San José State University

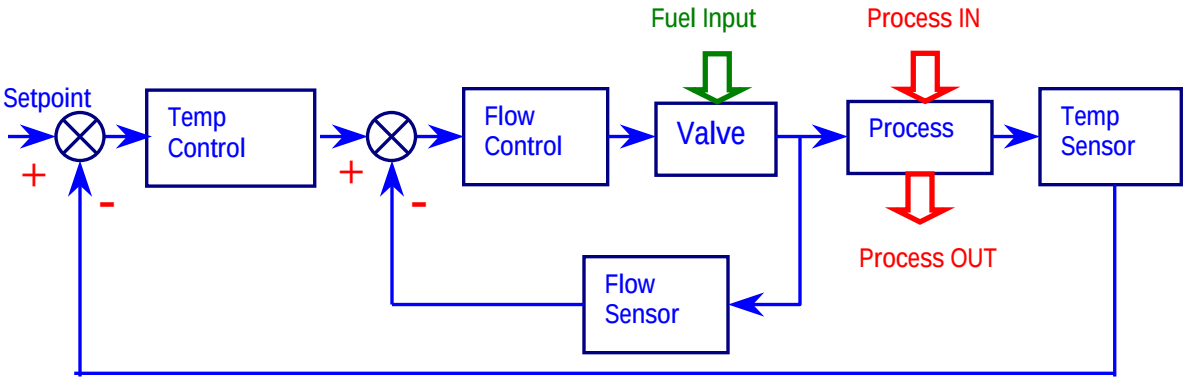
Aviation & Technology

Optoisolator

# TECH 167

# Control Systems

# Laboratory Experiments Manual




Dr. Julio R. García

2007

# Introduction

To get the most out of this lab manual it is recommended that students follow the following process:

1. Review a PowerPoint presentation about the lab activity.
2. Observe the lab outcome using LabVIEW
3. Build, troubleshoot and test the circuit using Multisim
4. Hardwire, troubleshoot and test the circuit using real devices and components
5. Compare measurements obtained in Multisim with the hardwired ones
6. Solve a problem applying learned concepts

The LabVIEW files can be open with version 7.1. The Multisim files can be open with version 7 or higher and are available to instructors. It is recommended that students create the circuit for each lab in Multisim so they can get proficient in the use of this software.

This material is based upon work supported by the National Science Foundation under Grant No. 0411330.

Any opinions, findings and conclusions or recommendations expressed in this material are those of the author and do not necessarily reflect the views of the National Science Foundation (NSF).

## TABLE OF CONTENTS

Page

# LAB 1 – WHEATSTONE BRIDGE

## Objectives

1. Build, test, and troubleshoot an application of the Wheatstone bridge to convert resistance changes to voltage changes using Multisim.
2. Hardwire the Wheatstone bridge of objective 1 and compare the measurements of the hardwire circuit with the measurements obtained using Multisim.

## Preliminary Information

Basically, the Wheatstone bridge is made up of two voltage dividers powered by a dual-power supply or a single source (see figure 1-1). Among the junctions of the voltage dividers, a galvanometer (a very sensitive current meter) has been connected with the purpose of monitoring the current flow from one voltage divider to the other.

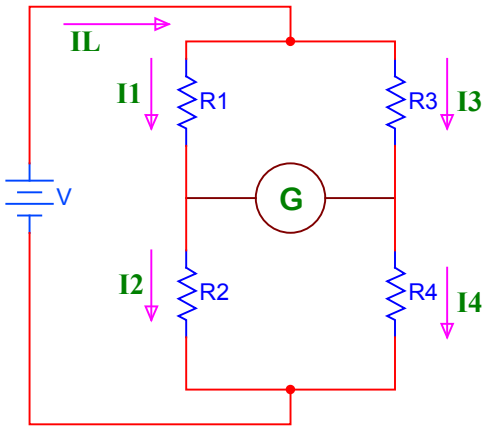



**Figure 1-1**

When DC current doesn't flow through the galvanometer, we say that the Wheatstone bridge is balanced and the following is achieved:

$$I_1 R_1 = I_3 R_3 \quad (1)$$
$$I_2 R_2 = I_4 R_4 \quad (2)$$
$$I_1 = I_2 \quad (3)$$
$$I_3 = I_4 \quad (4)$$

Replacing equations 3 and 4 in equation 2:

$$I_1 R_2 = I_3 R_4 \quad (5)$$

Finding $I_1$ in equations 1 and 5:

$$I_1 = \frac{I_3 R_3}{R_1} \quad (6) \qquad I_1 = \frac{I_3 R_4}{R_2} \quad (7)$$

Equating equations 6 and 7:

$$\frac{I_3 R_3}{R_1} = \frac{I_3 R_4}{R_2} \qquad (8)$$

Eliminating the term $I_3$ and ordering equation 8, we have:

$$\frac{R_1}{R_2} = \frac{R_3}{R_4}$$

**In conclusion, when the Wheatstone bridge is balanced, the following relationship is established:**

$$\frac{R_1}{R_2} = \frac{R_3}{R_4} \qquad (9)$$

The Wheatstone bridge has multiple applications; initially it was used to find the unknown value of a resistance by means of some modifications to figure 1-1.

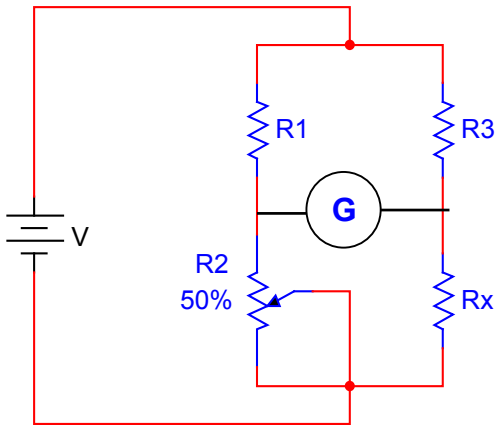



**Figure 1-2**

Assuming the values of R1 and R3 are known, and R2 has been replaced by a potentiometer, we will connect the unknown resistance whose value we want to find in the R4 position (in this case we will call it Rx).

**To find the unknown value, Rx; we follow the following procedure:**

1. We vary the potentiometer R2 until the galvanometer shows zero current. In this condition, we say that the Wheatstone bridge is balanced and we apply equation 9.

2. The R4 label is replaced by Rx in equation 9 and we find the value of Rx.

$$\frac{R_1}{R_2} = \frac{R_3}{R_X} \qquad (10)$$

$$RX = \frac{R2\,R3}{R1} \qquad (11)$$

With equation 11 determine the value of the unknown resistance.

**Notes:**

1. **The potentiometer and the unknown resistance Rx can be located in any part of the voltage dividers, being careful to replace them appropriately in equation 9.**

2. **The values of the known resistances can be the same or different.**

The Wheatstone bridge is widely used in electronic instrumentation by substituting one or more resistances with sensors. When the resistances of the sensors change, we obtain an output that is proportional to this variation. At the output of the Wheatstone bridge, instead of the galvanometer, we can connect an amplifier circuit that will allow us to activate a control system.

## Problem Statement

Build a circuit that converts resistance changes to voltage changes. We need to drive a 10 kΩ load and we have only a dual supply of ± 10 V. In addition, we want a gain of 2.2. One of the resistances is an RTD (Resistance Temperature Detector) that has a nominal resistance of 150 Ω at 25 °C. Due to product specifications two resistances in the Wheatstone Bridge (R1 & R2) should be 2.2 kΩ and 1 kΩ.

a. Design the appropriate circuit. Don't forget to design the interface circuit too.
b. Test the circuit under different RTD values (this is what will happen when the temperature changes).

Given the design requirements, and the block diagram, the schematic diagram for the circuit design is shown in Figures 1-3 and 1-4 respectively.

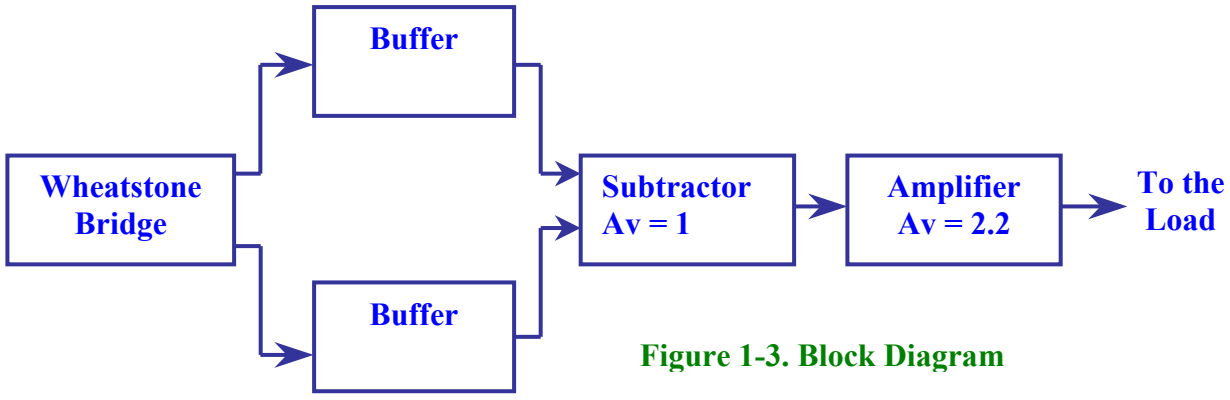



**Figure 1-3. Block Diagram**

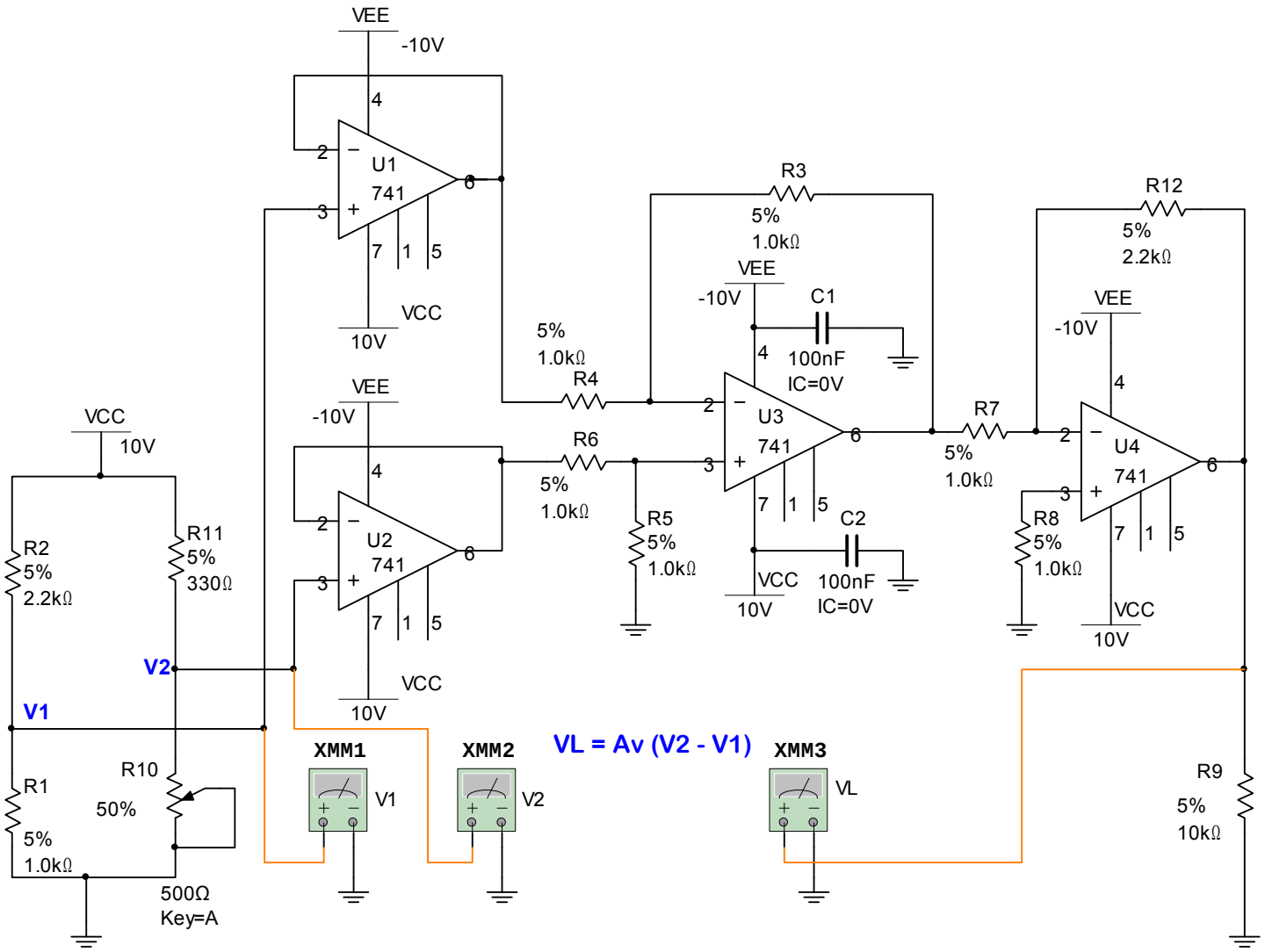



**Figure 1-4. Schematic Diagram**

## Circuit analysis

Basically, the Wheatstone bridge is made up of two voltage dividers (R2 - R1 and R11-R10) and powered by a single power supply ($V_{CC}$).

The resistances can have different values, however one of them (in this case R10) is a variable resistance that simulates the RTD and allows balancing the circuit.

The Wheatstone bridge is balanced when:

$$V1 = V2 \quad \text{then:} \quad \frac{R2}{R1} = \frac{R11}{R10} \qquad (1)$$

Taking into consideration that two of the resistances should have 2.2 kΩ and 1 kΩ, we will locate them in the positions of R2 and R1 respectively.

The RTD is essentially a temperature dependent resistor with a positive temperature coefficient. Its value increases when the temperature increases and vice versa. Potentiometer R10 will simulate the behavior of the RTD.

According to the specifications of the laboratory, at a 25º C temperature the RTD has a resistance of 150 Ω. This specification will be our reference point for the Wheatstone bridge. With this resistance value, we calculate the value of R11 using equation (1) in the following way:

$$R11 = \frac{(R2 * R10)}{R1}$$

$$R11 = \frac{(2.2\,K\Omega * 150\,\Omega)}{1\,K\Omega} = 330\,\Omega.$$

Since the reference point is 150 Ω, R10 should be greater than 150 Ω. We'll use a 500 Ω potentiometer.

Varying R10, to simulate the temperature variations, the circuit becomes unbalanced, producing small voltage variations (between points V1 and V2 of the Bridge). These voltage variations will be amplified by the 741s circuits to activate the 10K Ω load.

U1 and U2 are voltage followers (buffers), it is designed to prevent loading of V1 & V2 by the following amplifier stage. U3 is a differential amplifier with unity gain and its output is the potential difference between V1 and V2. The voltage gain of the differential amplifier is giving by the ratio $\frac{R3}{R4}$ or by $\frac{R5}{R6}$

Since the voltage gain should be 1, then R3 = R5 and R4 = R6. This can be easily accomplished by selecting R4 = R6 = 1 kΩ and R3 = R5 = 1 kΩ.

U4 is an amplifier with gain of 2.2 according to the problem requirements. The gain of this circuit depends on the ratio R12/R7, thus R12 should be 2.2 times greater than R7. Therefore, if we select R7 = 1 kΩ, then R12 should be 2.2 kΩ.

**Notes:**

a. **Use a linear potentiometer in Multisim and in the hardwired circuit.**
b. **Choose the potentiometer with a 1% increment in Multisim.**
c. **Select Vout = $V_{CC}$ – 2 V**

   **If $V_{CC}$ = 10V then Vout,max = 10V – 2 V = 8 V**

   **Therefore, never exceed Vout = 8 V when using $V_{CC}$ = 10V.**

## Procedure/Tasks Using Multisim

1. Build the circuit indicated in Figure 1-4 in Multisim.

2. Adjust R10 (by pressing the A or SHIFT-A keys) until V2 is approximately equal to V1. When this happens, VL should be approximately equal to 0. Is this correct? Explain.

Make V2 > V1 or V2<V1 and observe the voltage at the load (R9). Complete the following table with three values of V2 > V1 and three values of V2 < V1.

| V1 | V2 | VL |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

3. Replace the three 741 ICs by a single quad op-amp such as the LM324AJ. Note that the LM324AJ has four sections A, B, C and D. Each section shows up on the screen with its own power supply terminals, pin 4 (+V) and pin 11 (– V). You only need to connect the respective power supplies to any section: pin 4 should be connected to VCC and pin 11 should be connected to ground.

4. Adjust R10 (by pressing the A or SHIFT-A keys) until V2 is approximately equal to V1. When this happens, VL should be approximately equal to 0. Is this correct? Explain.

5. Make V2 > V1 or V2<V1 and observe the voltage at the load (R9). Complete the following table with three values of V2 > V1 and three values of V2<V1. Use the same values you did in step 3.

| V1 | V2 | VL |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

6. Comparing your answers in steps 5 and 6 with your answers in steps 2 and 3, can you use a single LM324AJ instead of three 741 ICs?

## Questions

1. Is it possible to use a single supply? Why or why not?

2. How can you determine the maximum potential difference between V1 and V2 to prevent the circuit reaching saturation voltages?

3. If R4 opens, how can you determine the voltage load (VL)? **Hint:** refer to the appropriate equation.

4. If the voltage gain is 3.3 instead of 2.2, what modification(s) would you do to the circuit?

## Procedure/Tasks – Hardwired Circuit

Now that you are familiar with the Wheatstone bridge using Multisim it's time to hardwire the circuit shown in Figure 1-3 using real components. However, before continuing, you are going to indicate the list of components and the equipment that you need to complete this task.

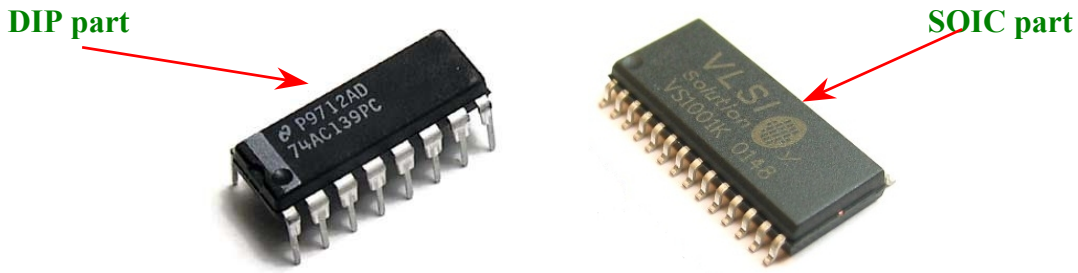
### List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

### Equipment

It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin. This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.

- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.
- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.

The later lab experiments also include this checklist for your reference.

1. Adjust R10 potentiometer until V2 is approximately equal to V1. When this happens, VL should be approximately equal to 0. Is this correct? Explain.

2. Make V2 > V1 or V2<V1 and observe the voltage at the load (R9). Complete the following table with three values of V2 > V1 and three values of V2<V1. Try to match the values you used in Multisim.

| V1 | V2 | VL |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

## Questions

1. Are the measurement obtained in the hardwired circuit similar to those obtained in Multisim? Explain any difference.

2. What difficulties did you encounter in hardwiring the Wheatstone bridge?

## Conclusions

After completing the Wheatstone bridge both in Multisim and hardwiring, what are your conclusions?

## Advanced Design

### If you want to earn extra credit, try the following design.

You are a manufacturing engineer working for an injection molding company. Your supervisor requests that you design and implement a circuit to measure temperature. The objective is to measure the environmental temperature in the injection molding machine area to determine how much it varies over the course of a day. He says these temperature measurements will vary widely, because he wants to measure ambient air temperature as well as surface temperature on the outside of the molds being used in the injection molding machines during the molding process.

### Knowns:

1. The temperature range your supervisor wants to measure has large range: anywhere from room temperature which is ~ 20°C to ~150°C (the maximum surface temp of a plastic mold during the molding process).
2. Your boss wants to measure to an accuracy of +/-1.0°C. This implies an ideal resolution of ~0.1°C. (Usually, we want the resolution to be better than the accuracy we need to measure. Cumulative errors in the sensor & circuit design as well as the implementation must add up to be less than the desired accuracy in order to meet performance requirements.)
3. Because of the possible high temperatures measurement, the Wheatstone Bridge circuit has to be located some distance away from the temperature sensor
4. The environment is an electrically noisy environment, due to large industrial machines being used.
5. Types of temperature sensors available for implementation: RTD, thermistor, and thermocouple (TC).
6. Because the environment has a widely varying temperature, use of a stable reference temperature is not practical.

