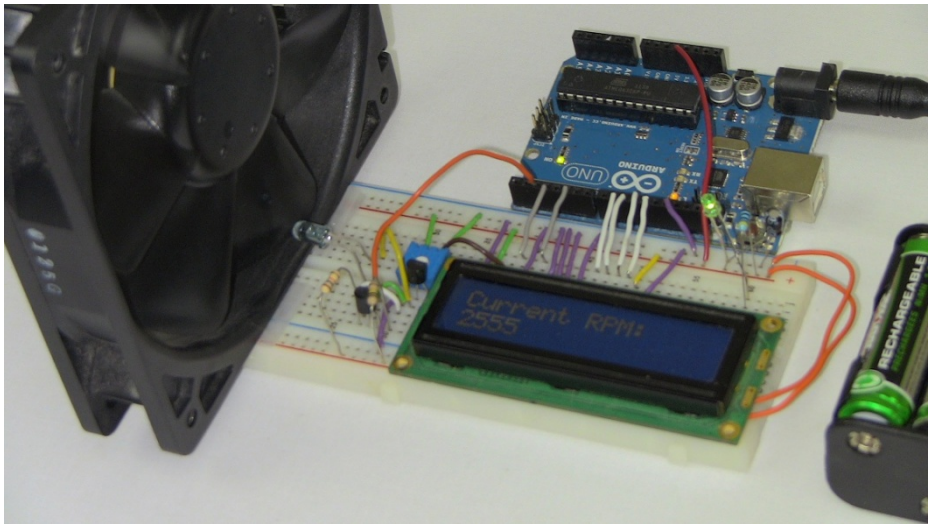
CENG314 EMBEDDED COMPUTER SYSTEMS

LABORATORY WORK 2

INTRODUCTION

A tachometer is a useful tool for counting the RPM (rotations per minute) of a wheel or basically anything that spins. The easiest way to build a tachometer is using a transmitter and receiver. When the link between them is broken, you know that something is spinning and can execute some code that calculates the current RPM of whatever is spinning to break the transmitter/receiver link.



In this lab session, we will explore how to use an IR transmitter and receiver break-beam for a tachometer application. Arduino will be used for all the processing and break-beam interruption counting. The end result will be a 16x2 LCD displaying the RPM of a computer fan.

MATERIALS

1x Arduino board
1x Breadboard
1x HD44780 compatible 2x16 character LCD
1x infrared led (transmitter)
1x infrared phototransistor (receiver/detector)
1x 100k ohm resistor
1x 330 ohm resistor
1x 100k ohm potentiometer
1x 10k ohm potentiometer
1x 12V computer cooling fan (80mm)

PRE-LAB

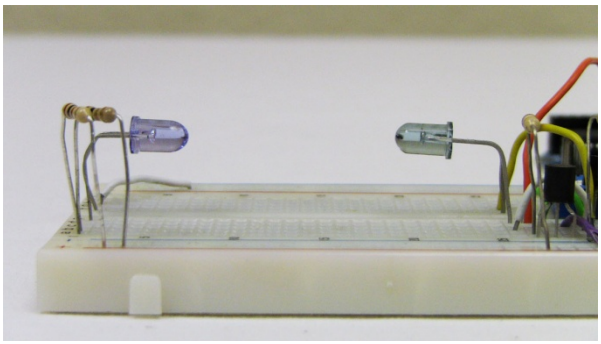
Please bring your ideas or solutions on the matters below in the beginning on the lab session in written form.

1. Describe your approach for translating input signal into numerical RPM values.
2. Calculate the input signal frequency that corresponds to exactly 1000RPM of a typical 7-blade fan.

LABORATORY WORK

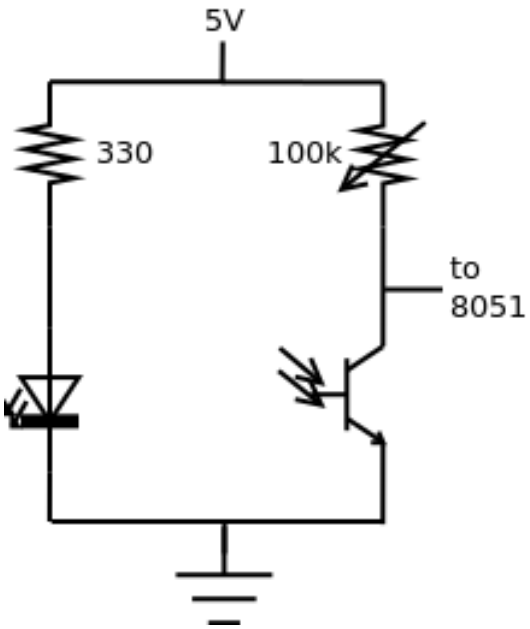
The purpose of this project is to build a single input, single output system. The input will come in the form of a signal state change from high (+5v) to low (+0v) which will occur when the IR break-beam is interrupted and the Arduino will then increment an internal counter. As time goes on, additional processing and calculation will occur as interrupts are trigger and the LCD will output the calculated RPM.

To create the IR break-beam, we will used an IR LED with a low value resistor so that it shines very bright. The receiver will be a phototransistor which biases 'on' whenever the IR LED's light is detected. A computer fan will be placed between the IR link and turned on so as to continuously generate an interrupt through some additional transistor logic circuitry. For output, the Arduino LCD interface that we saw last lab session will be used so that we can output the final RPM value to the LCD.



Since we're going to be counting the RPM for a CPU fan, first we need to realize that we're using an IR break-beam that counts every interruption. This is great, except we do need to realize that the CPU fan has 7 blades. This means 7 interrupts is 1 revolution.

If we keep track of the interrupt count, we can know that every 7th interruption means 1 full rotation has just occurred. Similarly, if we keep track of the time it takes for every full rotation to occur, we can then easily calculate the full RPM.



Follow the steps below to complete your goal for this lab session.

1. Setup your circuit. Make all connections carefully. Pay special attention to the polarity of the components.
2. Place the fan in between the IR transmitter and receiver. Make sure that fan blades interrupts the line of sight between the transmitter and receiver. Verify the output of the sensor using an oscilloscope if necessary.
3. Write an Arduino sketch that accepts the input signal from one of the interrupt pins. This program should count the pulses and translate the sensor signal into a numerical RPM value. Display the RPM value on the LCD.
4. Gently touch the center of the fan to slow its rotation speed. Observe that indicated RPM value changes.
5. Demonstrate your sensor output signal, Arduino sketch, and running circuit to the lab assistant.