3 Electronic Sensors and Signal Processing Module

Module Due: 15FEB2018

Pre-lab Due: 06FEB2018
(Sections 3.3.1.a. 3.3.2.a, 3.3.3.a)

3.1 Overview

The primary output of this module is a completed and functional prototype of the optical sensing subsystem that meets the requirements for the Pinball Machine Project.

In addition to design, fabrication and characterization of the optical detector and scoring feedback subsystems for the pinball machine, this module has other activities (see section 0) that build relevant skills both inside and outside of the lab sessions. These other activities include introductions to signal to noise ratio (SNR), phototransistors, operational amplifier circuits, and frequency domain filtering.

Save your circuits and optimize your breadboard space! You can keep most of your electronics that you design here for your final project, which will be **run completely from your breadboard.**

Prelab Activities: 3.3.1.a.1-2, 3.3.2.a, 3.3.3.a.1-3, 3.3.3.c.1,

3.1.1 Learning goals

Purposes ('What this module's activities seek to develop')

- Prepare students/teams for sensor development and integration into electronics design projects
    - Understanding figures of merit to characterize sensor/input subsystems
    - Understanding of frequency domain signal representations
    - Filter design skills and experience
    - Knowledge of basic signal processing methods and analog circuits

Knowledge/Functional goals ("Students/teams will (be able to)...")

- Design and build LED driver circuits
- Design, build, and characterize a phototransistor receiver
- Design, build, and characterize a first-order RC filter
- Analyze, design, build, and characterize different amplifier circuits using op-amps and discrete components
- Read and display sensor data with an Arduino
- Identify potential challenges/questions about an electronics design and test them through prototyping and experiments
- Document and evaluate electronic test results appropriately
- Create plans for next-step circuit prototype(s) based on existing prototypes and experiments

3.2 Deliverables

Please create a **single .doc(x) file** with all of the deliverables for each section into that document and upload it to TritonED. If an additional file is needed, upload it to TritonED (appropriately named) and take a screenshot to document the work in your single .doc(x) file. For each of the sections, what you need to include is described in green text, and the separate "**Module Deliverables Instructions**" document on Coursework includes further instructions on deliverables.

3.3 Activities

To augment the lectures, the following textbook sections are directly relevant to this material and to understanding the basics of electronics more generally.

Text: *Practical Electronics for Inventors*, Scherz

<u>Core Reading</u>

- 2.20-2.20.7: AC signals; 2.21: AC and RMS
- 3.6-3.6.5; 3.6.9-11: Capacitors (practical/usage)
- 5.7-5.7.5: Phototransistors
- 8-8.4: Op-amps (basics and topologies); (7-7.4 in Ed. 2)
- 9-9.4: Filters (basics); (8-8.4 in Ed. 2)
- 12.3-12.3.3 Combinational Devices; 12.7.1 Counter ICs; 12.8-12.8.2 Shift Registers

<u>Optional Reading</u>

- 2.23, 2.23.5-8, 2.23.10-14: Capacitors (theory);
- 2.24.10-12: energizing an LR circuit
- 3.7-3.7.4: Inductors
- 8.6-8.9: Op-amps (practical tips); (7.6-7.9 in Ed. 2)
- 12.11.1 LED Displays

3.3.1 Signal-to-Noise Ratio and Dynamic Range in Sensor Circuits

For any sensor system, two key figures of merit are signal-to-noise ratio (SNR) and dynamic range.

'Signal' is defined as the quantity of interest for the sensor. For example, if you know that your source is a 1kHz sinusoid, the signal will be a 1kHz sinusoid of some amplitude.

'Noise,' is defined as everything else that the sensor/system includes in the measurement that is not signal. Basically, any background that occurs when the signal is not present, or what would be left if we subtracted the signal from what we are measuring. Note: the term 'noise' has various meanings in different contexts, and in many definitions it must be a random process as distinct from interference (which is not random), but we will not worry about this distinction in this module.

SNR is a unit-less quantity that is the ratio of the signal level (typically in units of voltage or power, depending on the application) divided by the noise level (in matched units). In basic terms, the SNR tells you how well you are measuring what you want to measure—as opposed to noise or interference that confounds your measurement.

Dynamic Range is the range spanning the minimum input signal and the maximum input signal that your system can handle. In basic terms, dynamic range tells you how much variability you can have on the input signal and still get a useful measurement.