### Unknowns:

1. What kind of circuit is needed for the Wheatstone bridge?
2. What are the component values needed in the circuit?
3. Which temperature sensor is best for this application?
4. How would you properly interface the temperature sensor to the circuit?

# LAB 2 – SIGNAL CONDITIONING CIRCUIT

## Objectives

1. Build, test, and troubleshoot an analog signal conditioning circuit using Multisim.
2. Design an analog signal conditioning circuit to provide a range of desired output voltages in respond to a certain range of input voltages.
3. Hardwire the analog signal conditioning circuit of objective 1 and compare the measurements of the hardwired circuit with the measurements obtained in Multisim.

## Preliminary Information

The signal conditioning circuit is an electronic circuit that converts signals provided by a sensor to useful electric signals. These electric signals must meet specific criteria so that they are correctly interpreted and processed by the rest of the system's circuitry. The use of Op-amps allows signal conditioning circuits to be more compact and precise in their implementations.

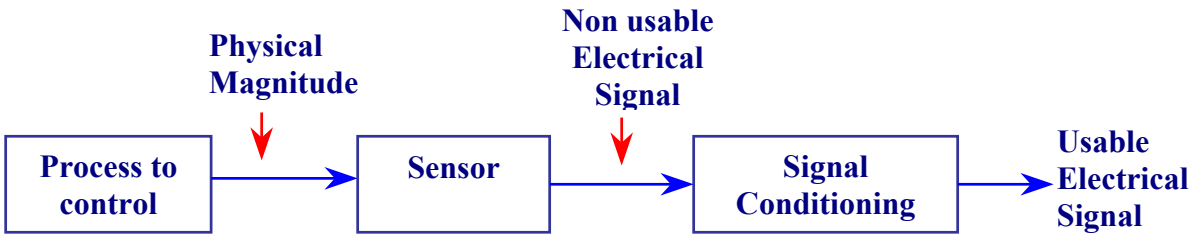



**Figure 2-1. Signal conditioning is important in a system of process control.**

**Signal conditioning carries out one or several of the following actions:**

1. **Change voltage levels** so that they are compatible with the following circuitry.
2. **Convert current to voltage**. Some sensors, such as the NTC (Negative Temperature Coefficient) and the PTC (Positive Temperature Coefficient) RTD (Resistance Temperature Dependent) convert the variations of the process to control resistance variations. Signal conditioning circuits provide the necessary current that converts a resistance variation to an appropriate voltage.
3. **Linearize** and compensate sensor's non-linear variations. This linearization is the conversion of non-linear signals into linear signals by using Taylor series.
4. **Convert the analog signal to digital signal.** The signal conditioning circuit ensures the analog signal is at levels that are compatible with the analog to digital conversion

circuitry. After having transformed the analog signals into digital, we can store their numerical representations on a memory, process them with an application program, display them on a monitor, send them through the Internet to another place, or print them.

5. **Convert the analog signal to current signal.** It is an industry standard that the control range is normalized from 4mA to 20mA dc. The minimum value of 4mA is defined as **"zero active"** because it offers the advantage of being able to detect an interruption of the connection between the sensor and the signal conditioning circuit. When the signal conditioning provides 0 mA at its output, it will be an indication that the sensor is defective or some other faulty circuit.

6. **Eliminate undesired signals such as noise.** The elimination of the undesired noise is carried out with analog filters (i.e., RC, RLC, etc.) or in more challenging applications with digital filtering.

7. **Isolate the sensor.** The signal conditioning circuit should isolate the sensor electrically when the sensed signal contains high voltage pulses that can affect the measurements and the subsequent circuitry of the system.

8. **Multiplex different signals**. Multiplexing consists of measuring several signals with a single measurement device. This is feasible when the signals to measure change slowly, such as when measuring temperature.

**Optoisolator**


In some applications, it is not necessary to process the entire range of the sensed signals through an analog-to-digital converter. Some applications might require only that the sensor acts as a switch. Therefore, the signal conditioning circuit in this case would consist of an Op-amp comparator that acts as an analog-to-digital converter of one bit.

The other applications process a complex waveform from a sensor. Therefore, it might be preferable to capture the energy of the waveform for later processing. In this case, the signal conditioning circuit consists of a rectifier and a filter.

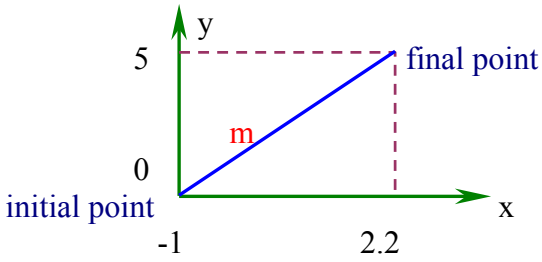
## Problem Statement

Build a circuit that converts an input voltage of – 1 V to + 2.2 V to an output voltage of 0 to 5 V. We have a dual supply of ± 12 V available and we need to drive a 10 kΩ load. Test your designs under different input voltages and verify that your output voltages are within ± 5% of the calculated ones.

## Solution

The input voltage (– 1 V to + 2.2 V) is the independent variable while the output voltage (0 to 5V) is the dependent variable.

By plotting the independent variable (x axis) and the dependent variable (y axis), we have:




Connect the intersection points, we find that the graph is a straight line, then:

$$y = mx + b \qquad (1)$$ **equation of the straight line.**

where:

$$m = \frac{y_1 - y_0}{x_1 - x_0} \qquad (2)$$ **slope of the straight line.**

Replace values in the slope of the straight line, we have:

$$m = \frac{5 - 0}{2.2 - (-1)} = 1.56$$

To find the value of the constant (b), in the equation of the straight line we replace the variables (x, y) for the coordinates of the initial point (–1, 0) or the coordinates of the final point (2.2, 5), through this linear equation:

$y = mx + b$ equation of the straight line

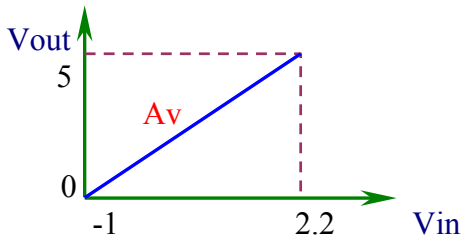
Use coordinates of the initial point (–1, 0) ➔ $0 = 1.56(-1) + b$ ➔ **b = 1.56**

Use coordinates of the final point (2.2, 5) ➔ $5 = 1.56(2.2) + b$ ➔ **b = 1.56**

**The result is the same; however, it is simpler to use the initial point as seen above.**

Replacing the variables and the constant of the linear equation for equivalent electronic terms we'll have:

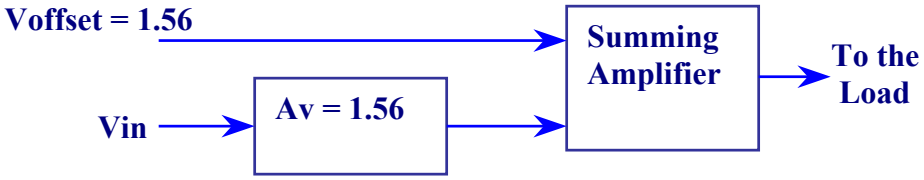
y = Vout
m = Av = 1.56
x = Vin
b = Voffset = 1.56




**The constant of the equation can be called any name; in this case we'll call it Voffset.**

Then: Vout = Av (Vin) + Voffset (4)

**Equation (4) indicates that we should use a circuit whose block diagram is:**



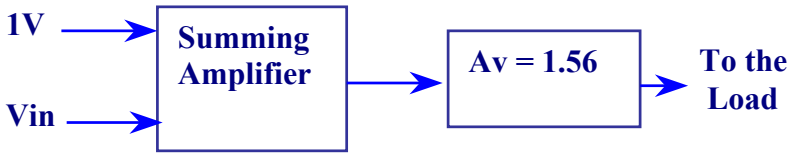

Replacing values in equation (4), we have:

Vout = Av(Vin) + Voffset (4)

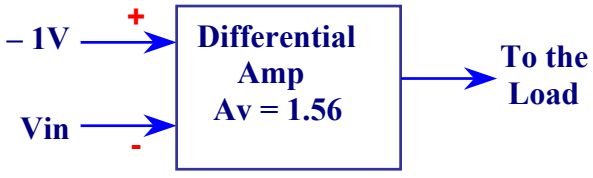
Vout = 1.56 (Vin) + 1.56

Vout = 1.56 (Vin + 1) (5)

**Equation (5) provides a second possible solution; the block diagram is the following:**




**The third possible solution comes from the previous circuit whose simplification is:**



**Note: In all the solutions, it is recommended to use an op-amp voltage follower between Vin and the input of the circuit. The purpose is to maintain impedance matching and prevent excessive loading of previous circuit stages.**

The design of the problem statement will be accomplished by using solution 2. Figure 2-2 shows the implementation of equation 5.

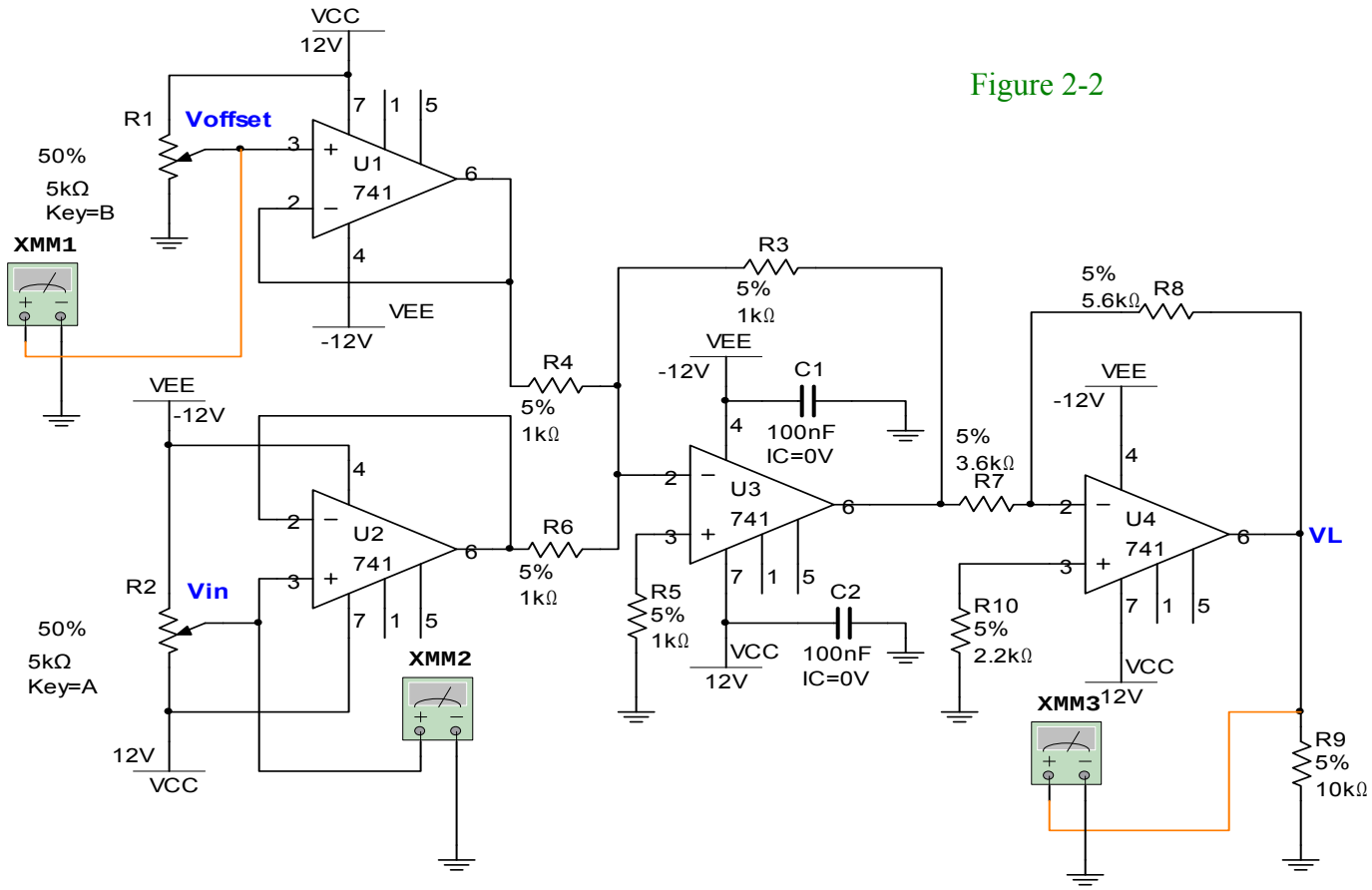



Figure 2-2

## Circuit analysis

Op-amps U1 and U2 are configured as voltage followers. **It is always a good practice to use a voltage follower or buffer between the input signal and the next stage**. This ensures the stability of the circuit. U3 is a summing amp with Av = 1. The gain depends of the relationship between resistors R3/R4 or R3/R6. Since the gain of the circuit is unity then R3 = R4 = R6; in this case we select 1 kΩ for each resistor.

U4 provides the circuit gain which is determined by the ratio between resistors R8/R7. According to equation 5, Av = 1.56 then R8 must be 1.56 times greater than R7. Assuming R7 = 3.6 kΩ, then the value of R8 would be 5.61 kΩ but we choose the closer standard value of 5.6 kΩ.

Notice in Figure 2-2 that we are using Voffset = –1 V (in this case due to components limitations it is –1.001 V).

## Procedure/Tasks Using Multisim

1. Build the circuit indicated in Figure 2-2 using Multisim (Select 2% increments for all the potentiometers).

2. Adjust R1 (by pressing the B or SHIFT-B keys) until Voffset is approximately equal to + 1 V.

3. Adjust R2 (by pressing the A or SHIFT-A keys) until the input voltage (Vin) is approximately equal to – 1 V. When this happens, the output voltage or $V_L$ should be approximately equal to 0.

4. Adjust R2 (by pressing the A or SHIFT-A keys) until the input voltage (Vin) is approximately equal to + 2.2 V. When this happens, the output voltage or $V_L$ should be approximately equal to + 5 V. What you have completed in steps 3 and 4 is to vary the input voltage, Vin from the given input range of –1 V to + 2.2 V. The output voltage range should be 0 to + 5 V. Is this correct? Explain any discrepancies.

5. Vary the input voltage, Vin from –1V to + 2.2 V. Adjust Vin to the values indicated on Table 2-1 and record the obtained output voltages, $V_L$.

**Table 2-1. Output voltage versus input voltage using Multisim**

| Vin | $V_L$ |
|---|---|
| – 1 V | |
| 0 | |
| 0.5 V | |
| 1 V | |
| 1.5 V | |
| + 2.2 V | |

6. Replace the four 741 ICs by a single quad op-amp such as the LM324AJ. Note that the LM324AJ has four sections A, B, C and D. Each section will have its own power supply terminals pin 4 (+V) and pin 11 (– V). You only need to connect the respective power supplies to any section: pin 4 should be connected to VCC and pin 11 should be connected to ground.

7. Repeat step 5 and complete Table 2-2.

**Table 2-2. Output voltage versus input voltage using a Quad op-amp using Multisim**

| Vin | $V_L$ |
|---|---|
| – 1 V | |
| 0 | |
| 0.5 V | |
| 1 V | |
| 1.5 V | |
| + 2.2 V | |

8. Compare Tables 2-1 and 2-2. Are they about the same? Describe any discrepancies and explain the possible causes.

## Questions

1. Implement equation 5 by using solution 1 (Amplifier and Summer).

2. Design a circuit that converts input voltages of –1.5 V to + 3.5 V to output voltages of 0 to 8 V. We have a dual supply of ± 12 V available and we need to drive a 10 kΩ load.

## Procedure/Tasks – Hardwired Circuit

Now, you are familiar with the signal conditioning circuit using Multisim and it's time to hardwire the circuit shown in Figure 2-1 using real components. Repeat what you did in Lab 1, that is, going through a checklist of components and the equipment that you need to complete this task. This will reduce the careless errors and damaging to components.

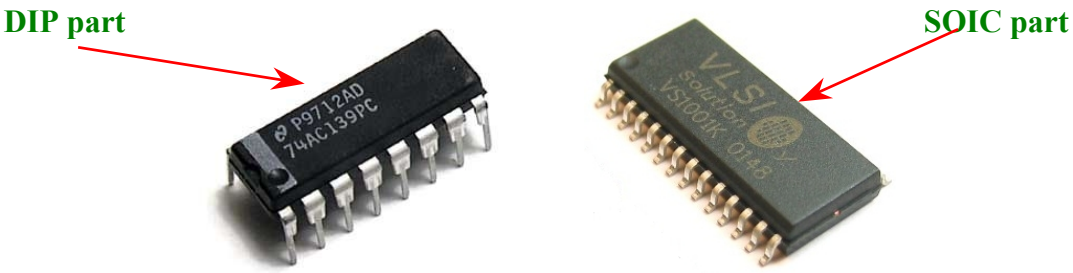
## List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

**Equipment**

It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin.

This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.

- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.
- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.

**The QJ49E Wheatstone Resistance Bridge is a highly stable, precision Wheatstone Bridge for the precision measurement of meters, components, windings and other instruments.**

## Hardwired Circuit Procedure

1. Adjust R1 potentiometer until Voffset is approximately equal to +1 V.

2. Adjust R2 potentiometer until the input voltage (Vin) is approximately equal to –1 V. When this happens, the output voltage or $V_L$ should be approximately equal to 0.

3. Adjust R2 until the input voltage (Vin) is approximately equal to + 2.2 V. When this happens, the output voltage or $V_L$ should be approximately equal to + 5 V. What you have done in steps 1 and 2 is to vary the input voltage, Vin, from the given input range of –1 V to + 2.2 V. The output voltage range should be 0 to + 5 V. Is this correct? Explain any discrepancies.

4. Vary the input voltage, Vin, from –1V to + 2.2 V. Adjust Vin to the values indicated on Table 2-3 and record the obtained output voltages, $V_L$.

**Table 2-3. Output voltage versus input voltage in the hardwire circuit**

| Vin | $V_L$ |
|---|---|
| – 1 V | |
| 0 | |
| 0.5 V | |
| 1 V | |
| 1.5 V | |
| + 2.2 V | |

## Questions

1. Compare the results of Tables 2-1 and 2-3. Are they about the same? Describe any discrepancies and explain what could be the possible causes.

2. What difficulties have you encountered in hardwiring the signal conditioning circuit?

## Conclusions

After completing the signal conditioning circuit both in Multisim and hardwire, what are your conclusions?

## Advanced Design

### If you want to earn extra credit, try the following design.

You work in a failure analysis laboratory whose sole goal is to protect the end users' safety from defect components and investigate the failures. Your manager (the VP of R&D) has given you the task of designing a circuit that can take an unconditioned output signal from an accelerometer sensor and convert it into a reliable signal that can be analyzed in LabVIEW. The accelerometer is used in a drop test fixture to measure the impact forces on DUTs (Device Under Test). Because the accelerometer is in an abusive environment (i.e., being dropped all the time), it has minimal internal circuitry. This minimal circuitry can only put out a relatively small bipolar voltage (as a function of impact forces). The output signal must be interfaced to a data acquisition card in a PC that can only accommodate 0 to 5V signal inputs.

### Knowns:

1. The accelerometer signal output is in the range of – 2.5V to +3.2V.
2. The accelerometer's output impedance is ~1.5KΩ
3. The signal conditioning circuit output must be 0V to 5 V (because the data acquisition card requires this input signal level to function properly)
4. The input impedance of the data acquisition card is high enough that circuit loading should not be a concern.
5. The circuit will be located in an electrically noisy environment (hydraulic actuators are driven to power the drop test fixture).

### Unknowns:

1. What is the circuit configuration needed to properly interface the accelerometer output to the data acquisition card input?
2. What are the component values needed to properly implement the circuit design?
3. How do we deal with the electrically noisy environment? (**Hint:** this will be addressed by choosing the proper circuit configuration and careful routing of the wires from the accelerometer to the signal conditioning circuit.)

# LAB 3 – DIGITAL TO ANALOG CONVERTER (DAC)

## Objectives

1. Learn how to design a DAC circuit that converts digital input signals to expected corresponding analog voltage levels.

2. Build, test, and troubleshoot a DAC circuit using Multisim.

3. Hardwire the DAC circuit of objective 1 and compare the measurements of the hardwired circuit with the measurements obtained using Multisim.

## Preliminary Information

Digital to Analog Converters or DACs are used to convert digital values, represented by a binary or BCD format, into a corresponding analog voltage level. The binary or BCD value to be converted is usually inputted into the DAC in a parallel fashion with each bit from LSB to MSB having its own corresponding digital input.

A **DAC's resolution**, or smallest representative analog voltage, corresponds to the number of bits it can accommodate and the reference voltage used. While 8-bit ($2^8 = 256$ analog value) DACs are usually commonplace in industry, DACs can range from simple 6-bit (64 analog value) configurations up to 20-bit (1048576 analog value) configurations.

**Typical DACs are available in either current output (IDAC) or voltage output (VDAC) configurations**. While the voltage output VDACs are more convenient to implement, they tend to be slower and more expensive than their current output counterparts (refer to figure 3-1 for a VDAC example).

Therefore, for high-speed applications, circuit designers usually choose current output IDACs and then use a high-speed op-amp to provide the I-V conversion at the output of the DAC. For some low cost applications, an IDAC with a simple RC filter on its output is often enough to meet certain non-demanding, high-input impedance applications.

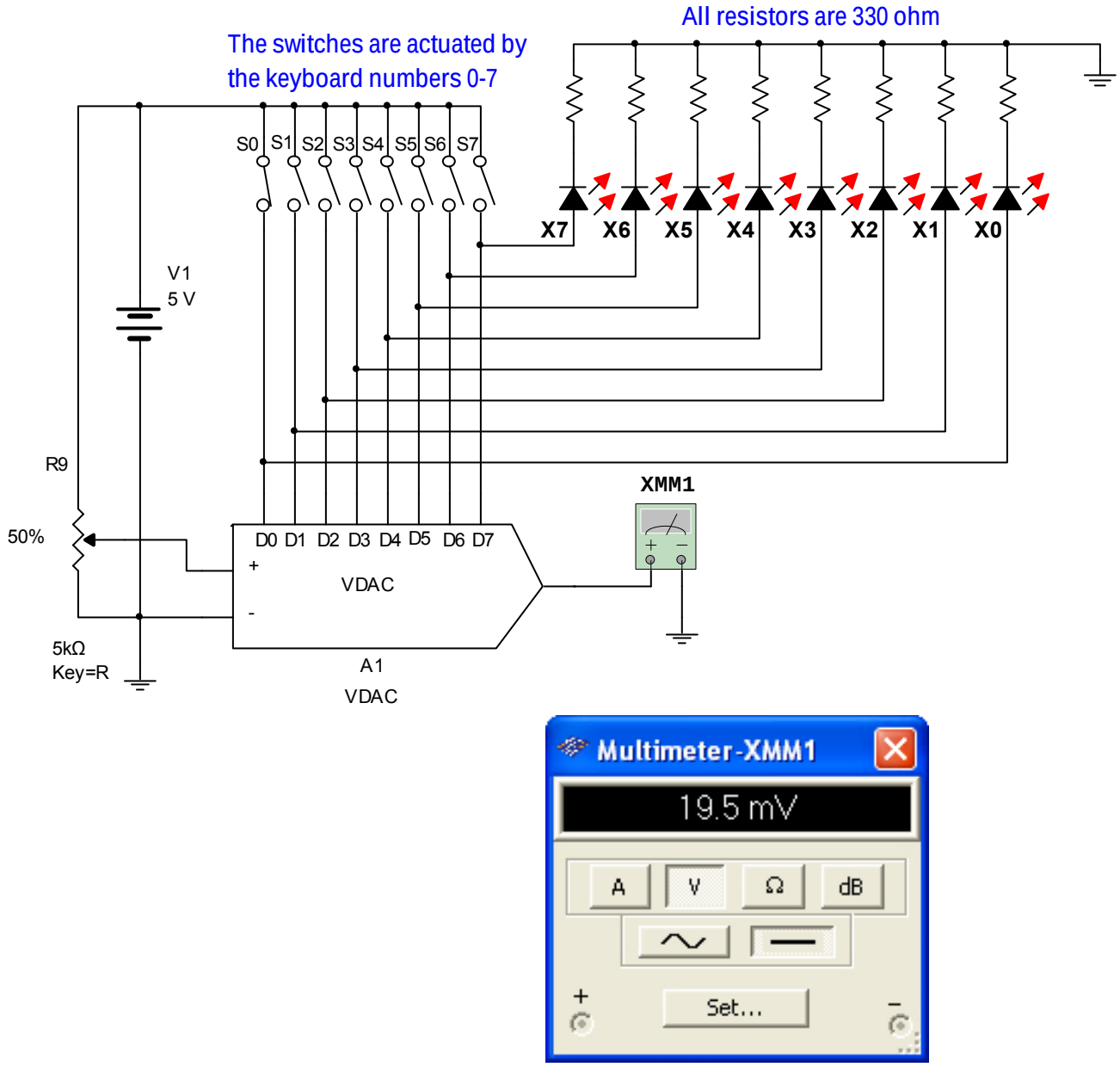



**Figure 3-1. VDAC implementation (Note the value of Vout with the LSB set high, i.e., S0 closed).**

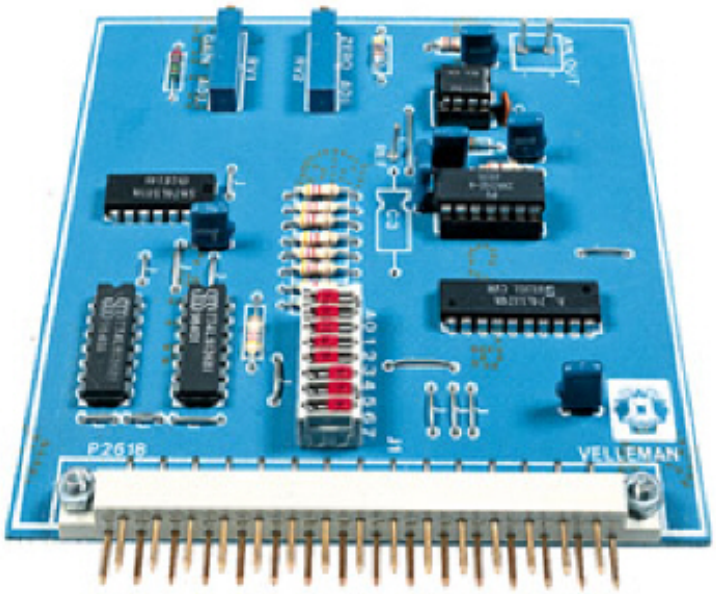
## Problem Statement

Design an 8-bit DAC circuit that is capable of driving a 1K Ω load. Use a reference voltage between 3 $V_{DC}$ to 5 $V_{DC}$. The circuit should be designed such that it provides a positive voltage (unipolar) output.

## Solution

To ensure that we can drive a 1K Ω load without excessively loading the DAC output, the most prudent approach would be to use an IDAC with an op-amp to provide the I-V conversion and load driving capability. Because we desire an unipolar signal output from the DAC both the –Vref input and –Iout output are tied to ground. Refer to figure 3-2 below for a schematic on one solution for this circuit design.

**The Digital-to-Analog Converter Card is intended for use with the Intelligent Motherboard for the RS232 Serial Port Extension System, and is an addition to the other plug-in cards in this range. The D-to-A (Digital-to-Analog) Converter transforms a binary number, presented to it via the system data bus from the Intelligent Motherboard, into an analog voltage output. This output from the D-to-A Converter Card could then be used to control analog equipment, such as the volume of an automated mixing desk or the control of lighting, and even motors and actuators. etc.**




**Digital-to-Analog Converter Card**

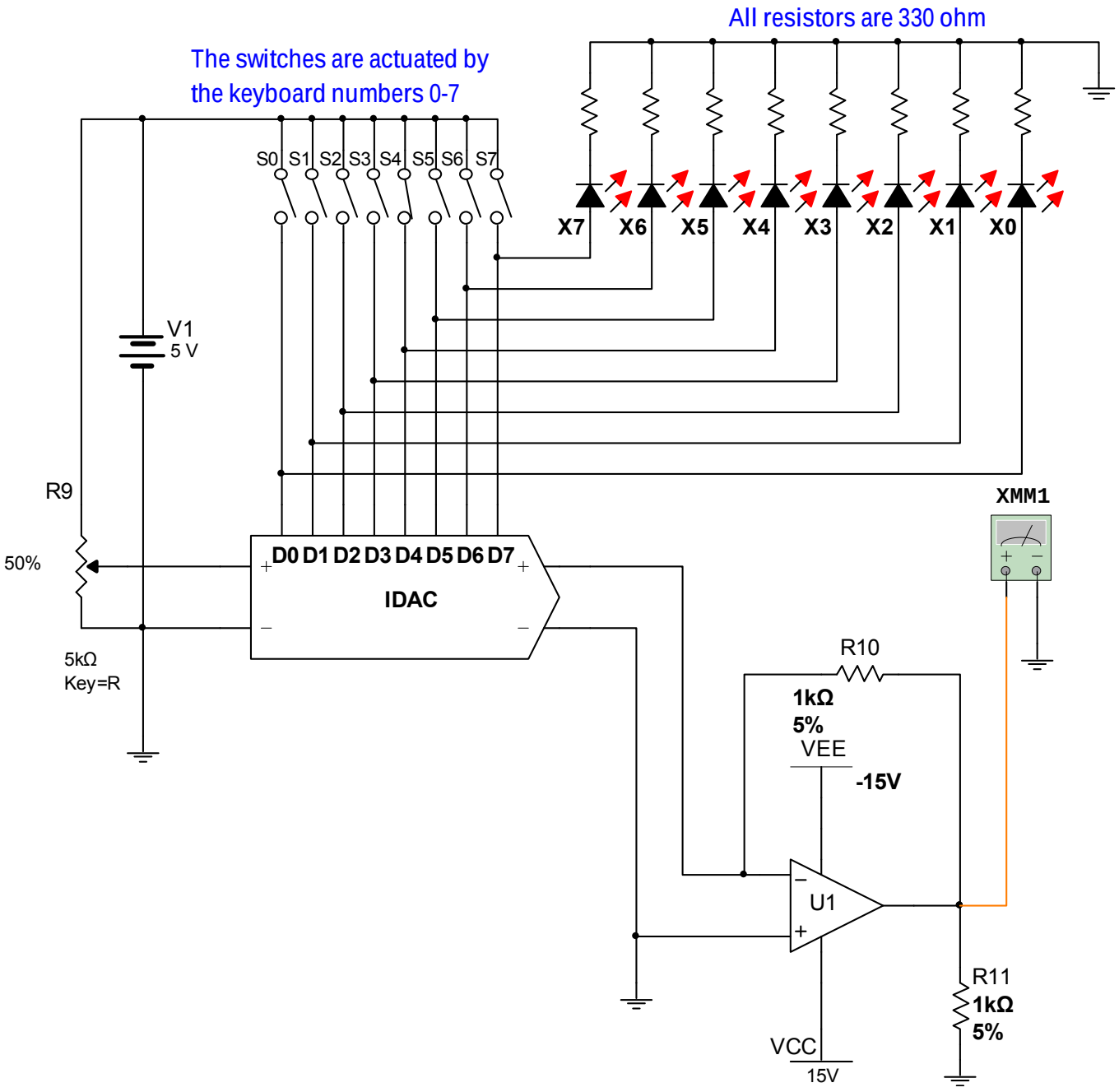



**Figure 3-2 – Example of IDAC design implementation in Multisim with I-V drive capability for 1K Ω load. (note indicator order is MSB (X7) → LSB (X0), left to right ).**

## Circuit analysis

Referring to figure 3-2, we see a reference voltage of 5V (V1) with R9 setting the reference current into the IDAC. Switches S0 – S7 enable setting of the binary value input into the IDAC, with S0 representing the LSB and S7 representing the MSB.

On the output of the IDAC we have a general purpose op-amp (e.g., 741) configured with +/– 15V bipolar voltage supplies. These voltage supplies must be greater than the reference voltage to ensure full 0 – 5V output compliance (+/– 10V or +10V/–5V would have also been adequate).

The op-amp's feedback resistor (R10) is chosen to give the proper scaling of the desired output voltage as a function of the IDAC's output current.

Note that the inverting input of the op-amp is supplied by the +Iout output of the IDAC but the Vout of the op-amp is positive based on the design requirements. One might expect a negative voltage at the output of the 741, however, the +Iout pin of the IDAC actually sinks current rather than sources it. Therefore, the voltage drop across U1's feedback resistor is positive to negative with respect to the op-amp's output. R11 is the load resistance as identified in the circuit design requirements.

### Mathematical analysis of the DAC circuit is as follows:

- When designing a DAC circuit, one of the important things to understand is the **resolution capability of the DAC** being used. As mentioned previously, DAC resolution is a function of both the number of bits being accommodated by the DAC and the Vref being used.

**Equation #1: DAC resolution = Vref / $2^n$**

Where: **Vref = the DAC reference voltage**

**n = # of bits accommodated by the DAC (e.g., 8-bit DAC)**

***Note:*** ***upon observation of equation #1, it is clear that the DAC resolution could be improved by either increasing the number of bits employed or decreasing the reference voltage. However, increasing the number of bits increases cost and conversion time and decreasing the reference voltage makes the conversion more sensitive to errors on noise.***

Therefore, in the circuits of figure 3-1 & 3-2:

**Resolution = 5V / $2^8$ = 19.5 mV**

This means that the smallest analog voltage step size that can be represented by the DAC with a Vref = 5V is 19.5 mV.

- **To determine Vout for any binary value input:**

  **Equation #2: $V_{out} = V_{ref} (N / 2^n)$**

  Where: **Vout is the DAC's output voltage**
  **(after I–V conversion with an IDAC or the Vout of a VDAC)**

  **Vref = the DAC reference voltage**

  **N = is the decimal equivalent to the binary input value**

  **n = # of bits accommodated by the DAC (e.g., 8-bit DAC)**

  Therefore:

  If we close the switch for only the LSB (S0 in figure 3-1, binary value = 00000001) we should see approximately 20 mV at the output, because: 5V (1/256) = 19.5 mV

  With all 8 switches closed (i.e., binary value = 11111111) we should see approximately 5V ($255 / 2^8$) = 4.98V at the output (this is illustrated in figure 3-2)

## Procedure/Tasks Using Multisim

1. Build the circuit indicated in figure 3-2 using Multisim. Choose 1% increment for the potentiometer (R9)

2. Adjust R9 (by pressing the r or SHIFT-r keys) until Vout is approximately equal to 4.98V with S0 – S7 all closed.

3. If R9 is properly adjusted for the correct Iref, Vout should equal 4.98V with S0 – S7 all closed and Vout should equal approximately 0V with S0 – S7 all open.

   ***Note:** Vout will actually be on the order of µV with S0 – S7 open due to offset voltage and bias currents of the op-amp. However, considering the DAC's resolution of 20mV, this Vout value is essentially zero.*

4. Using Multisim and equation #2 complete the following Table 3 –1 of DAC Vout values for given binary input values.

**Table 3 –1**

| S7 – S0 Input (Binary) | Input (Dec) | Input (Hex) | Vout Calculated | Vout Measured |
|---|---|---|---|---|
| 00000000 | | | | |
| 00000001 | | | | |
| 00000010 | | | | |
| 00000100 | | | | |
| 00001000 | | | | |
| 00010000 | | | | |
| 00100000 | | | | |
| 01000000 | | | | |
| 10000000 | | | | |
| 11000000 | | | | |
| 11100000 | | | | |
| 11110000 | | | | |
| 11111111 | | | | |

**Questions**

1. When the binary input value was increasing, did you notice a pattern in the corresponding Vout values? What form did this pattern of Vout values take? Was it linear, or non-linear?

2. Based on the results cited in question 1, explain the reason for the observed Vout value pattern?

3. Replace R10 with a 10 KΩ resistor and note the Vout values for binary inputs of 00000001 and 11111111. How does the R10 resistor value change affect Vout? Why?

4. Set U1's $V_{CC}$ to +5V and $V_{EE}$ to ground and note the Vout values for binary inputs of 00000001 and 11111111. How do these circuit parameter changes affect Vout? Why?

## Procedure/Tasks – Hardwired Circuit

Now, you have tested the circuit to satisfy the problem statement using Multisim, it is time to hardwire the circuit shown in Figure 3-2 with real components. We will use the DAC0800 LCN DIP IC and whose schematic diagram is depicted in Figure 3-3.

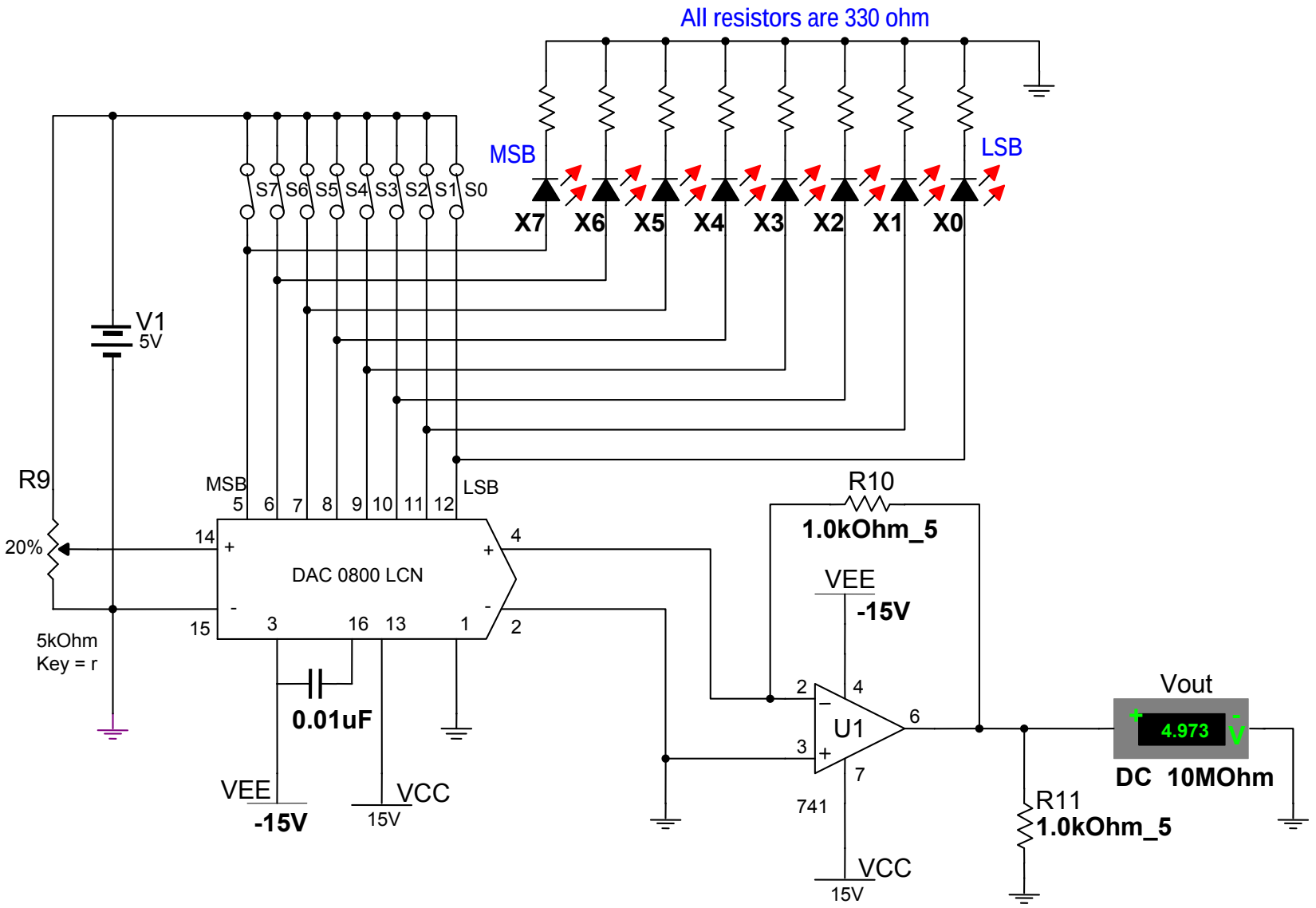



**Figure 3-3. Schematic diagram of an IDAC using the DAC0800LCN DIP IC.**

Follow the same steps as you did in the previous labs, you are going to indicate the list of components and the equipment that you need to complete this task.