The following skills-building activities introduce fundamental quantities of signal, noise, signal-to-noise ratio (SNR) and dynamic range in evaluating sensor circuits.

3.3.1.a Signal-to-noise ratio and Dynamic Range Concepts

3.3.1.a.1 Calculating SNR for example real-world signals: EKG examples (pre-lab activity)

For the following 2 plots, estimate the noise amplitude [in Volts], the signal amplitude [in Volts], and then calculate the signal-to-noise ratio. Assume, for each, that you are using an Arduino analog input channel (maximum input level: 5V).

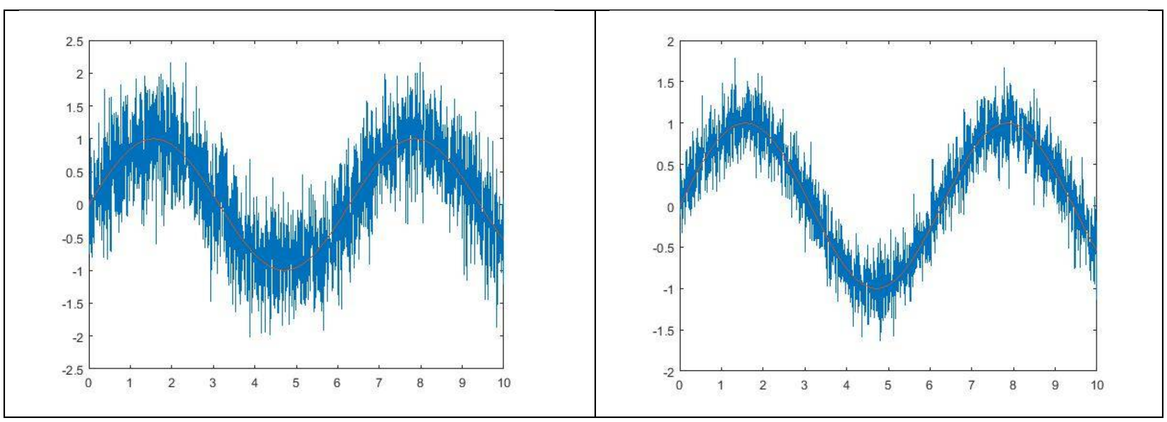
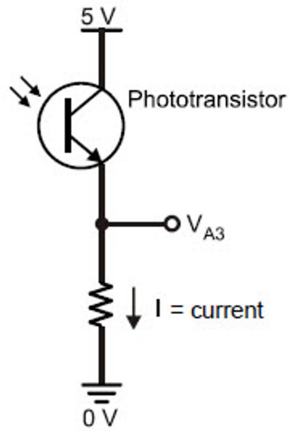


Figure 1a and 1b: Sinusoidal signal with noise for SNR estimation.

To see the concepts of signal, noise, and SNR in practice, we will design and build a basic optical transmitter/receiver system and measure its performance as a function of different system variables. It should transmit a 500Hz square wave (generated by a function generator) across an air gap of at least 2cm and receive it and display it (on an oscilloscope).



An optical transmitter/receiver comprises a light source (e.g. LED) and a light detector (phototransistor), which acts like a resistor: the resistance decreases with more light and increases with less light.

NOTE 1: We have phototransistors that measure visible and infrared light, as well as ones that has an infrared filter so it does not measure visible light. Both are useful, but because infrared phototransistors do not pick up ambient light and only IR spectrum and above, they are a more robust option.

NOTE 2: Debugging is a challenge if you can't tell if an IR LED is on or not. However, most face cameras on our phones do not have an IR filter so you can use these cameras to see if an IR LED is on or not.

3.3.1.a.2 **Plan**: Create a circuit diagram for the simplest possible optical transmitter/receiver. (pre-lab activity)

- Create a circuit diagrams for the simplest possible optical transmitter/receiver. You will have two circuits: one to power your led, the other to power your optical sensor.
- Remember to appropriately **limit the LED current**

3.3.1.a.3 **Plan**: Create a test for demonstrating how well your transmitter/receiver works under various conditions you would be interested in (e.g. ball rolling past, in sunlight, etc.)

- What outputs will you measure/report on the transmitter and on the receiver circuits?
- What variables will you test for and what are their effects on the received signal? (consider various environmental/setup factors)

3.3.1.b Signal-to-noise ratio and Dynamic Range in Practice (lab activity)

3.3.1.b.1 **Build** a transmitter and receiver circuit

- Build the circuits you designed in 3.3.1.a.2

3.3.1.b.2 **Test** the circuit/system according to your test plan from 3.3.1.a.3

- Plot the results and/or assess
    - o Did your tests work out?
    - o Do you need to redesign or rebuild and repeat to get a good test result, or is the circuit about as good as it can be without changing the design in a fundamental way?