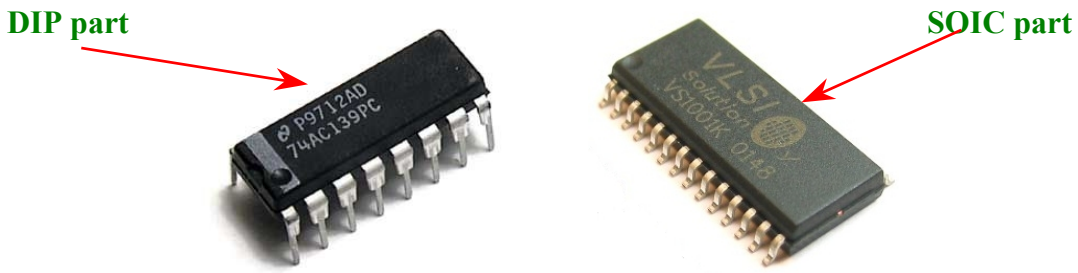
**List of components**

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| **1** | **National DAC0800 (or equivalent)** | **A1** | **1** |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

**Equipment List**

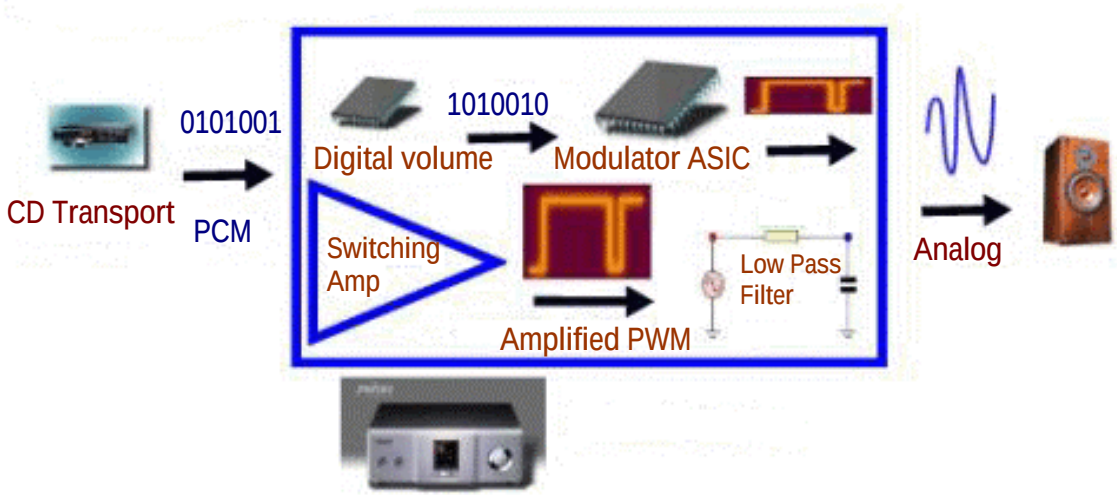
It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin. This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.

- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.
- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.




***DIGITAL AMPLIFICATION***

***The main difference is that digital-to-analog conversion is performed after amplification. The digital amplifier transforms input PCM (pulse code modulation) signal into another digital format called PWM (Pulse Width Modulation), which then amplifies it in the digital domain. The PWM amplification stage is a type of switching circuit, and is therefore not influenced by nonlinearity and transistor noise.***

***The final stage of audio production in the signal path is a simple passive low-pass filter. The low-pass filter transforms PWM digital signal into analog power signal, which can directly drive a loudspeaker.***

## Hardwired Circuit Procedure

1. Build the circuit indicated in figure 3-3 using a breadboard and actual components.

2. Adjust R9 until Vout is approximately equal to 4.98V with S7 – S0 switches all closed.

3. If R9 is properly adjusted for the correct Iref, Vout should equal 4.98V with S7 – S0 switches all closed and Vout should equal approximately 0V with S7 – S0 switches all open.

   ***Note:*** ***Vout will actually be on the order of µV with S7 – S0 switches open due to offset voltage and bias currents of the op-amp. However, considering the DAC's resolution of 20mV, this Vout value is essentially zero.***

4. Using hardwired circuit measurements and equation #2, complete the following Table 3-2 of DAC Vout values for given binary input values.

**Table 3-2**

| S7 – S0 Input (Binary) | Input (Dec) | Input (Hex) | Vout Calculated | Vout Measured |
|---|---|---|---|---|
| 00000000 | | | | |
| 00000001 | | | | |
| 00000010 | | | | |
| 00000100 | | | | |
| 00001000 | | | | |
| 00010000 | | | | |
| 00100000 | | | | |
| 01000000 | | | | |
| 10000000 | | | | |
| 11000000 | | | | |
| 11100000 | | | | |
| 11110000 | | | | |
| 11111111 | | | | |

## Questions

1. Compare the results of Tables 3-1 and 3-2. Are they about the same? Describe any discrepancies and explain what could be the possible causes.

2. Replace R10 with a 10 KΩ resistor and note the Vout values for binary inputs of 00000001 and 11111111. How does the R10 resistor value change affect Vout? Did you obtain the same results in Multisim?

3. Set U1's $V_{CC}$ to +5V and $V_{EE}$ to ground and note the Vout values for binary inputs of 00000001 and 11111111. How do these circuit parameter changes affect Vout? Does the circuit work as expected? How do the results obtained with the hardwired circuit compare to those obtained with Multisim?

## Conclusions

After completing the DAC circuit implementation in both Multisim and hardwiring, what are your conclusions? What similarities do they hardwired circuit and the circuit simulation share? What major differences did you notice between the two?

## Advanced Design

**If you want to earn extra credit, try the following design issue.**

Your supervisor informs you that the department is having problems with an expensive piece of test equipment and she wants you to investigate. You find the piece of equipment in question is a multi-channel, modular data logger that uses multiple data acquisition cards. Manufacturing engineers using the unit say that its 12-bit multi-channel DAC card is unreliable and producing questionable results. Using what you now know about DACs, how would you go about resolving this issue?

### Knowns:

1. The manufacturing engineers complaining about the “bad DAC board” have not taken measurements to verify its alleged performance issues. Their complaints stem from the fact that fixtures apparently function correctly, although the DAC board produces results that imply otherwise.

2. Because the equipment cost is quite expensive, outright replacement of the equipment is out of the question.

3. The data logger repair cycle time is about three months due to a shortage of replacement parts. This situation poses a very important company need: checking and calibrating production test fixtures.

4. Your supervisor does not care how you resolve the problem as long as it is completed in a timely manner that does not cost too much money. (i.e., you cannot purchase new DAC boards)

5. You have the data sheets on the 12–bit DAC data acquisition card. Therefore, you know how to set its Vref levels, and its current output gain settings (which are similar to those in the circuit in figure 3-2).

6. The data logger is used to check fixtures that run on +/– 12V power supplies with digital logic levels of 0 – 5V

7. Required resolution for the test fixture application must be a minimum of 5 mV to meet performance requirements.

8. DAC board measurements show Vout to be equal to 3 mV with a binary input of 000000000001 and 11.99V with a binary input of 111111111111.

9. You must show the manufacturing engineers proof of your findings, i.e., calculations showing the DAC is either functioning properly or not and offer a solution to their problems.

**Unknowns:**

1. How do you confirm whether or not the DAC board is functioning correctly?

2. If the DAC board is malfunctioning, can you design a temporary solution using a 12-bit DAC and associated components, knowing what you did with DACs and this particular application? (**Hint:** refer to digital circuit textbooks and manufacturer's DAC data sheets and application notes for other in insights).

3. If you do not have enough information to design a work around solution for the issue, what other information is required and why is it needed?

# LAB 4 – ANALOG TO DIGITAL CONVERTER (ADC)

## Objectives

1. Learn how to design an ADC circuit that converts analog input signals to corresponding binary output values.

2. Build, test, and troubleshoot an ADC circuit using Multisim.

3. Hardwire the ADC circuit of objective 1 and compare the measurements of the hardwired circuit with the measurements obtained using Multisim.

## Preliminary Information

In many applications, **ADCs and DACs are used together to provide a digital interface to an analog world**. One such example is the entertainment industry. The music industry uses analog to digital conversion (ADC) techniques to record musicians playing in a studio (analog signal source) and this analog signal is encoded into a digital format which is then stored in compact disks (CD) and sold to the consumer. The consumer then takes his/her compact disk and plays it in a CD player.

The CD player uses digital to analog conversion (DAC) techniques to decode the digital signal and reconstitute the original recorded sounds of the musicians into an analog format. This analog signal in the audio band (20 Hz to 20 kHz) is then amplified and used to drive speakers. This same analogy also applies to the film industry, which now stores and distributes its movies on DVD (Digital Video Disk).

Analog to Digital Converters or ADCs are used to convert analog signals into corresponding digital signals. The digital signal generated in this fashion is a binary representation of the original analog signal's amplitude.

At a very fundamental level, ADCs determine the digital representation of an analog input signal by comparing it to a given reference voltage. The resulting digital value represents the proportion between the analog input signal and the reference voltage.

Using an example, if Vin = 2.5V and Vref = 5V for an 8-bit ADC we would expect to see a binary value of 10000000 at the ADC output. Why?

> **Because 2.5 / 5V = 50% and 50% of 28 = 256 = 10000000. We will cover this in more detail in the circuit analysis section below.**

Many ADC techniques are currently employed in industry; however, we will provide a brief overview of only the techniques that are most widely used.

1. **Parallel converter or Flash converter –** uses many comparators connected in parallel, each with a different reference voltage. The outputs of these comparators are then input into a priority encoder. This encoder provides a binary output based on which comparator outputs are high and which are low.

   The number of bits the flash ADC accommodates determines the number of comparators used. e.g., an 8-bit flash ADC would require $2^8 - 1 = 255$ comparators to implement the 8-bits. Therefore, while flash ADCs are the fastest converters, but they are limited in the number of bits they can accommodate.

   Generally, flash converters are limited to 4-bit up through 10-bit digital output resolutions.

2. **Dual-Slope converter –** this type of converter uses an op-amp integrator, comparator, digital control circuits, and digital clock circuits, along with a counter to generate the binary or BCD value from an analog source signal.

   The fundamental idea is to switch the integrator input in a synchronized fashion between the analog signal to be converted and a known reference voltage (a negative voltage). This enables both charging and discharging of the integration capacitor at a constant rate.

   When the integration capacitor charges to a specific level, it triggers the comparator. The comparator then triggers both discharging of the integrator capacitor and a counter that is used to count during the capacitor's discharge cycle.

   The value of the count is the amplitude of the analog input signal. This count value is then converted to a binary or BCD value and made available at the latch outputs of the ADC. Although the part count and circuit complexity for the dual-slope converter is much less than that of the flash converter, the dual-slope converter is considerably slower due to the charge and discharge times required for its integrating capacitor.

3. **Successive-Approximation converter –** this type of converter employs the use of a DAC and a successive approximation register (SAR), along with control logic circuits, latch and output drivers to perform its ADC operations.

   Initially, all binary or BCD bits in the successive approximation register are set to zero. Once the conversion starts, the MSB is set to "1" and the bits (e.g., 10000000 for an 8-bit converter) are then fed, in a parallel fashion, into the DAC input, where the binary or BCD value is converted by the DAC into a reference voltage.

   This reference voltage is then fed into a comparator input where it is compared with the analog input signal. If the analog input signal is greater than the DAC derived reference voltage, the MSB is left as a "1" in the SAR; otherwise, it is set to "0". Then, this new binary value is shifted out to the DAC where a new reference voltage is generated and compared with the analog input signal.

This function essentially provides a means of performing a binary search on the analog input voltage level. This cycle continues until all binary bit values, from the MSB to the LSB, in the SAR have been set accordingly. Finally, the value is shifted out from the SAR into latch and output drivers to be made available outside of the ADC.

Most of the off-the-shelf ADC ICs incorporate the SAR type converter.

## Problem Statement

Design an 8-bit ADC circuit that utilizes LEDs to indicate its binary output value. Use a reference voltage of 2.5V to 5 $V_{DC}$. Design the circuit such that it will continually update its binary output for a changing analog input signal. Design the circuit to accommodate positive voltage (unipolar) operation only.

## Solution

The implementation of the above design requirements is relatively straightforward considering the unipolar operating requirement. To enable the device to operate in a unipolar fashion, the positive Vref must be tied to the 2.5 to 5V reference as described above and negative Vref must be tied to ground.

Additionally, setting the ADC to constantly update based on a changing analog input signal requires an external clock to toggle the ADC's SOC (Start of Conversion) pin. Additionally, the ADC's OE (Output Enable) pin must be pulled high to ensure constant updating of the binary output value.

Refer to figures 4-1 & 4-2 below for a schematic of such implementation.

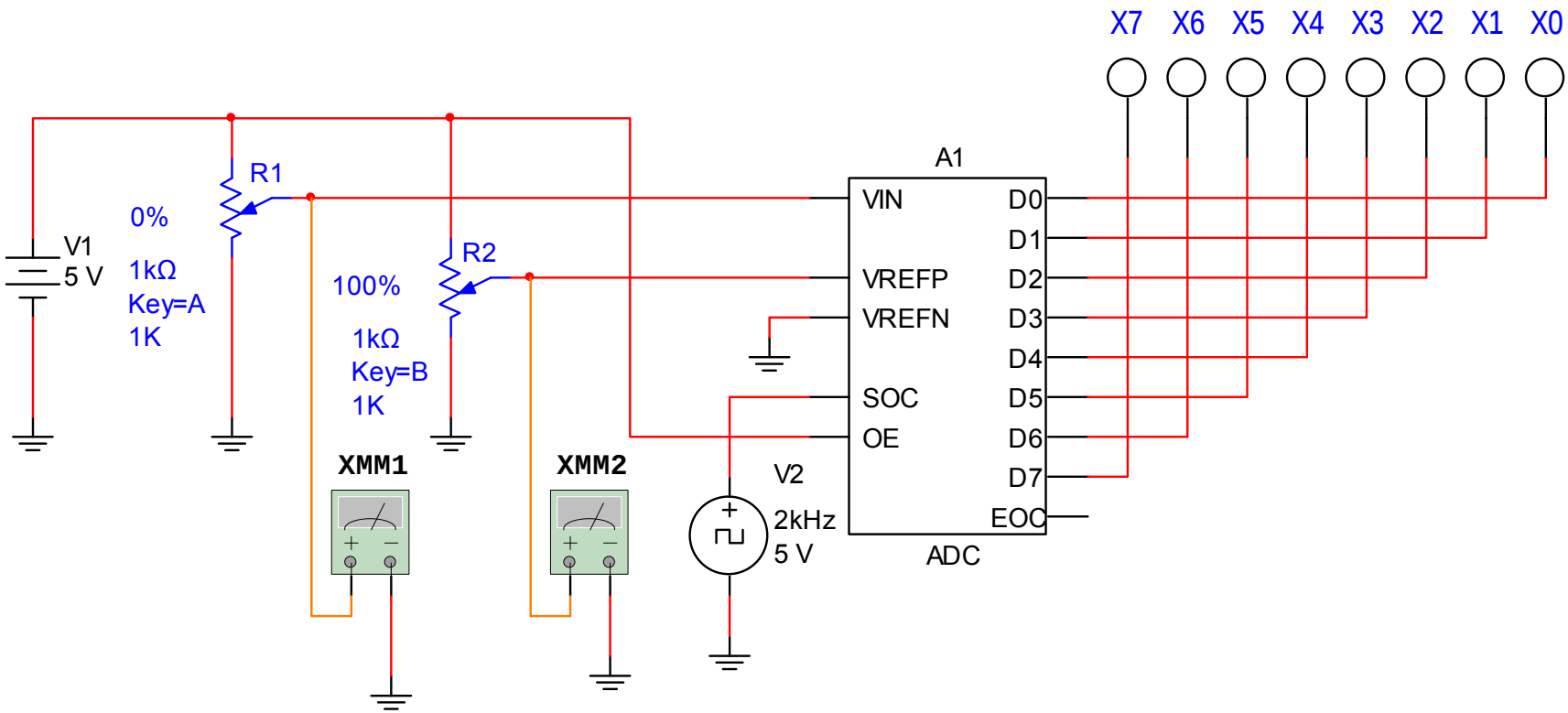



**Figure 4-1. ADC output with the analog input voltage equal to approximately 0V and +Vref = 5V**

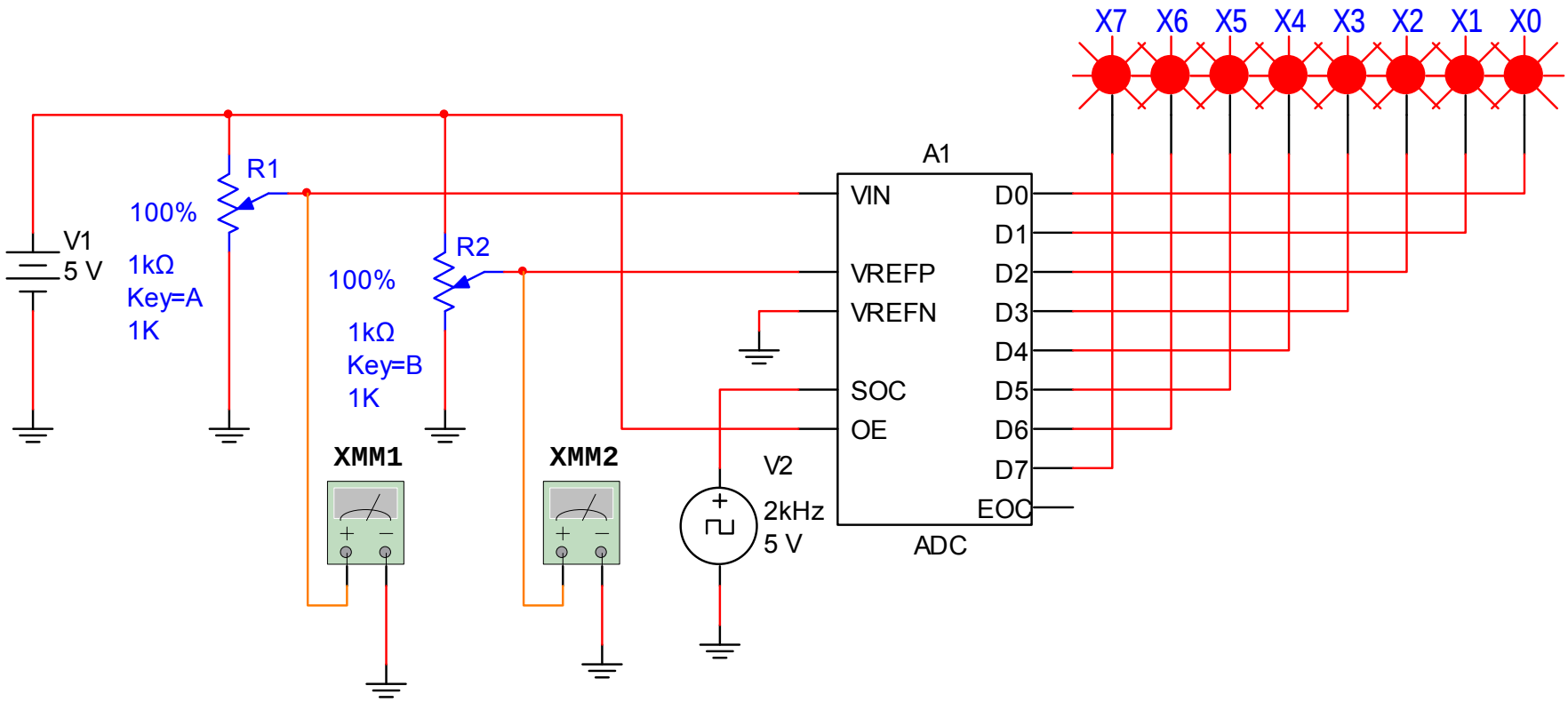



**Figure 4-2. ADC output with the analog input voltage equal to the reference voltage of 5V and +Vref = 5V.**

## Circuit analysis

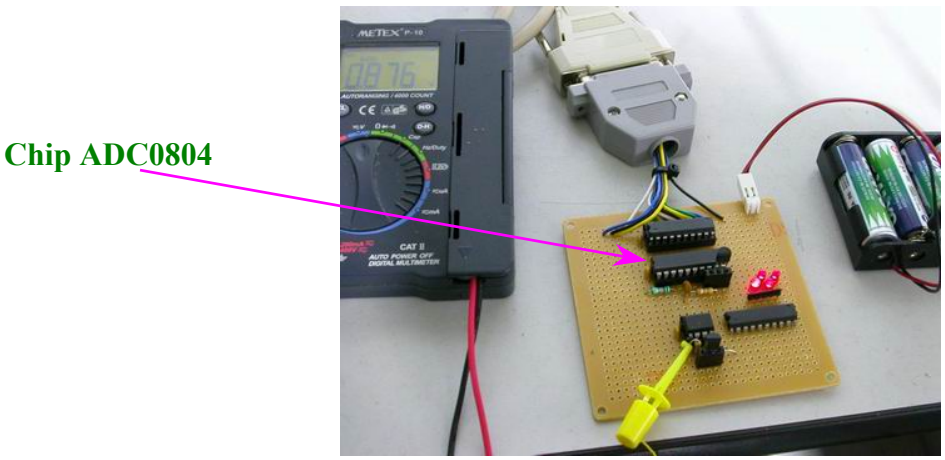
Referring to figures 4-1 & 4-2, we see an adjustable reference voltage available to +Vref (VREFP in the schematic) via voltage supply V1 and potentiometer R2. The varying analog input signal is also derived from V1 using potentiometer R1 to provide signal variation to the VIN input of the ADC.