You probably pointed the LED and photodiode directly at each other, and broke contact to see the change in signal. Turn the LED and photodiode so that they are now side by side and facing one direction. Move a piece of paper back and forth towards the pair.

- When do you see a signal change, and therefore how close does an object have to pass by before you pick it up in this configuration?

3.3.2 Analog Filters

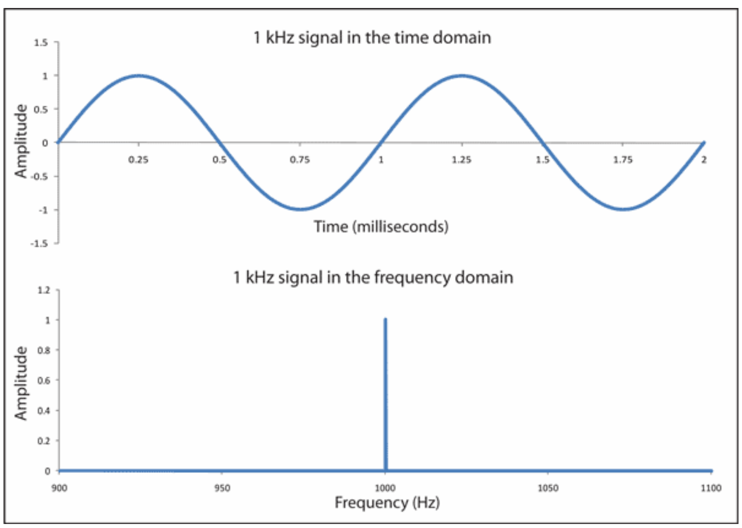
Filters perform the important task in many electronic circuits of distinguishing between signals and interference/noise that exist in a circuit in different frequency ranges. For example, 60Hz and 120Hz interference is all around us due to the electrical circuits in most buildings, and they get into sensitive electronic measurements because any conducting wire acts like an antenna.

A great trick is to **provide a signal of a known frequency,** and then using analog filters, **leave only those frequencies of interest** to the system. This is a powerful way to improve signal-to-noise and dynamic range, which allows your system to work in many different environments and under different conditions.

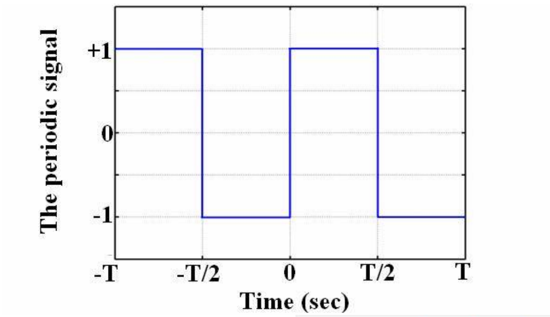
In this section, you will analyze, design, build, and characterize simple RC filters. Note: It may seem tedious, but with a simple (e.g., Excel) data sheet and practice, characterizing a filter takes <5 minutes.

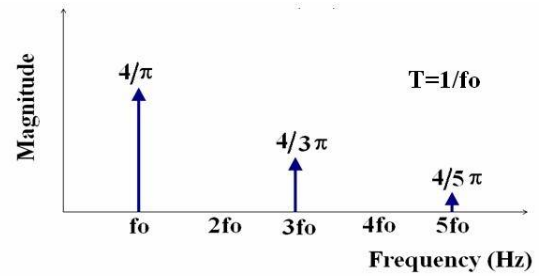
To understand and analyze the behavior of the filters we should first understand the difference between **time and frequency domains signal representation**. The general idea is based on Fourier theorem that states that any signal can be represented as a sum of sinusoidal signals (of different frequencies and amplitudes). The plot that shows all the sinusoidal components of the signal is known as a frequency domain plot or spectrogram of the signal. Let's take a look at the example:

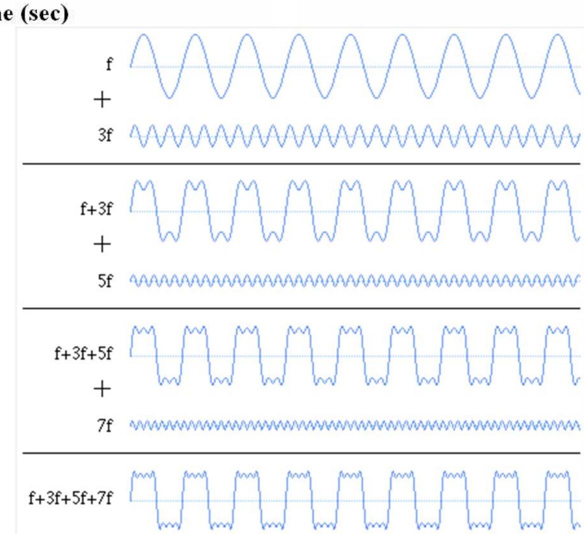
A sine wave and its frequency domain representation:



A square wave is comprised of the infinite sum of sinusoids:

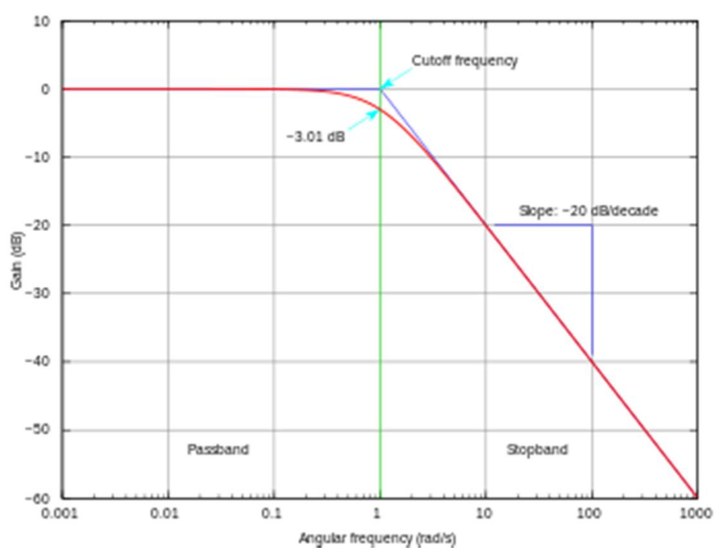






As expected you can observe square wave sinusoidal components on the spectrogram. Note here that the majority of the signal is accounted for at frequency **fo**, which is exactly the frequency of the square wave**.**

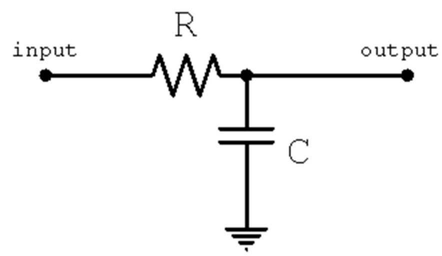
If we want to extract only the base frequency of the square wave above (sine component at f_0) then we should consider using a filter. Here is an example of frequency domain magnitude behavior of a **1st order low pass filter:**



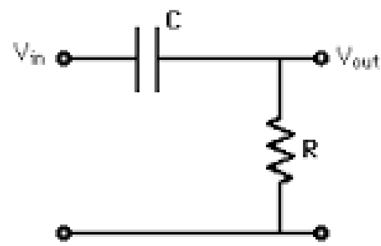
**note the Gain (y-axis) is in dB and not absolute amplitude*

The cutoff-frequency shows where the signal power drops (or *attenuates*) by 3dB, which corresponds to $\frac{1}{2}$ or 50% in signal power and $\frac{1}{\sqrt{2}}$ or 70.1% drop in signal voltage (recall that when we deal with power of the signal dB = 10log(P/P0) and with voltage dB = 20log(V/V0)), and is effectively where the amplitude of the filtered signal will start dropping off considerably. As you can see, at higher frequencies the filter reduces their amplitude significantly, thus making only ***low*** signals ***pass*** through. Alternatively, a high pass filter attenuates low frequencies and passes high frequencies.

A **first order low pass** filter design is shown below:



A **first order high pass** filter design is shown below, where



In both cases, **1/RC rad/s is your cutoff frequency.**

A quick note on capacitors: Some capacitors are unidirectional! They are called *electrolytic* capacitors (looks like cylinders). Since we are dealing with oscillating signals, they are not suitable and can in fact blow up. You should be using *ceramic* capacitors (looks like discs), which allow current the flow both ways.

3.3.2.a Analysis and Design of Simple Filters (pre-lab activity)

- Design a low-pass RC circuit with a cutoff frequency of approximately 500Hz
- Design a high-pass RC circuit with cutoff frequency of 500Hz

3.3.2.b Analysis and Design of Simple Filters (lab activity)

- Build analog filters using resistors and capacitors
    - Build your low-pass RC circuit with a cutoff frequency of approximately 500Hz
    - Build your high-pass RC circuit with cutoff frequency of 500Hz
- Generate on a function generator a square 1kHz input signal, 0-5V peak to peak
    - For both circuits, plot on the oscilloscope the input signal and the filtered signal simultaneously. Save/Sketch this behavior in your notebook
    - Which filter(s) preserve a DC offset on the input and why?
- For your optical transmission/detection circuit from 3.3.1, provide a driving PWM of 1kHz, and choose a filter that will filter out electrical noise, such as 60Hz/120Hz, but still captures the driving signal you give it. On the oscilloscope show both the signal measured from the phototransistor before filtering and after filtering simultaneously. Save/Sketch this behavior in your notebook. What is the peak-to-peak amplitude before and after filtering?

3.3.3 Operational Amplifiers

This activity introduces amplifiers, and specifically operational amplifiers, as useful circuit blocks for building electronics and electromechanical systems.

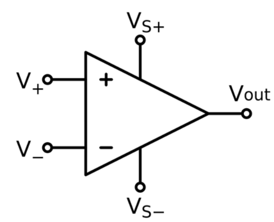
- Analyze, design, build, and characterize different amplifier circuits using an op-amp and discrete components

Amplifiers are an important class of analog signal processing circuits, as the signal coming into most sensors or receivers is too weak to be used directly as it is. For example, a sensor/receiver might see 5-50mV (in some cases much less) instead of a nice 0.5-5V signal that matches the data recording part of the system—so a voltage amplifier with a gain of ~10-100 can provide a more suitable signal to use and record with the receiver electronics.

In some cases, the mismatch is in the voltage scale, and in other cases, the mismatch is in the current or power requirements. For example, the input signal might have plenty of voltage, but not enough power to drive current through the filter that you are using, so an amplifier can help maintain the voltage level. Another example is an electric guitar, which produces fine signals in the mW range, but requires an amplifier to make those signals move a speaker enough to project the sound over a large stadium (requiring ~100kW).

While it is possible, and sometimes essential, to build amplifiers from discrete components (e.g., transistors and inductors) this is more complex than necessary for most applications. Especially for the low-frequency (<100kHz) applications in this module and class, **operational amplifiers** provide a convenient option for getting amplification with low complexity.

Operational Amplifiers ('op-amps') are integrated circuits (ICs) that put lots of components together into a convenient package that can be easily plugged into an electronic circuit to support various types of amplifier circuits with different properties appropriate to the application: voltage/current, low-gain/high-gain, low-noise, single-sided, wide-band/narrow-band, etc.



Before op-amps, it required much more skill, time, and troubleshooting to amplify a signal. Complexities of op-amps are far greater than can be covered in this module, and it takes years to get to know and exploit their various strengths and limitations, but by the end of this module, you will be able to build a decent sensor signal amplifier for your pinball apparatus armed with ~$1 in components, a datasheet, and a few equations. For example, if the signal received by the sensor in the game field appears too low to be captured and analyzed reliably by the ADC then an op-amp based amplifier can be incorporated to fix that. The following will augment the lecture materials and introduce basic op-amp analysis tools and canonical op-amp circuits.

3.3.3.a Analysis and Design of Op-amp Circuits (pre-lab activity)

The best way to first understand op-amps in their glory is to take their ideality to heart and ignore how it works. This is why they call them the *golden rules.* Here are the rules you apply to your circuits involving op-amps that will let you analyze them:

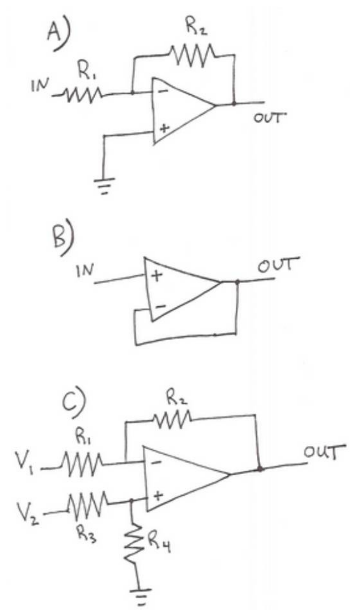
Golden Rules of Opamps

1. The **inputs (V+ and V-) draw no current.**

2. The output has zero impedance. This means that the output voltage is independent of output current, and thus, **the output voltage is not affected by the load at the output.**
3. **In *negative feedback*** (ie. there is a path between output to the negative input), the output attempts to do whatever is necessary to make the voltage difference between the inputs zero. Thus, **the inputs are always equal.** In *positive feedback*, no such behavior exists, and your negative and positive inputs can be different.