Additionally, we note –Vref (VREFN) pulled to ground to enable the unipolar output operation. The SOC (Start of Conversion) pin is tied to a square wave signal generator to provide the necessary clock source to ensure continual updating of the ADC.

Note, when designing such a continually updating (i.e., free-running) ADC application, care must be taken to ensure the SOC clock frequency provides enough time to allow the converter to complete the conversion process in its entirety.

Failure to do so will result in erroneous conversion in the binary output values. In this case, a clock frequency of 2 KHz provides a period of 500 µs, which is more than enough time for a worst case ADC conversion cycle **(e.g., the National Semiconductor ADC0804 requires a worst case conversion time of 114 µs).**

The OE pin of the ADC is pulled to ground to ensure the binary outputs are constantly updated as conversions are completed. Data output pins are tied to indicators per the problem definition requirements.



Mathematical analysis of the ADC circuit is as follows:

The following parameters are important when characterizing the performance of an ADC circuit design:

1) An **ADC's % resolution** can be defined in either of the following ways:

**Equation #1: ADC % resolution = [1 / ($2^n$)] x 100%**

**Or**

**Equation #2: ADC % resolution = (Step Size / Vref) x 100%**

2) An **ADC's Step Size** is defined as the smallest possible voltage representation of the ADC or the voltage that is equivalent to its LSB value:

**Equation #3: ADC step size = Vref / ($2^n$)**

Where: **ADC** % resolution represents % of Vref

**Vref** = the ADC reference voltage

**n** = # of bits accommodated by the ADC

**(e.g., 8-bit ADC = 256 values)**

***Note:*** ***Observing equations #1 & 2, it is clear that the ADC resolution could be improved by either increasing the number of bits employed or decreasing the reference voltage. However, increasing number of bits increases cost and conversion time and decreasing the reference voltage makes the conversion more sensitive to errors due to noise.***

Therefore, in the circuits of figure 4-1 & 4-2:

**Step size** = 5V / ($2^8$) = 19.53 mV

**% Resolution** = [19.53mV / 5V] x 100% = 0.3906%

**Or**

**% Resolution** = [1 / ($2^8$)] x 100% = 0.3906%

This means that the smallest analog voltage step size that can be represented by the ADC with a Vref = 5V is 19.53 mV and that this step size is 0.3906% of Vref (or Full Scale Voltage)

**To determine the binary value for a given Vin:**

**Equation #4: ADC decimal number output = INT [(Vin / Vref) $2^n$]**

Then convert the ADC decimal number to binary to obtain binary equivalent of Vin

Where: **Vin** is the ADC analog input voltage to be converted

**Vref** = the ADC reference voltage

**n** = # of bits accommodated by the ADC (e.g., 8-bit ADC)

**INT** indicates taking the integer of the calculated value by truncating the calculated quantity in [ ]

Referring to figure 4-1:

ADC decimal number output = INT [ (5µV / 5V) $2^8$] = INT [0.000256] = $0_{10}$

$0_{10} = 00000000_2$

Referring to figure 4-2:

ADC decimal number output = INT [ (5V / 5V) $2^8$] = INT [256] = $256_{10}$

$256_{10} = 1 \times 2^8 + 0 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$

$256_{10} = 100000000_2$

However, because the ADC is limited to a maximum value represented by 8-bits or $2^8$–1, its full-scale output will show the following:

$255_{10} = 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 11111111_2$

This value is represented in the circuit simulation in figure 5-2

As an additional example, assume Vin = 2.5V, then we would expect to see the following at the ADC output:

ADC decimal number output = INT [ (2.5V / 5V) $2^8$] = INT [128] = $128_{10}$

$256_{10} = 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 10000000_2$

## Procedure/Tasks Using Multisim

1. Build the circuit indicated in figures 4-1 & 4-2 using Multisim. Choose 0.1% increment for potentiometers R1 & R2 and set adjustment keys as shown.
2. Adjust R1 (by pressing the A or SHIFT-A keys) until VIN (Vin) is equal to 5V
3. Adjust R2 (by pressing the B or SHIFT-B keys) until VREFP (Vref) is equal to 5V
4. Using Multisim and equation #4 complete the following table of ADC binary output values for given voltage input values.

### Table 4-1

| Vin (R1 setting as indicated by U1) | Binary Output Calculated | Binary Output Indicated (X7 – X0) |
|---|---|---|
| 0V | | |
| 0.020V | | |
| 0.040V | | |
| 0.080V | | |
| 0.160V | | |
| 0.320V | | |
| 0.640V | | |
| 1.280V | | |
| 2.500V | | |
| 3.750V | | |
| 4.375V | | |
| 4.680V | | |
| 5.000V | | |

## Questions

1. As the Vin value was increasing, did you notice a pattern in the corresponding binary output values? What form did this pattern of binary values take? Was it linear, or non-linear?

2. Based on the results cited in question 1, explain the reason for the observed binary output value pattern.

3. Set R2 such that VREFP is set to 2.500V as indicated on U2. If you were to vary Vin over the same voltage range as in Table 4-1, what would the binary output values indicate for each Vin value? Why?

4. If Vin exceeded Vref in a real ADC IC, would it actually damage the IC? If not, what would the binary output values be for a given Vin that was larger than Vref? Refer to the data sheet of the National Semiconductor ADC0804 IC and check the absolute maximum ratings & electrical characteristics section to reinforce your understanding.

## Procedure/Tasks – Hardwired Circuit

Now that you have tested the circuit to satisfy the problem statement using Multisim, it is time to hardwire the circuit depicted in Figures 4-1 & 4-2 using real components. As you did in the previous labs, you are going to indicate the list of components and the equipment that you need to complete this task. Refer to figure 4-3 below for a schematic of the hardwired circuit.

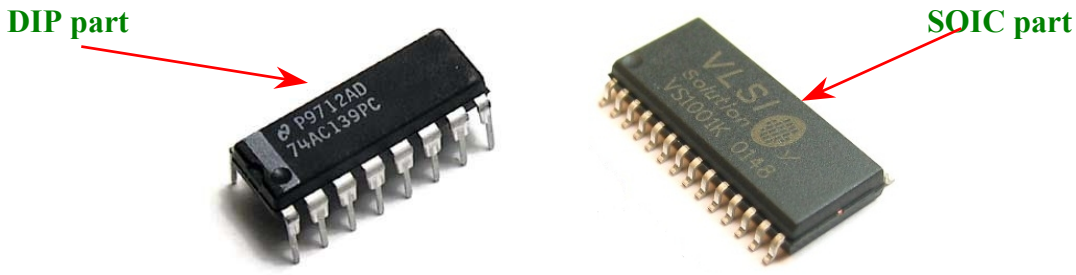
### List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| **1** | **National ADC0804 (or equivalent)** | **A1** | **1** |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

**Equipment List**

It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin. This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.

- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.
- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.

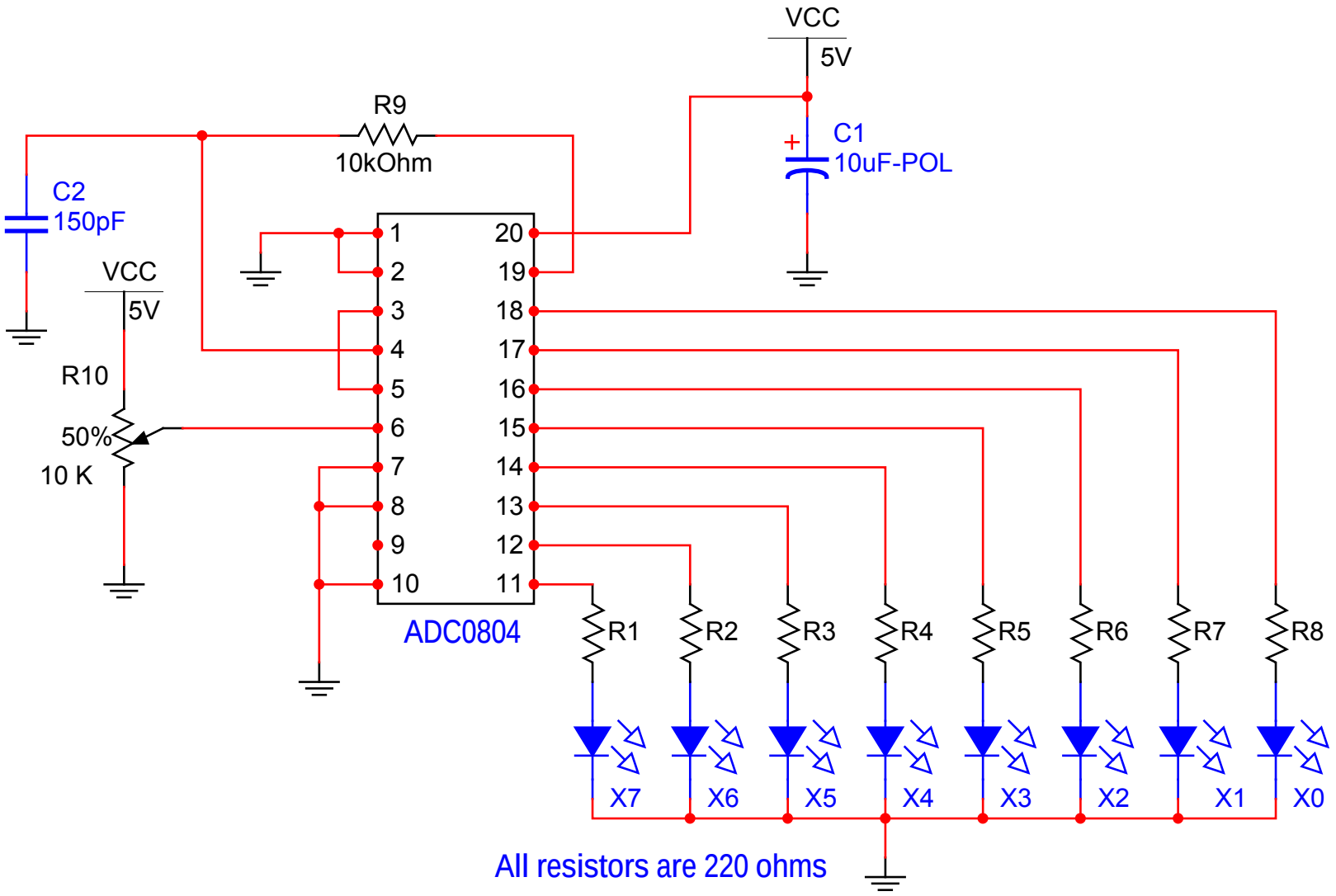



**Figure 4-3. ADC0804**

## Hardwired Circuit Procedure

1. Build the circuit indicated in figure 4-3 using a breadboard and actual components. Refer to the National ADC0804 data sheet for pin function descriptions.

2. Adjust R10 until Vin (pin 6) is equal to 0V.

3. Using equation #4 and hardwired circuit measurements complete the following table of ADC binary output values for given voltage input values.

**Table 4-2**

| Vin (R10 setting) | Binary Output Calculated | Binary Output Indicated X7 → X0 |
|---|---|---|
| 0V | | |
| 0.020V | | |
| 0.040V | | |
| 0.080V | | |
| 0.160V | | |
| 0.320V | | |
| 0.640V | | |
| 1.280V | | |
| 2.500V | | |
| 3.750V | | |
| 4.375V | | |
| 4.680V | | |
| 5.000V | | |

## Questions

1. Compare the results of Tables 4-1 and 4-2. Are they about the same? Describe any discrepancies and explain what could be the possible causes.

2. Add another 10K potentiometer to the circuit of figure 4-3 such that the wiper is connected to pin 9 (Vref / 2). Connect the potentiometer's remaining pins to $V_{CC}$ & ground. The new potentiometer should now be connected in a manner similar to R2 in figures 4-1 & 4-2. Adjust the potentiometer such that 1.25V is applied to pin 9 of the ADC0804. Repeat the measurements performed for Table 4-2. How do these results compare the results in Table 4-1, question #3 in the Multisim section?

3. Review ADC literature (textbooks, data sheets, applications notes, etc.), then name and describe three kinds of errors often associated with ADC operations.

## Conclusions

After completing the ADC circuit implementation in both Multisim and hardwiring, what are your conclusions? What similarities do the hardwired circuit and the circuit simulation share? What major differences did you notice between the two?

## Advanced Design

### If you want to earn extra credit, try the following design issue.

Reviewing the extra credit problem from Lab 3 – Digital to Analog Converters, we had the following scenario:

Your supervisor informs you that the department is having problems with an expensive piece of test equipment and she wants you to investigate. You find the piece of equipment in question is a multi-channel, modular data logger that uses multiple data acquisition cards. Manufacturing engineers said that its 12-bit multi-channel DAC card was unreliable and produced questionable results. Using what you now know about DACs, how would you go about resolving this issue?

### Knowns:

1. The manufacturing engineers complaining about the "bad DAC board" have not taken measurements to verify its alleged performance issues. Their complaints stem from the fact that fixtures apparently function correctly, although the DAC board produces results that imply otherwise.
2. Because the equipment cost is quite expensive, the instant replacement of the equipment is out of the question.
3. The data logger repair cycle time is about three months due to a shortage of replacement parts and it fills a very important function: checking and calibrating production test fixtures.
4. Your supervisor does not care how you resolve the problem as long as it is completed in a timely manner that does not cost too much. (i.e., you cannot purchase new DAC boards)
5. You have the data sheets on the 12-bit DAC data acquisition card. Therefore, you know how to set its Vref levels, and its current output gain settings (which are similar to those in the circuit in figure 4-2).
6. The data logger is used to check fixtures that run on +/– 12V power supplies with digital logic levels of 0 – 5V
7. Required resolution for the test fixture application must be a minimum of 5 mV to meet performance requirements.
8. DAC board measurements show Vout to be equal to 3 mV with a binary input of 000000000001 (12 bits) and 11.99V with a binary input of 111111111111 (12 bits).
9. You must show the manufacturing engineers proof of your findings, i.e., calculations showing the DAC is either functioning properly or not and offer a solution to their problem.

**Unknowns:**

1. How do you confirm whether or not the DAC board is functioning correctly?
2. If the DAC board is malfunctioning, can you design a temporary solution using a 12-bit DAC and associated components, knowing what you do about DACs and this particular application? (**Hint:** refer to digital circuit textbooks and manufacturer's DAC data sheets and application notes for other insights.)
3. If you do not have enough information to design a work around solution for the issue, what other information is required and why is it needed?

Based on what you have learned from this ADC lab, how could you use an ADC to solve this DAC issue? Draw a schematic of any proposed circuit design solution you might have in mind and describe how the circuit functions. Explain how your circuit design would solve the problem.

# LAB 5 – TEMPERATURE SENSOR

## Objectives

1. Build, test, and troubleshoot a temperature sensor circuit using Multisim.
2. Design a temperature sensor circuit that detects temperatures higher than the specified value.
3. Hardwire the temperature sensor circuit of objective 1 and compare the measurements of the hardwired circuit with the measurements obtained in Multisim.

## Preliminary Information

Sensor circuits, in some applications, may use a comparator circuit. As its name indicates, a comparator compares the voltage applied to one input of the op-amp with a known voltage or reference voltage ($V_{REF}$) applied to the other input terminal.

In its simpler form, a comparator is an open loop configured op-amp, with two analog input terminals and a digital output. The output can be a positive or negative saturation voltage depending on the polarity and how big the applied input voltage is. In practice, the op-amp comparator is usually used with some form of positive feedback with hysteresis to ensure proper output signal saturation. However, for the purposes of this lab we will rely on the more simplified open loop configuration, but you should be aware of the uses of positive feedback and hysteresis in actual industry applications.

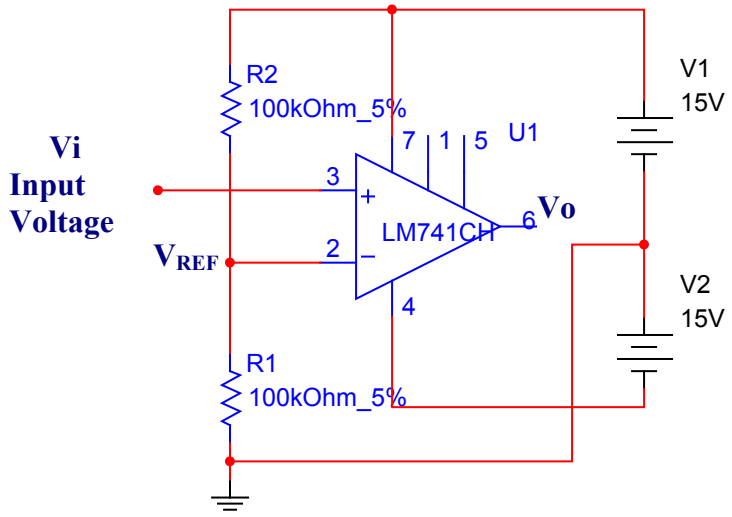



**Figure 5-1. Op-Amp comparator with $V_{REF}$ in the inverting input and dual power supply configuration**

In the circuit of figure 5-1, we observe that the reference voltage ($V_{REF}$) is the voltage drop across resistor R1. Resistors R1 and R2 are in series with the power source V1 (15V) and as such they form a voltage divider.

Since the two resistors are equal (100 kΩ each), the reference voltage will be half of 15V or 7.5V.

In the circuit of figure 5-1, you can verify that:

**When Vi > $V_{REF}$ ➔ Vo = + Vsat = + 13 V**

**When Vi < $V_{REF}$ ➔ Vo = – Vsat = – 13 V**

**Note: The condition of Vi = $V_{REF}$ is not stable.**

Figure 5-2 shows the op-amp comparator with the reference voltage present in the non-inverting input and the input voltage is applied to the inverting input.

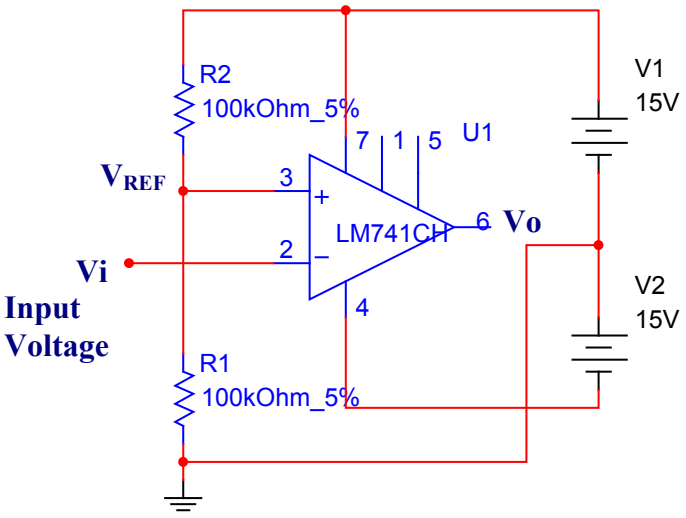



**Figure 5-2. Op-Amp comparator with $V_{REF}$ in the non-inverting input and dual power supply configuration**

In the circuit of figure 5-2, the results are the opposite of the comparator with $V_{REF}$ in the inverting input of figure 5-1; this means:

**When Vi > $V_{REF}$ ➔ Vo = – Vsat = – 13 V**

**When Vi < $V_{REF}$ ➔ Vo = + Vsat = + 13 V**

## Establishment of the Reference Voltage

To establish VREF, we only need to modify the values of R1 and R2 of the op-amp comparator, applying the rule of the voltage divider as follows:

$$V_{REF} = \frac{V1(R1)}{R1 + R2} \qquad (1)$$

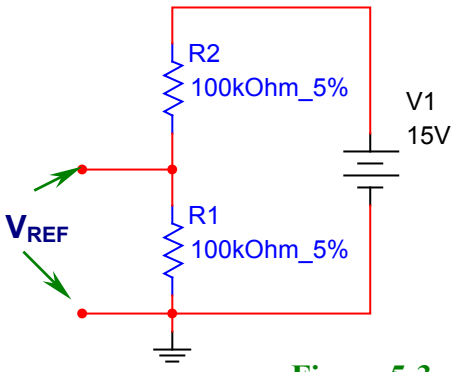



**Figure 5-3**

**To apply the rule of the voltage divider (equation 1) we choose the value of one resistor and we calculate the value of the other.**

### Example:

Calculate the values of R1 and R2 for a reference voltage equal to 3.5 V. The source voltage is 15V.