Other behaviors of opamps to take note of:

4. The power rail to the opamp (Vs+, Vs-) limits the output range. Thus, Vout can never go beyond this range.
5. when there is no feedback (e.g. no path from output to an input), the op-amp acts as a comparator (i.e. if $V_+ > V_-$, out = V_{s+} ; $V_+ > V_-$, out = V_{s-})



For the following questions, refer to the figure on the right (note, an opamp is a powered device, and therefore has a Vs+ and Vs- (typically GND); we have not shown it here for the sake of simplicity.)

3.3.3.a.1 Analyze the circuit pictured in 'A'

- Relate the input voltage to the output voltage with an equation.
- This is an **inverting** amplifier. Why?

3.3.3.a.2 Analyze the circuit pictured in 'B'

- Relate the input voltage to the output voltage with an equation.
- This is called a **buffer**. What is one potential use for this circuit (see Rule 3 and think what might happen if you chain multiple filters together)?

3.3.3.a.3 Analyze the circuit pictured in 'C'

- Relate the input voltage difference (V1-V2) to the output voltage with an equation. Assume R1 = R3 and R2 = R4 and calculate the gain.
- This is called a **differential amplifier**. How is this different from the circuit in 3.3.3.a.1?

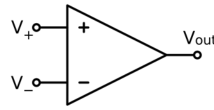
3.3.3.b Building and Using Op-amp Circuits (prelab + lab activity)

3.3.3.b.1 Amplifying a weak signal using a inverting amplifier

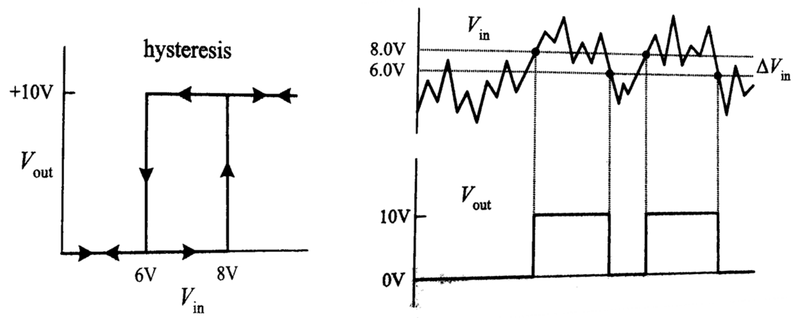
- Build a non-inverting amplifier for an approximately 10x gain. Avoid using resistors any smaller than 1kOhm, since the circuit will draw more current/power.
- On the oscilloscope, measure the outputs given the following inputs and document what you observe (sketch or screen capture):
  - (i) Square wave, 0 to 0.5V
  - (ii) 2V DC. What happens if you set your gain too high?

3.3.3.c Comparator with Hysteresis

Below is a comparator which outputs high when an input voltage exceeds a reference voltage; it outputs low when the input voltage is below the reference voltage. In this case, input voltage is to V+ and reference voltage is to V-; i.e. if V+ - V- > 0, Vout = Vs, otherwise Vout = 0 (assuming voltage rails on the comparator is Vs and 0).



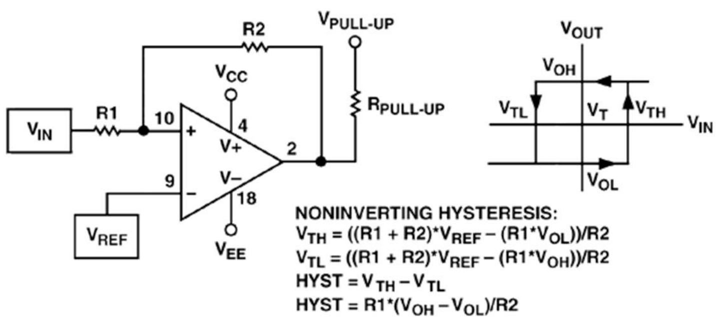
You can implement this using an opamp. Vout will be at the opamp's positive supply rail at HIGH and negative supply rail at LOW. A problem arises when your input is noisy, which can cause the difference of V+ and V- to flicker back and forth near the threshold. Ideally you an input voltage to slightly exceed the threshold and lock in "high", but dip down slightly below the threshold before it switches back to "low". This looks like the below (assume ~7V threshold, arrows indicate path of input voltage Vin):



This is called hysteresis, and to do this on a comparator is not too difficult.