### Solution.

Using the circuit of figure 5-3 and assuming R1 = 10KΩ, we find the value of R2 in equation 1.

$$V_{REF} = \frac{V1(R1)}{R1 + R2} \qquad (1)$$

$$R1 + R2 = \frac{V1(R1)}{V_{REF}} \quad \Rightarrow \quad R2 = \frac{V1(R1)}{V_{REF}} - R1$$

Replacing values

$$R2 = \frac{15V(10K)}{3.5V} - 10K = 32.857\ K\Omega$$

Then R2 = 33KΩ (commercial value).

**To obtain an exact reference voltage, replace resistor R2 by a potentiometer and adjust it until you obtain the desired voltage.**

## Op-amp Comparator with Single Source

When a single source is used, the output voltage will be zero or positive as shown in Figure 3-4.

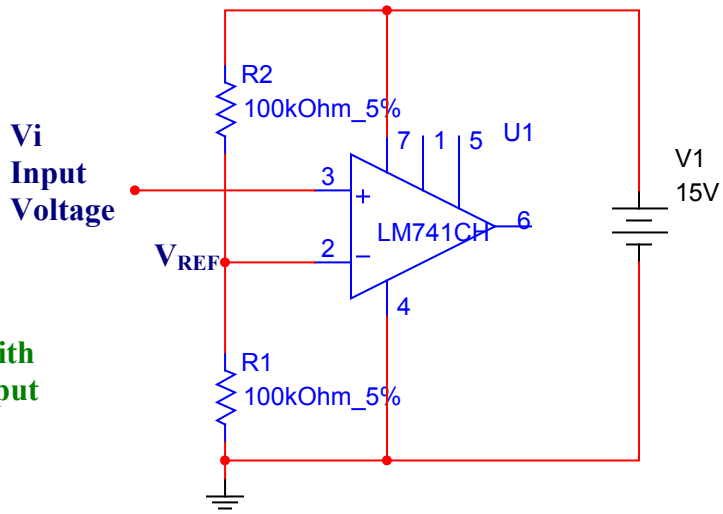



**Figure 5-4.**
**Op-amp Comparator with $V_{REF}$ in the inverting input and single source.**

Since $V_{REF}$ is connected to the inverting input:

**When Vi > $V_{REF}$ ➔ Vo = + Vsat = + 13 V**

**When Vi < $V_{REF}$ ➔ Vo = 0 V**

On the other hand, if $V_{REF}$ is connected to the non-inverting input:

**When Vi > $V_{REF}$ ➔ Vo = 0 V**

**When Vi < $V_{REF}$ ➔ Vo = + Vsat**

## Problem Statement

A sensor provides temperature sensitivity of 200 µV/°C. Design a circuit that activates an alarm when the temperature reaches 300 °C. Use a single 10 V supply, if possible. The alarm could be any type, visual- or sound-based.

## Solution

We need to determine the reference voltage, Vref. According to the problem statement, the sensor provides 200 µV/°C and the temperature that we need to detect is 300 °C. Thus,

$$Vref = 300\ °C * 200\ \mu V/°C = 60\ mV$$

With this information we can design the respective circuit that is in Figure 5-5.

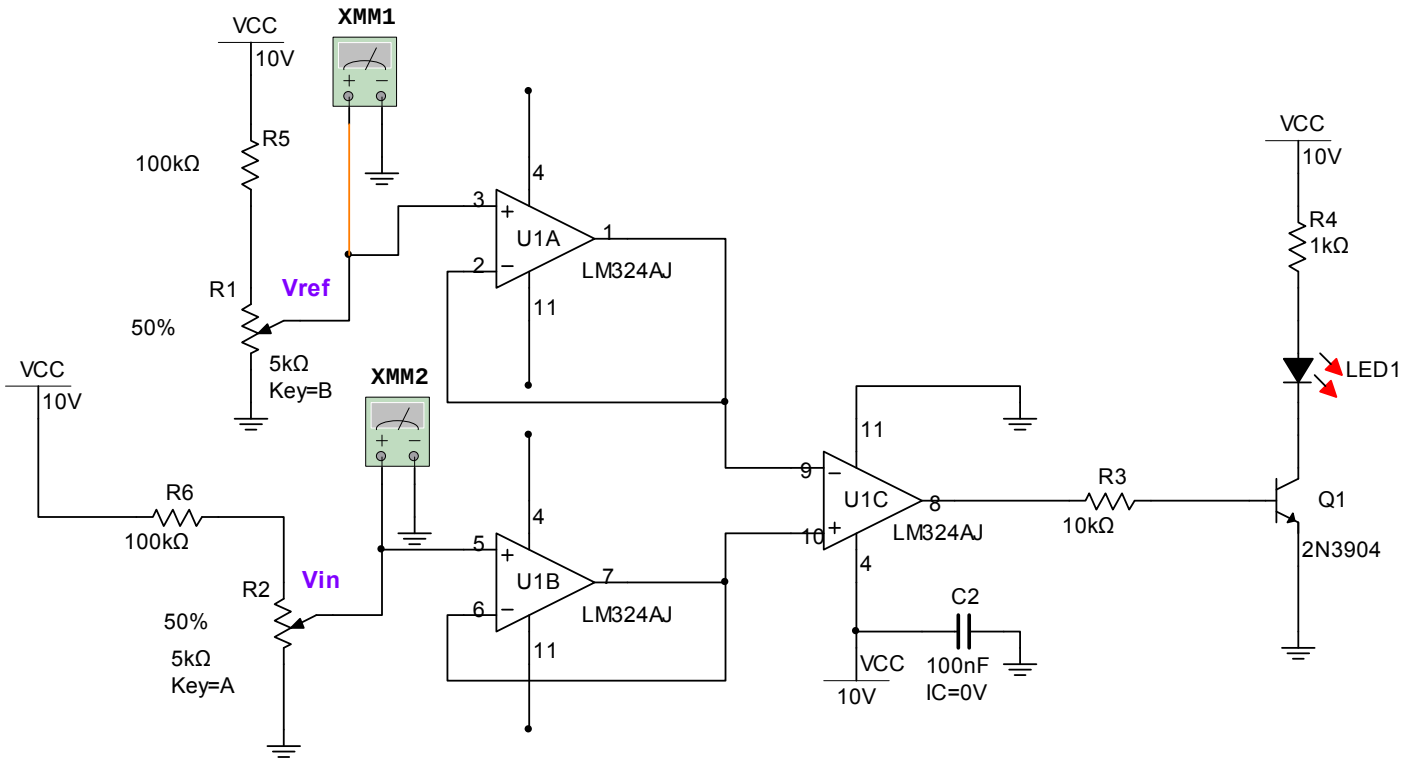



**Figure 5-5. Temperature sensor**

## Circuit analysis

To simplify the design we are going to use the quad op-amp LM324AJ. Notice that we are using three sections out of the four that this IC has. Each section has its own power supply terminals (pins 4 and 11); however, we need to connect the power supplies to one section only as shown in U1C in Figure 5-5. At this point you should be quite familiar with some portions of the temperature sensor depicted in Figure 5-5. It is appropriate then that you are able to answer the following questions.

**Question 1.** What type of configuration are U1A and U1B?

**Question 2.** What is the purpose of U1A and U1B?

**Question 3.** Is U1C an inverting or a non-inverting comparator? Why?

**Question 4**. What is the purpose of capacitor C2?

Vref is obtained by varying potentiometer R1. Potentiometer R2 is used to simulate the voltage that would be supplied by a temperature sensor. This voltage is proportional to the temperature being measured by the temperature sensor

Transistor Q1 and its associated components provide the current that will activate the LED. The LED can be activated directly from the U1C op-amp; however, it is a good design practice to isolate the load from the op-amp for stability purposes. In this circuit we need a single source

power supply because there is no need to have negative voltages at the output. We are using an NPN transistor so we just need positive voltage at the base.

## Procedure/Tasks Using Multisim

1. Build the circuit indicated in Figure 5-5 using Multisim. Choose 2% increment for the potentiometers

2. Adjust R1 (by pressing the B or SHIFT-B keys) until Vref is approximately equal 60 mV.

3. Adjust R2 (by pressing the A or SHIFT-A keys) until the input voltage (Vin) is approximately equal to 60 mV. When this happens, the LED should be ON. Is this correct?

4. Leaving Vref at approximately 60 mV, adjust Vin (by adjusting potentiometer R2) to the values indicated on Table 5-1 and record if the LED is ON or OFF.

**Table 5-1. LED status versus input voltage using Multisim**

| Vref (mV) | Vin (mV) | LED |
|---|---|---|
| 60 | 20 | |
| 60 | 40 | |
| 60 | 60 | |
| 60 | 80 | |
| 60 | 120 | |
| 60 | 140 | |

## Questions

1. Under what conditions the LED is ON or OFF?

2. If Vref = 100 mV and Vin = 90 mV, is the LED ON or OFF? Why?

3. Replace the Q1 transistor by a PNP transistor such as the 2N3906 using Multisim. Test your answer and the transistor and its associated components.

**In most process applications the temperature sensor is inserted into a thermowell or protection tube. This protects the sensor from its environment and facilitates easy removal and replacement. These assemblies generally consist of a head, nipple-union-nipple and thermowell. Smart industrial thermocouples and RTDs are available in virtually any calibration and resistance temperature coefficient.**

## Procedure/Tasks – Hardwired Circuit

Now that you have tested the circuit to satisfy the problem statement using Multisim, it's time to hardwire the circuit depicted in Figure 5-5 using real components. However, as you did in Labs 1 and 2, before continuing you are going to generate the list of components and the equipment that you need to complete this task.

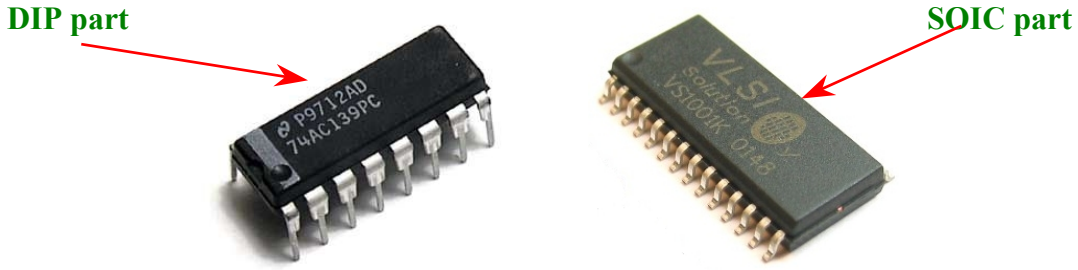
### List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

**Equipment**

It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin. This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.
- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal

path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.

- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.

## Hardwired Circuit Procedure

1. Adjust R1 potentiometer until Vref is approximately equal to 60 mV.

2. Adjust R2 potentiometer until the input voltage (Vin) is approximately equal to 60 mV. What is the condition of the LED?

3. Leaving Vref at approximately 57 mV, adjust Vin (by adjusting potentiometer R2) to the values indicated on Table 5-2 and record if the LED is ON or OFF.

**Table 5-2. LED status versus input voltage in the hardwire circuit**

| Vref (mV) | Vin (mV) | LED |
|---|---|---|
| 60 | 20 | |
| 60 | 40 | |
| 60 | 60 | |
| 60 | 80 | |
| 60 | 120 | |
| 60 | 140 | |

## Questions

1. Compare the results of Tables 5-1 and 5-2. Are they about the same? Describe any discrepancies and explain what could be the possible causes.

2. If Vref = 100 mV and Vin = 90 mV, is the LED ON or OFF? Did you get the same result in Multisim?

3. Replace the Q1 transistor by a PNP transistor such as the 2N3906 as you did using Multisim. Does the circuit work as expected?

## Conclusions

After completing the temperature sensor circuit both in Multisim and hardwire, what are your conclusions?

## Advanced Design

### If you want to earn extra credit, try the following design.

You work for a product manufacturing company that uses a JIT (Just-In-Time) inventory system. One of the workstations requires the use of 15 ball bearings during its assembly procedure for each product. Because these ball bearings are small and difficult to handle, the company has built an automated ball bearing dispenser machine for operators to use. This machine works great, except for the fact that it is a completely enclosed container (to hold the ball bearing inventory and keep it clean). As a result, there is no way to see when the ball bearing inventory is low. Because ball bearing shortages have stopped the line numerous times, your supervisor has tasked you to design a level sensing circuit that sets off an alarm when the ball bearing inventory gets low.

### Knowns:

1. The environment is electrically noisy, due to the large number of automated assembly machines being used.

2. Your supervisor does not care how you implement the design, as long as it can reliably detect when the level of ball bearings has gotten below a desired inventory level.

3. Your supervisor wants this detection implementation to be relatively simple so that it can be placed to detect different levels of ball bearing inventory at different workstations.

### Unknowns:

1. What kind of sensor do we use to detect the ball bearing level?

2. What kind of circuit do we need to detect the sensor output and trip an alarm? (**Hint:** look at the temperature sensing circuit design in figure 5-5. How should this design be modified for this application?)

3. What kind of components do we need to implement this circuit design?

4. How do we deal with the electrically noisy manufacturing environment and how does it affect our circuit design? (**Hint:** look at the temperature sensing circuit design in figure 5-5. How should this design be modified for this application?)

5. Who thought it was a good idea to design a product that needed 15 ball bearings, and why? (This is a rhetorical question).

# LAB 6 – SCR CIRCUIT

## Objectives

1. Build, test, and troubleshoot an SCR circuit using Multisim.
2. Hardwire the SCR circuit of objective 1 and compare the measurements of the hardwired circuit with the measurements obtained using Multisim.

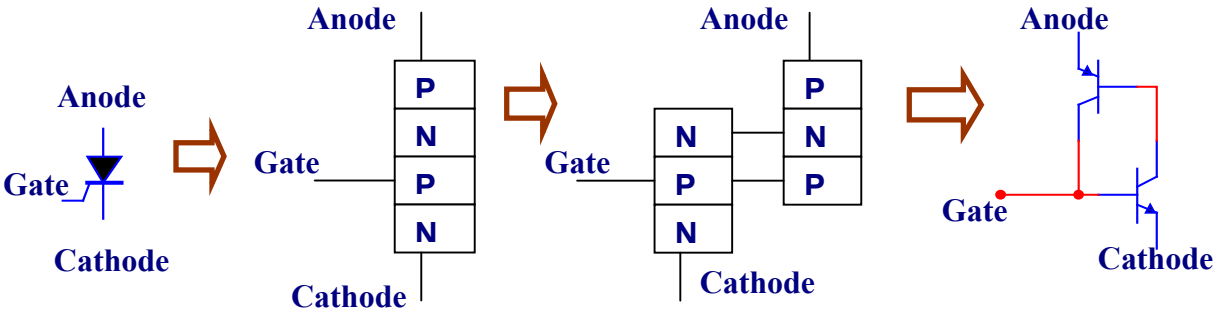
## Preliminary Information

The Thyristor is a device of four layers (PNPN). There are several devices of four layer construction; in this case we will discuss three of them: the SCR, the TRIAC and the DIAC.

## Silicon Controlled Rectifier (SCR).

The SCR is used to control high power because its junctions can be manufactured to pass current in the hundreds of amperes and to withstand high reverse bias of several hundred volts.

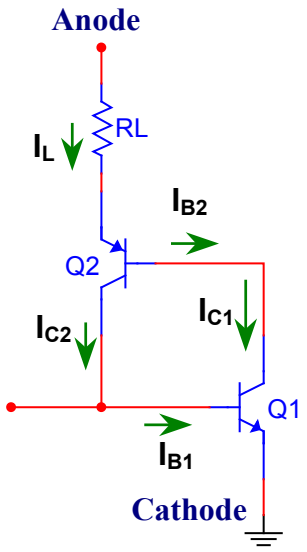
The combination of high reverse voltage and high current handling is impossible for a power transistor. An SCR can be considered as two interconnected transistors as follows:




SCR Symbol | Equivalence with four layers | Equivalence with two transistors | Equivalence with two transistors

For current analysis of an SCR we use a discrete transistor circuit model. This model provides a view of an SCR with its equivalent transistor circuit connected to a load $R_L$. Using this circuit we will calculate the value of $I_L$

From the figure we have:




**SCR - equivalence with two transistors**

$$I_L = I_{C2} + I_{B2} \quad (1)$$

$$I_{B2} = I_{C1} \quad (2)$$

$$I_{B1} = I_{C2} \quad (3)$$

By definition:

$$I_{C1} = h_{FE1}(I_{B1}) \quad (4)$$

$$I_{C2} = h_{FE2}\, I_{C1} \quad (5)$$

Substituting $I_{C1}$ in equation 4 with equation 2:

$$I_{B2} = h_{FE1}(I_{B1}) \quad (6)$$

Substituting $I_{C1}$ & $I_{B2}$ in equation 1 with equations 5 and 6:

$$I_L = h_{FE2}\, I_{B2} + h_{FE1}\, I_{B1} \quad (7)$$

Substituting $I_{B2}$ in equation 7 with equation 6:

$$I_L = h_{FE2}\, h_{FE1}\, I_{B1} + h_{FE1}\, I_{B1} \quad (8)$$

Factoring, we have:

$$I_L = h_{FE1}\, I_{B1}\, (h_{FE2} + 1) \quad (9)$$

Considering that:

$$h_{FE2} >> 1 \quad \text{then} \quad (h_{FE2} + 1) \cong h_{FE2} \quad (10)$$


**Formas físicas del SCR.**

Substituting $(h_{FE2} + 1)$ in equation 9 with equation 10:

$$I_L \cong h_{FE1}\, h_{FE2}\, I_{B1} \quad (11)$$

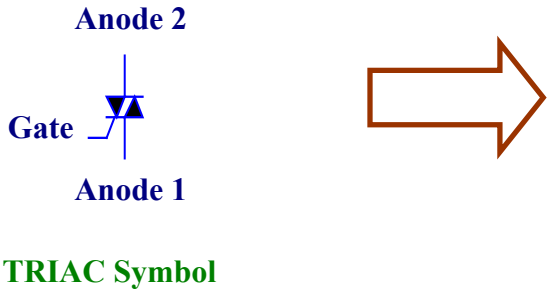
According to equation 11, a small $I_{B1}$ current applied to the SCR's gate will produce a big current, $I_L$.

The SCR remains off until an external current is applied at the gate. Once it triggered, the SCR will remain on even if we remove the signal at the gate. The SCR will switch off only when the current it passes falls below a value called the holding current or the voltage applied between the anode and the cathode is removed or reversed in polarity.

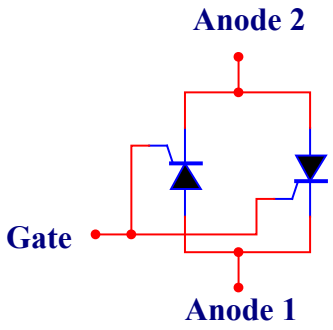
The SCR is usually used as a high power diode rectifier that can be controlled by a relatively small signal at its gate. This implementation is used to rectify AC, thus creating a pulsed DC current, and control the availability of this DC current.

## THE TRIAC

It can be considered as two SCRs with a single gate.



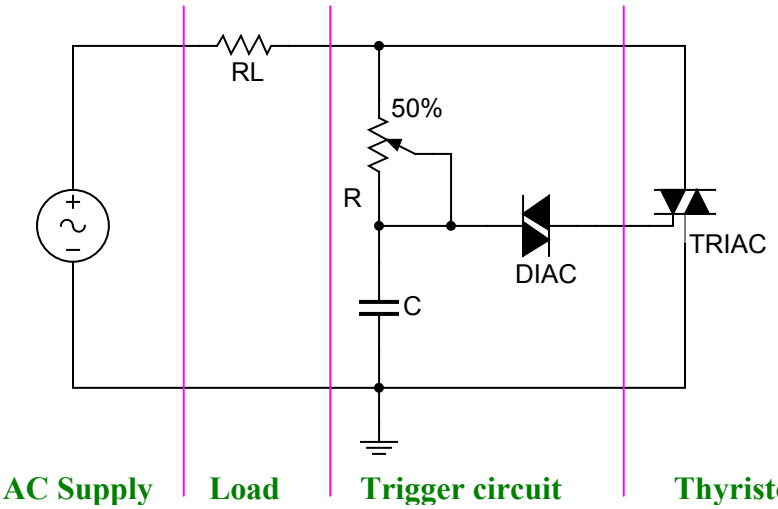

TRIAC Symbol



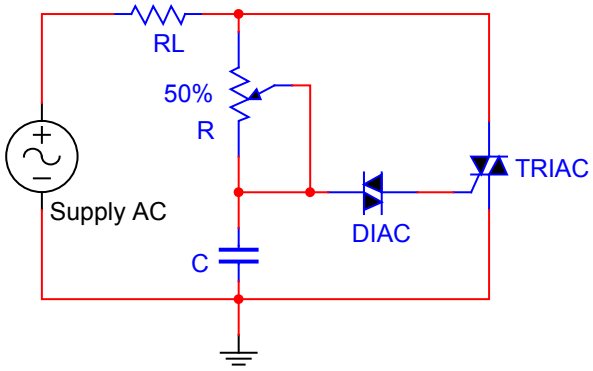

Equivalence of the TRIAC with two SCRs

The TRIAC can be activated (triggered) with positive or negative pulses applied to the gate.

A small gate current is usually able to control a high load current, for example: 20 mA of gate current can control 25 Amp of current in the load.




This schematic shows the components of a thyristor circuit (In this case it is a TRIAC) that controls the amount of current applied to the load ($R_L$).




When an alternating voltage is applied to the circuit, current flows through resistor R and charges the capacitor. As the applied voltage increases, the charge in the capacitor increases too. Meanwhile, the TRIAC remains off.

When the voltage across the capacitor is greater than the breakover voltage of the DIAC, the capacitor discharges through the DIAC producing a pulse current that triggers the TRIAC.

Once the TRIAC is activated, it remains on and a high current flows through the load $R_L$. On the other hand, the voltage between the terminals of the RC network is limited by the voltage between the anode and cathode of the TRIAC which is about 1 volt.

This situation continues until the half cycle of the AC source ends. At that point the applied voltage is about zero and the current through the TRIAC drops below the "holding current", turning the TRIAC off.

The AC source then enters the following half cycle, the capacitor starts charging again (in the opposite direction) and the cycle repeats.

With appropriate RC values, the output can be varied from zero until near the maximum power by adjusting the potentiometer.

**DIAC Symbol**

**The DIAC** is a bidirectional diode connected at the SCR or TRIAC's gate when using alternating current. The DIAC has the following property: for small applied voltages there is no current flow, but if the applied voltage is increased above a value called breakover voltage, the device begins to conduct. The breakover voltage is around 30 V.

Among the more common applications of the TRIAC are lamp dimmers and motor speed controllers.

**Physical shapes of TRIACS**

## Problem Statement

Build an SCR or TRIAC circuit that controls the power across a lamp which constitutes the load. For safety reasons use a power transformer that provides a secondary voltage of 12.6 VAC. You should be able to vary a potentiometer to control how much power is going to the load.

## Solution

The appropriate circuit is shown in Figure 6-1.

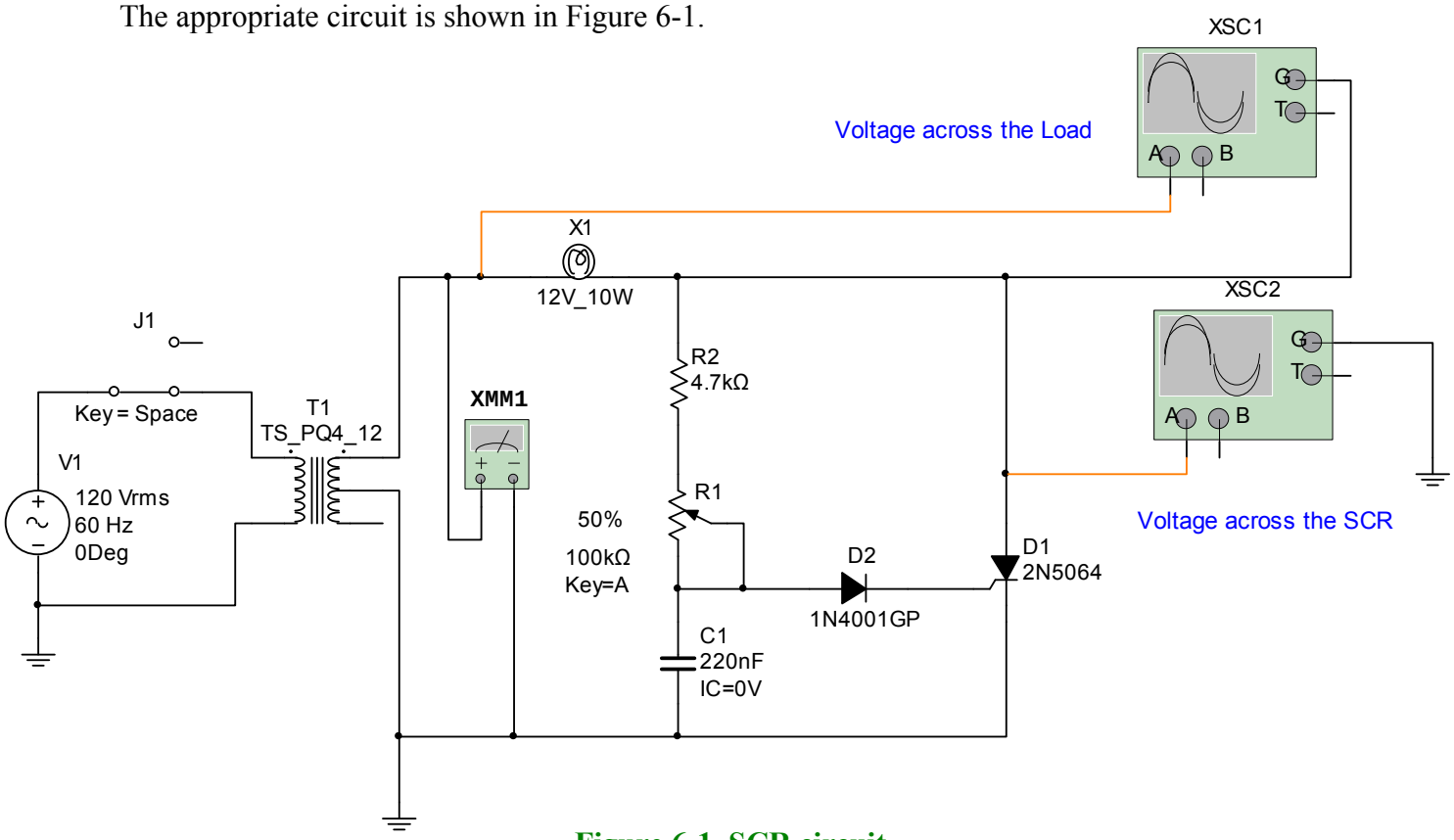



**Figure 6-1. SCR circuit**

## Circuit analysis

Because we are using a low AC voltage for safety reasons, transformer T1 reduces the 120 VAC to about 12 VAC. R1 and C1 determine the time constant; in other words, how fast or how slow the capacitor charges. In this case, the time is varied by R1. R2 is added as a limiting resistor in the event that R1 is inadvertently adjusted to 0 Ω. Notice that instead of a DIAC we are using a diode. A DIAC needs a voltage greater than 30 V to conduct. Therefore, with the input voltage being only 12 VAC, a DIAC will not work. In addition, an SCR cannot withstand negative voltages at its gate; therefore, a diode allows only the positive alternations to be applied to the gate and prevents the negative alternations from reaching the gate.

The lamp X1 constitutes the load. By varying the time constant through R1 you will notice how the duty cycle of the lamp will also vary. You can view the waveforms with oscilloscopes in Multisim. Oscilloscope XSC1 measures the voltage across the load (X1) and oscilloscope XSC2 measures the voltage across the SCR (D1).

## Procedure/Tasks Using Multisim

1. Build the circuit in Figure 6-1 using Multisim. Make sure that you use two oscilloscopes, one to observe the signal at the load (X1) and the other to observe the signal at the SCR (D1). In addition, ensure that X1 is a 12 V lamp.

2. Adjust R1 (by pressing the A or SHIFT-A keys) and observe the waveforms at the load and at the SCR. In addition, observe the duty cycle of the lamp and see how it corresponds to the SCR waveforms.

3. You should notice then that R1 controls how much voltage is applied to the lamp (load). Figure 6-2 shows an example of the voltage that you should observe across the load and across the SCR.

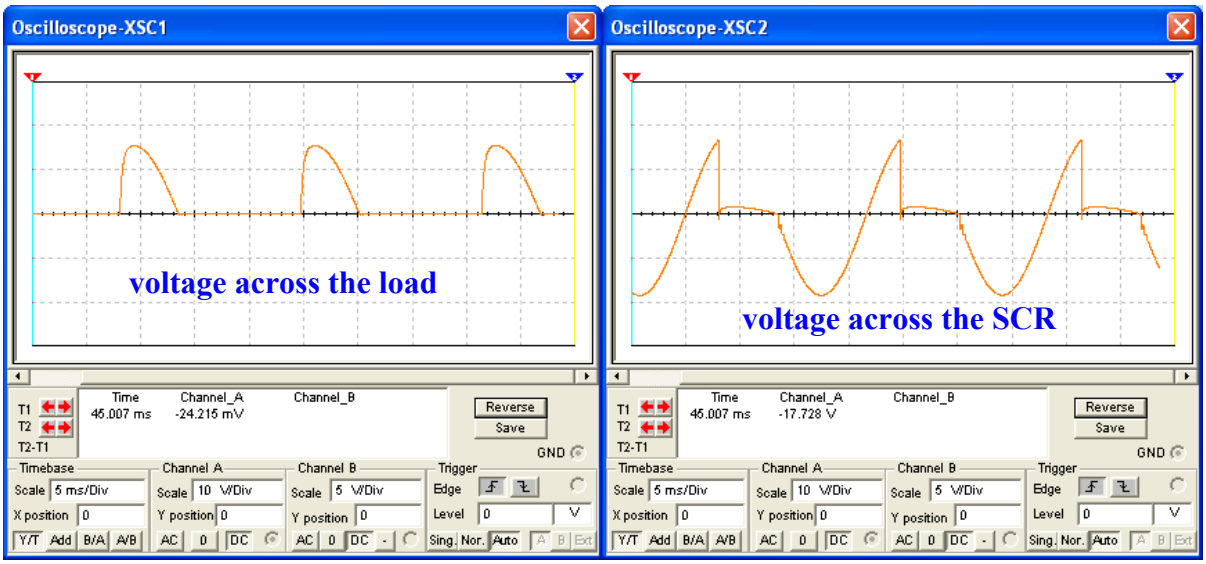



**Figure 6-2. Waveforms across the load and across the SCR.**

By observing the waveforms in Figure 6-2, it can be seen that when the SCR is conducting the voltage across the SCR is about zero. Additionally, when the voltage across the load is zero, the SCR is off and the corresponding portion of the applied voltage appears across the SCR.

4. Build the circuit depicted in Figure 6-3. Notice that we are using a higher VAC and we can use the DIAC (D2). In this case, make sure that X1 is a 120 V lamp because the applied voltage is 120 VAC. D3 ensures that the capacitor charges during the positive alternations only, because the SCR's gate cannot receive negative voltages.

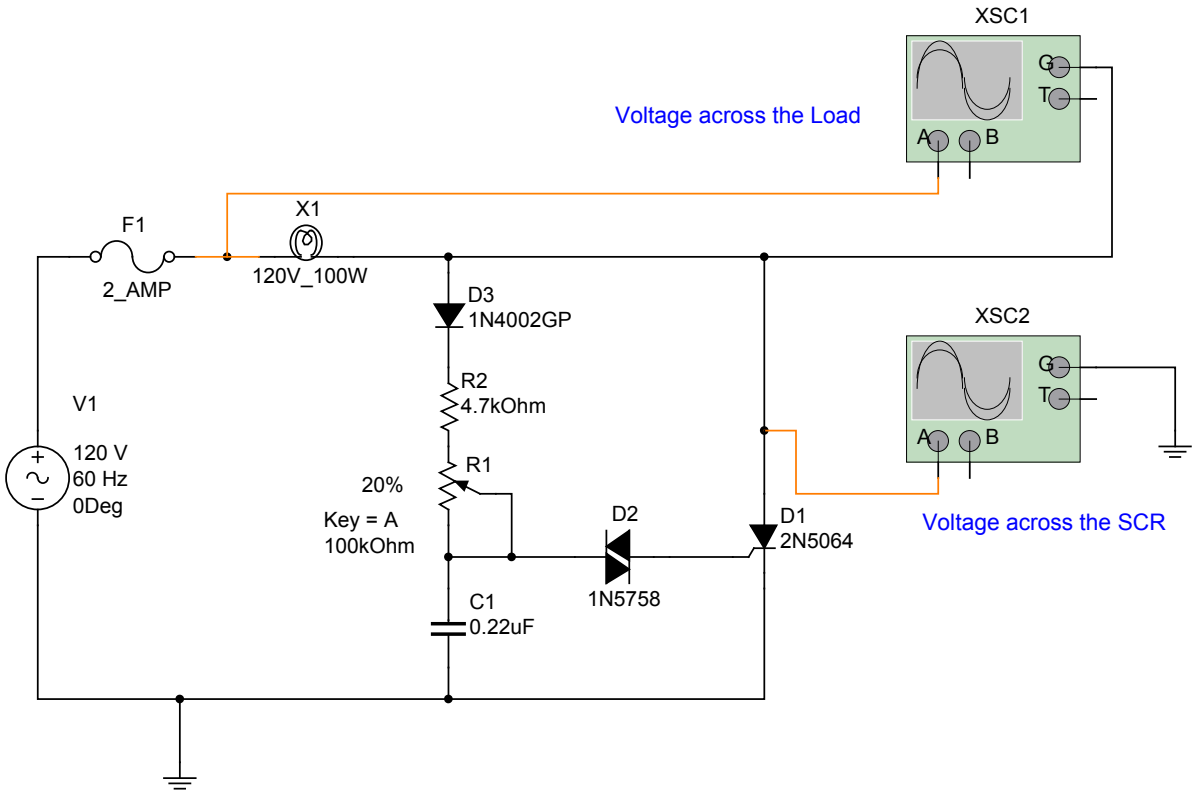



**Figure 6-3. SCR circuit with a DIAC and a higher input voltage**.

5. Repeat steps 2 and 3. Do you observe the same behaviors and waveforms as in the circuit of Figure 6-1? Describe any difference and explain the reasons.

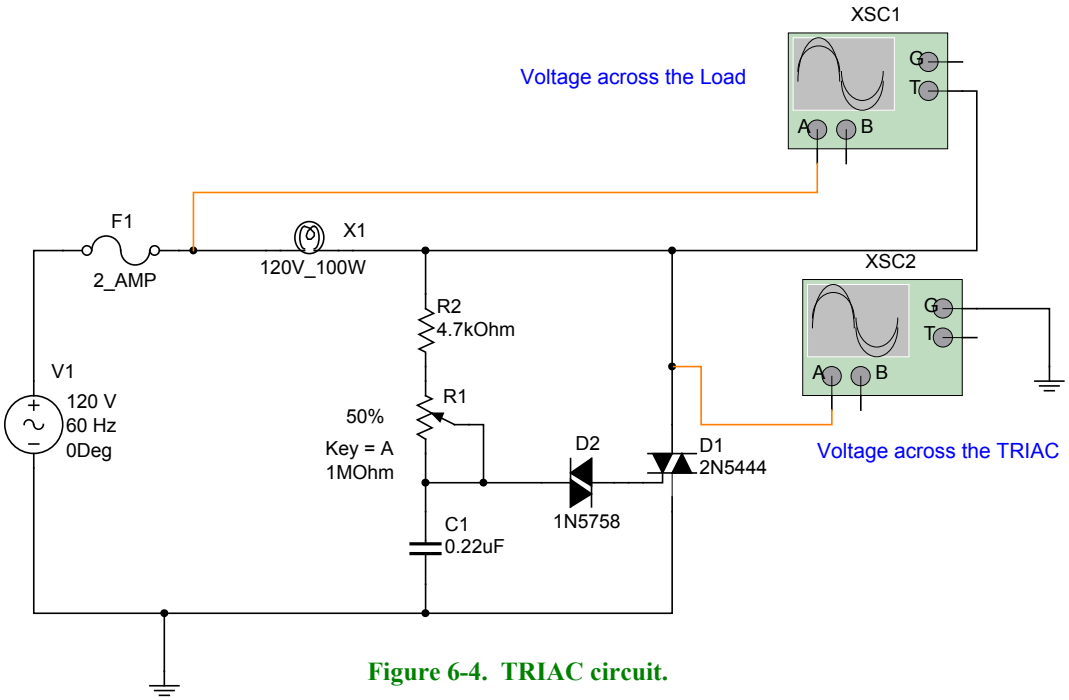



**Figure 6-4. TRIAC circuit.**

6. Now, build the circuit indicated in Figure 6-4 using a TRIAC instead of an SCR.

7. By activating the circuit, observing the waveforms, and the lamp behavior what are your conclusions of the circuit in Figure 6-4 respect the one in Figure 6-3?

## Questions

1. What would happen if you use a 12-V lamp in the circuit of Figure 6-3?

2. Can you replace the SCR by a TRIAC in Figure 6-3? Explain

3. Assuming that the breakover voltage of DIAC D2 is 30 V in Figure 6-3 and the voltage across the capacitor reaches – 30 V (negative voltage), Is this possible? Why?

4. Why is the reason that the circuit in Figure 6-4 does not need diode D3 as in Figure 6-3?

## Procedure/Tasks – Hardwired Circuit

You are going to hardwire one of the circuits that you built and analyzed using Multisim. This time you are going to hardwire the circuit shown in Figure 6-1. As you did in previous labs, before continuing you are going to indicate the list of components and the equipment that you need to complete this task.

### List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

### Equipment List

1. Hardwire the circuit shown in Figure 6-1. If you don't have access to an oscilloscope, then disregard it in this part. You will observe the circuit behavior by looking at the lamp brightness. If you do not have an oscilloscope, then skip step 2 and go to step 3.

2. When taking the measurements in the hardwired circuit, connect the oscilloscope to measure the voltage across the load only. **DO NOT** attempt to use a dual-trace oscilloscope to measure the voltage across the load and the voltage across the SCR (D1). What you can do is connect the oscilloscope as XSC1 in Figure 6-1 to observe the waveform at the load. Then, disconnect the oscilloscope and connect it the way XSC2 is connected in Figure 6-1 to observe the waveform across the SCR.

3. Adjust R1 and observe the lamp's brightness. If you have an oscilloscope, observe the waveform across the load first. Then connect the oscilloscope across the SCR and observe the waveforms across it.

4. What do you observe with the lamp's brightness as you vary the R1 potentiometer?

5. Do you observe the same results when you use Multisim? Describe any discrepancies.

## Questions

1. Can we substitute the lamp by a resistor?

2. What would happen if diode D2 is reversed?

3. What would happen if diode D2 opens?

4. What would happen if diode D2 shorts?

5. What difficulties have you encountered in hardwiring the SCR circuit of Figure 6-1?

## Conclusions

After completing the SCR circuit both in Multisim and hardwire, what are your conclusions?

# LAB 7 – THE INTEGRAL CONTROLLER

## Objectives

1. Build, test, and analyze an integral controller circuit using Multisim.
2. Predict the expected output waveform of an inverting integral controller given a specific input waveform and then verify it using Multisim.
3. Design a non-inverting integral controller given a certain time constant and voltage supplies using Multisim.

## Preliminary Information

Read "Controllers with Multisim" to gain an understanding about process control and how to simulate controllers using Multisim.

The integral controller monitors the average error over a period of time. If an offset error exists, the integral unit will, during the time that error exists, accumulate the value of the error and output a correction signal whose rate of change (slope or *dv*/*dt*) is equal to the magnitude of the accumulated error signal..

An integral unit is usually used together with a proportional unit and the two units are designated as a PI controller (Proportional Integral).

Let's assume that he signal shown in Figure 7-1 is applied to an integral controller.