Use this if you are trying to covert an analog signal to a binary on/off signal, and you wish to make sure it only captures a transition from "on/off" cleanly (e.g. a rising or falling edge, even when noisy, is only captured once). For example, a light signal that gets blocked...

So how do you design a hysteretic comparator? First, decide what you want your input voltage (Vin) to compare to (Vref). Then decide, what is the width of the hysteresis?



3.3.3.c.1 Designing a hysteretic comparator (Prelab Activity)

- Design a comparator with hysteresis with a VT = 2.5V, a width of 1V, and a Vout = {0,~5} V.

3.3.3.c.2 Verify the behavior of a hysteretic comparator (Lab Activity)

- Implement the circuit above. Check using the function generator, giving it a 0-5V sine wave.

Use this comparator circuit for at least one of your circuit designs in the following sections. **Note:** there are pre-packaged ICs with hysteretic behavior (Schmitt triggers with both inverting and non-inverting outputs) that are often very convenient but not flexible (upper and lower trigger points are set by the supplier). Explore the part SN74HC14 we have in stock and you can choose to use it (as opposed to 3.3.3.c.2) for your circuit design.

3.3.3.d Phototransistor Proximity / Obstruction Sensor Design

In this activity, you will design, build and test a simple optical sensor. You may use this device to detect the ball at certain locations on the playfield. **You may use this part as a way to complete 3.3.5.** Two configurations are possible: with the LED and the phototransistor facing each other and a ball running through it, or and LED and phototransistor pair side-by-side, and a ball runs past it and reflects the LED light into the receiver. Here are some things to remember:

1. Your sensor should not be affected by ambient light.
2. Your sensor should have a good range of voltages that can be picked up by something like an Arduino input pin.
3. Your Arduino can measure a voltage between 0-5V at any point in time, but this poses a problem if the signal is oscillating. What filter stage might you create to detect the peaks effectively? (There are a variety of single or multi-stage methods that could be used, see **Module 3 Extra Circuits** for some accessory circuits that will be helpful).

3.3.3.d.1 Plan: Design the system, circuit, and tests

- Based on your experience with the circuit that you built and tested in 3.3.1.b-c, design a system for detecting an object and create requirements and a block diagram of the system. Incorporate the circuit above to provide a 0 or 5V output, so that above a certain light threshold, you get a 5V, otherwise a 0. This can be read on a digital input.

3.3.3.d.2 Build

- Build the proximity sensor circuit according to the design.

3.3.3.d.3 Test

- Verify that the sensor works as expected.

3.3.3.d.4 Assess

- Analyze verification test results – did it work out as expected?

3.3.4 Controlling Many Digital Outputs

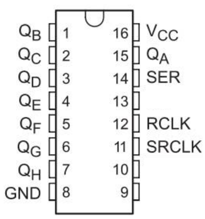
This activity introduces digital logic circuitry which can control complex input/output pairs using simple conditional rules.

3.3.4.a Four-Digit 7-Segment Display Design

In this activity, you will design, build, and test a circuit to display digits on at least four 7-segment display digits. Using 7-segment displays is a simple way to receive readable visual feedback from your system.

To display visual scoring information of the pinball game you need to design a circuit to light **four 7-segment display digits**. You have only one Arduino with a very limited amount of digital pins available, but shift registers (SN74HC595) will help reduce the number of pins needed to control the displays.

Serial-in-parallel-out shift registers can be used to convert serial data to parallel data. This is useful when you want to control many digital outputs using a sequence of bits coming from a single Arduino pin. Our shift registers have 3 inputs (SER, SRCLK, RCLK) and 8 outputs (QA-QH). SER is the serial input which accepts digital sequences (e.g., 10010101) and SRCLK is the clock signal that must pulse once for every bit pushed through SER. The outputs QA-QH will update to reflect the last 8 bits you sent through SER only when there is a rising edge (i.e., a LOW to HIGH transition) on RCLK. The example code below demonstrates how easy it is to use a shift register using an Arduino.



Basic Arduino Code to Handle Shift Registers

```
digitalWrite(RCLK, LOW);                   // Force register clock to 0
shiftOut(SER, SRCLK, LSBFIRST, B10010101); // Internally load serial data
digitalWrite(RCLK, HIGH);                  // Rising edge to update outputs
```

Note, a 7 segment LED module can be lit up using only 1 shift register. However you need to light up 4. Do you need 4 shift registers? *No*. You can do a trick by using a single transistor for each set of LEDs that will disconnect them from power when they aren't being used, so that each 7 segment module can be connected to the same QA-QB pins. How would this be designed? (hint – we have P-MOSFETs that might prove useful…)

3.3.4.a.1 Plan: Design the system, circuit, and tests

Create a circuit diagram of the display system of four 7-segment displays. Check your circuit with a TA before you move on. Create an Arduino program that can display 4-digit numbers as you desire. You may want to check online for code involved in shift registered and using the "shiftOut" function. Don't forget to ensure that maximum current ratings on 7-segment diodes are preserved.

TIP: Consider how you will mount the 7 segment display. Because this may require a significant amount of soldering, you may want to consider soldering leads to the 7 segments first before you run them onto your breadboard. (we have ribbon cables to make it tidy…)

3.3.4.a.2 Build

- Build the proximity sensor circuit according to the design.

3.3.4.a.3 Test

- Perform circuit subsystem verification tests, based on specifications/plans
- Record appropriate data to document test results

3.3.4.a.4 Assess

- Analyze verification test results
- Document analysis in lab notebook and in a short electronic write-up

3.3.5 Optical Detector and Scoring Feedback Subsystems Design Cycle

This activity ties together all the previous subtasks in 3.3.1-3.3.4, with the pinball machine system design from Module 1, to create a functional set of pinball detector and scoring feedback sub-systems. **You may keep the circuits above and simply use them to complete this part of the module.**

3.3.5.a Optical Detector and Scoring Feedback Subsystems Design Cycle: Plan

- Create a specific/refined subsystem design and block diagram that meets the pinball machine system requirements.
- Show the circuit design for the subsystems that can meet all of the requirements (if you are using the same circuit as a previous part of the Module, refer to this circuit here; otherwise draw out any new circuits).
- CAD the mount for your optical sensor/LED pair and your 7-segment display. Tip: if you want to use multiple optical sensor/LED pairs on your board, consider a modular mount that could be replicated over and over.
- Use the subsystem block diagram and design to write circuit design specifications
    - E.g., 'the sensors must be able to detect a ball rolling past at speeds up to X m/s and at Y mm away from the sensor.'
    - E.g., 'the value shown on 7-segment display must increase in response to ball detection'
- Use the specifications/metrics to create a verification testing plan that you will execute.

3.3.5.b Optical Detector and Scoring Feedback Subsystems Design Cycle: Build

- Fabricate circuits on a breadboard according to the plan/layout

3.3.5.c Optical Detector and Scoring Feedback Subsystems Design Cycle: Test

- Perform circuit subsystem verification tests, based on specifications/plans
- Record appropriate data to document test results

3.3.5.d Optical Detector and Scoring Feedback Subsystems Design Cycle: Assess

- Analyze verification test results
- Create notes for potential redesign (in lab notebook/digital format)

Note: along with some of the circuits from activities in 3.3.2 and □, in the Scherz textbook, you may find the circuit topologies in supplemental document provided by the instructor.

Looking Ahead:

Most likely, if you did all the prelabs and came prepared with circuit diagrams, you will have completed the module with time to spare.

You now have all the skills to integrate everything together for the playfield! When you are ready, this is the best time start working out the **CAD** for the complete playfield, which is one half of the module 4 deliverables (the other half being the built playfield) which will need to be a working system at the end of Module 4. Time to get a head start!

At this time you can also consider adding in more sensors (e.g. replicating your optical sensor) or consider placements of flashing patterns of LEDs, which will help make your final project more interesting and fun. Also, consider implementing one or more sensors that are available to you, such as **piezo-electric film sensors** (registers a voltage spike when hit), **spring-loaded switches** (registers a signal when switched on), **Hall Effect sensors** (measures nearby magnetic fields --- which a ball can amplify and disturb when passing through in a recognizable way), **conductors like copper tape** (could be used as a switch). There is very little circuitry involved for these, and the instructor/TAs will be happy to help you integrate it into your system.