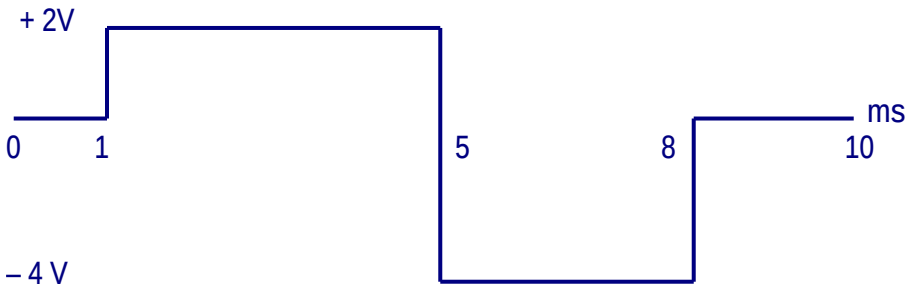



**Figure 7-1. Input waveform applied to an Integral Controller**

The block of an integral controller is depicted in Figure 7-2 and its corresponding output equation is as shown in equation 1.

$$\text{Vout} = \int \text{Vin} \quad (1)$$


**Figure 7-2. Block diagram of an integral controller.**

To integrate the input signal, we will divide it in four functions that we call: $f_1(x)$, $f_2(x)$, $f_3(x)$, and $f_4(x)$ as indicated in Figure 7-3.

Then: $Vin = f_1(x) + f_2(x) + f_3(x) + f_4(x)$ (2)

Observing the graph of the input signal, each function is defined in the following way:

$f_1(x): \; y = 0, \quad 0 < x < 1$ (3)

$f_2(x): \; y = 2, \quad 1 < x < 5$ (4)

$f_3(x): \; y = -4, \quad 5 < x < 8$ (5)

$f_4(x): \; y = 0, \quad 8 < x$ (6)

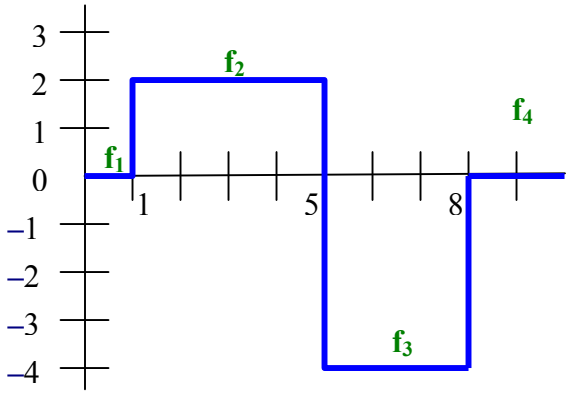



**Figure 7-3. Input signal divided into four segments for analysis purposes.**

Integrating both members of equation 2:

$$\int Vin = \int_0^1 f_1(x)\,dx + \int_1^5 f_2(x)\,dx + \int_5^8 f_3(x)\,dx + \int_8^\infty f_4(x)dx \qquad (7)$$

Replacing equations 1, 3, 4, 5, and 6 in equation 7 and integrating each term, we have:

$$\int Vin = Vout \qquad (8)$$

$$\int_0^1 f_1(x) = \int_0^1 0\,dx = [\,0\,]_0^1 = 0 \qquad (9)$$

$$\int_1^5 f_2(x) = \int_1^5 2\,dx = [\,2x + C\,]_1^5 \qquad (10)$$

$$\int_5^8 f_3(x) = \int_5^8 -4\,dx = [\,-4x + C\,]_5^8 \qquad (11)$$

$$\int_8^\infty f_4(x) = \int_8^\infty 0\,dx = [\,0 + C\,]_8^\infty = C \qquad (12)$$

## Output Waveform

To graph the output waveform, we will proceed to find the intersection points of equations: 9, 10, 11 and 12.

**Step 1:** Finding the points for equation 9: $\int_0^1 f_1(x)\,dx = [\,0\,]_0^1 = 0$

The result of equation 9 is zero, thus the graph is a straight line with a slope of zero (parallel to the x axis), starting at the initial point (0, 0) until the final point (1,0) because this it is the distance that covers the function $f_1(x)$. **See line "a" in Figure 7-7.**

**Step 2:** Finding the points for equation 10: $\int_1^5 f_2(x)\,dx = [\,2x + C\,]_1^5$

**Calculation of C constant**. See figure 7-4.

1. The value of x is similar to the common point where the previous function ends and the following function begins. In this case it is 1.

2. To solve for the integration constant (C) we must take the function we are integrating and set it equal to the y value endpoint calculated from the previous interval. In this case it is zero.
   For: 2x + C and solving for C:

   $2(1) + C = 0 \quad \rightarrow \quad C = -2$

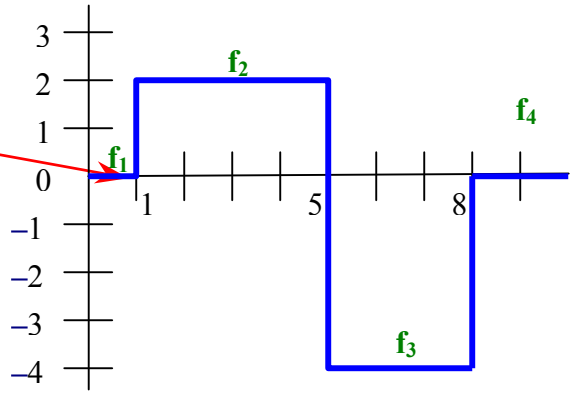



**Figure 7-4. Graph to find the C constant value from equation 10.**

Evaluating equation 10 with the minimum x value (1) and maximum x value (5):

$$\int_1^5 f_2(x) = \int_1^5 2\,dx = [\,2x + C\,]_1^5$$

For x = 1 ➔ $[2x + C]_{x=1}$

Replacing values: $2(1) + (-2) = 0$ ➔ $\int f_2(x)dx = 0$

For x = 5 ➔ $[2x + C]_{x=5}$

Replacing values: $2(5) + (-2) = 8$ ➔ $\int f_2(x)dx = 8$

The points for the graph of equation 10 are: Initial point: (1, 0)
Final point: (5, 8)

**See line "b" in Figure 7-7.**

**Step 3:** Finding the points for equation 11: $\int_5^8 f_3(x)\,dx = \left[ -4x + C \right]_5^8$

**Calculation of C constant**. See figure 7-5.

1. The value of x is 5

2. Solving for C in $f_2(x)$ we take the function we are integrating and set it equal to the y value endpoint calculated from the previous interval. In interval $f_2$ this value was 8.

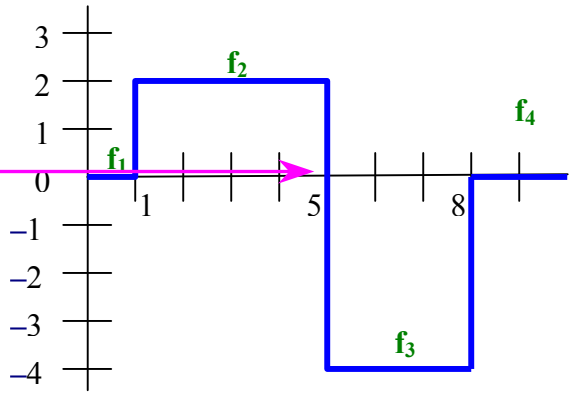



**Figure 7-5. Graph to find the C constant value from equation 11.**

For: $-4x + C$ and replacing values:

$-4(5) + C = 8 \quad \rightarrow \quad C = 28$

Evaluating equation 11 with the minimum x value (5) and maximum x value (8):

$$\int_5^8 f_3(x)dx = \int_5^8 -4\,dx = \left[ -4x + C \right]_5^8$$

For $x = 5 \quad \rightarrow \quad [-4x + C]_{x=5}$

Replacing values: $-4(5) + 28 = 8 \quad \rightarrow \quad \int f_3(x) = 8$

For $x = 8 \quad \rightarrow \quad -4x + C$

Replacing values: $-4(8) + 28 = -4 \quad \rightarrow \quad \int f_3(x) = -4$

The points for the graph of equation 11 are:

Initial point: (5, 8)
Final point: (8, – 4)

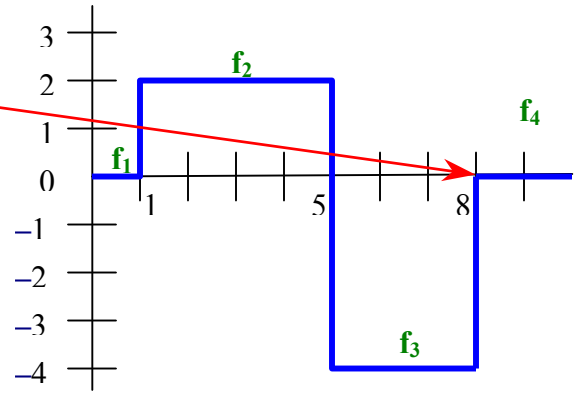
**See line "c" in Figure 7-7.**

**Step 4:** Finding the points for equation 12: $\int_{8}^{\infty} f_4(x)dx = [\,0\,]_{8}^{\infty} = C$

But $C = 0x + C$

**Calculation of C constant**. See figure 7-6.

1. The value of x is 8
2. Solving for C in $f_3(x)$ we take the function we are integrating and set it equal to the y value endpoint calculated from the previous interval. In interval $f_3$ this value was – 4.




For: $0x + C$ and replacing values

$0(8) + C = -4 \rightarrow C = -4$

**Figura 7-6. Graph to find the C constant value from equation 12.**

The graph is a straight line with a zero slope (parallel to the x axis), starting from the point (8, – 4) towards infinity.

**See line "d" in Figure 7-7.**

**Note**: ***This wave form (red) is the result of using an integral controller with non-inverted output.***

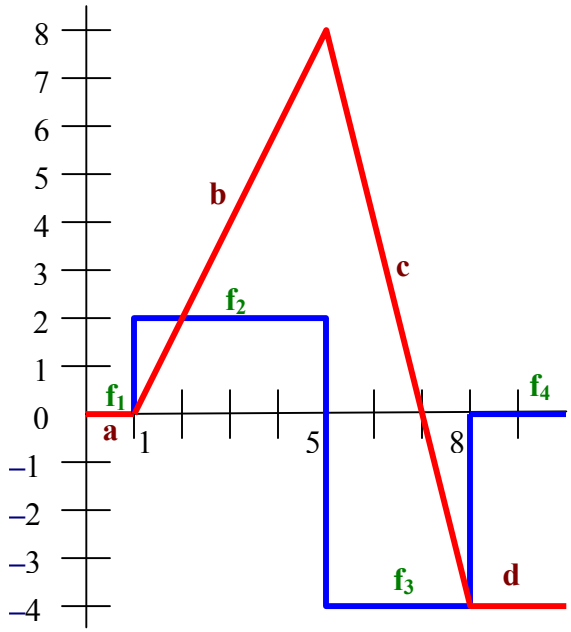



**Figure 7-7. Output of a non-inverting controller**

For an inverting integral controller, the output waveform (red line) is the one that is indicated in the adjacent graph.

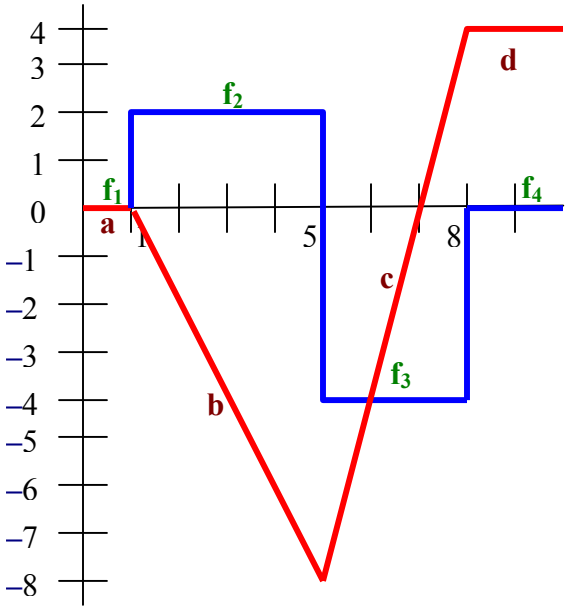



**Figure 7-8. Output of an inverting controller**

## Problem Statement

The input signal shown in Figure 7-1 is available. This figure is shown on page 1 but it's repeated here for convenience. Draw the expected output indicating its amplitude. The Integral Controller has a time constant RC = 1 ms and uses a supply of ± 12 V.

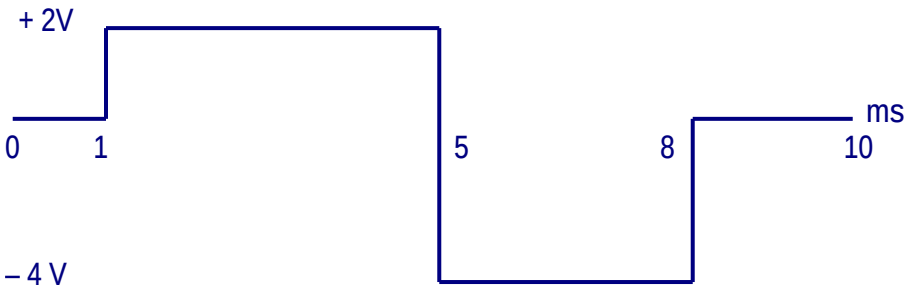



**Figure 7-1. Input waveform applied to an Integral Controller**

We assign points at the beginning of each segment as shown in Figure 7-9. These points are $t_0$, $t_1$, $t_2$, $t_3$ and $t_4$.

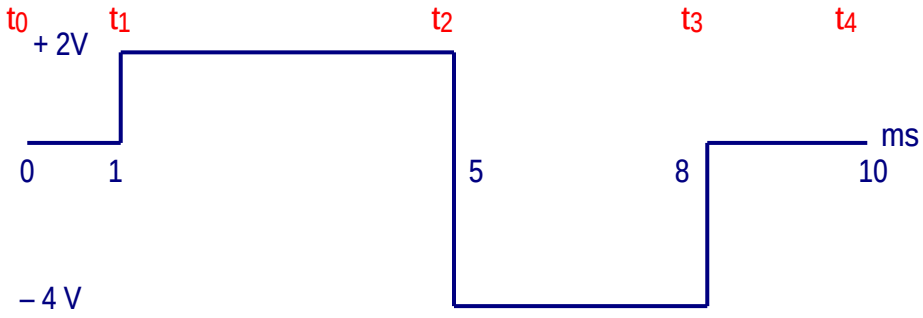



**Figure 7-9. Input waveform divided by times**

## Expected Output Waveform

The equation of an integrator with a constant error input is:

$$Vo = \frac{-Vin\ t}{RC} + C \qquad (13)$$

The procedure is as follows:

**(a) For time $t_0$ to $t_1$ and using equation 13 we have:**

Vin = 0 V; $t = t_1 - t_0 = 1 - 0 = 1$ ms; RC = 1 ms; C = 0.

C is the previous Vo. Since the previous or initial Vo is not given we assume it is 0.

$$Vo = \frac{-Vin\ t}{RC} + C = \frac{-(0)(1ms)}{1\,ms} + 0 = 0$$ **Segment "a" in Figure 7-8.**

**(b) For time $t_1$ to $t_2$ and using equation 13 we have:**

Vin = 2 V; $t = t_2 - t_1 = 5 - 1 = 4$ ms; RC = 1 ms; C = 0.

C is equal to 0 because the calculated previous Vo is 0.

$$Vo = \frac{-Vin\ t}{RC} + C = \frac{-(2V)(4ms)}{1\,ms} + 0 = -8V$$ **Segment "b" in Figure 7-8.**

**(c) For time $t_2$ to $t_3$ and using equation 13 we have:**

Vin = - 4 V; $t = t_3 - t_2 = 8 - 5 = 3$ ms; RC = 1 ms; C = – 8 V.

C is equal to – 8 V because the calculated previous Vo is – 8 V.

$$Vo = \frac{-Vin\ t}{RC} + C = \frac{-(-4V)(3ms)}{1\,ms} + (-8) = +4V$$ **Segment "c" in Figure 7-8.**

**(d) For time $t_3$ to $t_4$ and using equation 13 we have:**

Vin = 0 V; $t = t_4 - t_3 = 10 - 8 = 2$ ms; RC = 1 ms; C = + 4 V.

C is equal to + 4 V because the calculated previous Vo is + 4 V.

$$Vo = \frac{-Vin\ t}{RC} + C = \frac{-(0V)(2ms)}{1\,ms} + 4V = +4V$$ **Segment "d" in Figure 7-8.**

## Procedure/Tasks Using Multisim

1. Build the circuit shown in Figure 7-10 using Multisim.

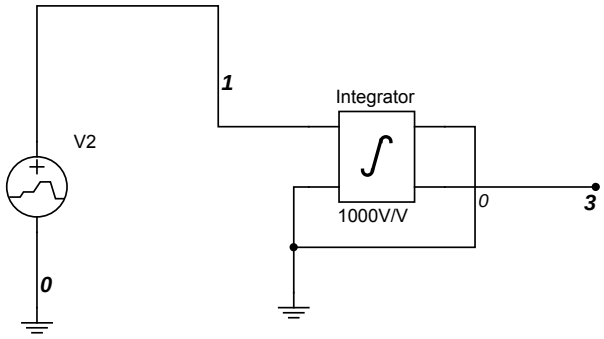



**Figure 7-10. Inverting Integrator controller**

The Integrator block is the Voltage Integrator block in Multisim.

## Voltage Integrator Block

In Multisim, you can have an inverting integrator or a non-inverting integrator as indicated in Figure 7-11. In this case we are using an inverting integrator.

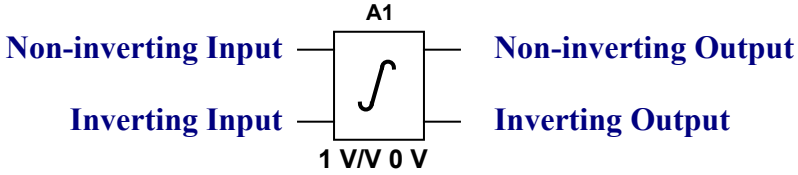



**Figure 7-11. Integrator block in Multisim**

For the inverting integrator we can use the non-inverting input and the inverting output or the inverting input and the non-inverting output. In either case both the input and output terminals that are not used must be grounded. In this lab we'll use the non-inverting input and the inverting output.

The output equation of an integrator was indicated previously in equation 13 and repeated here for convenience:

$$Vo = \frac{-Vin\ t}{RC} + C \qquad (13)$$

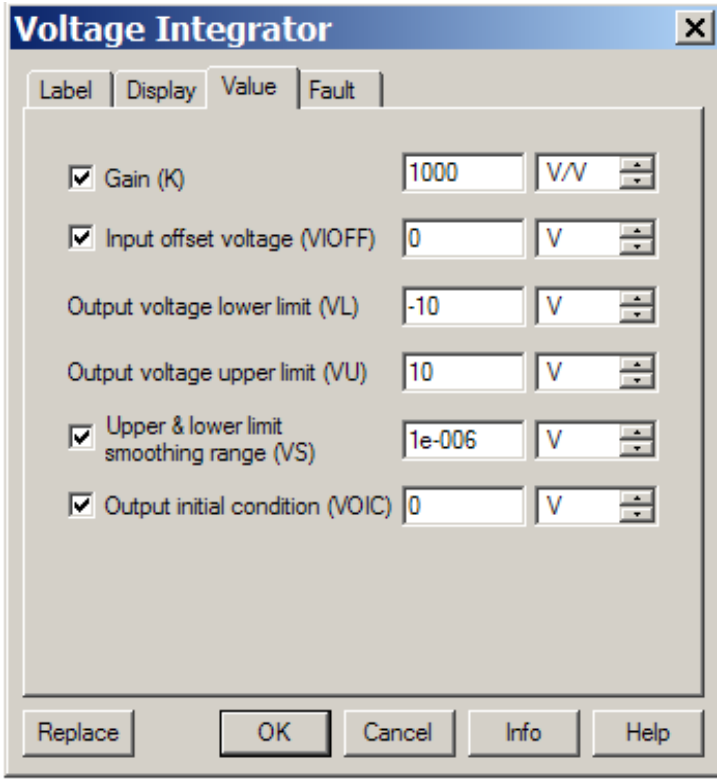
If we make $\frac{1}{RC} = K$ then we have:

$$Vo = - Vin\ t\ K + C \qquad (14)$$

According to the problem statement RC = 1 ms; thus, K = 1/RC = 1/0.001 s = 1000 (1000 V/V in the Integrator Block).

In addition, the supply is ± 12 V. This means that the saturation voltage is ± 10 V (2 V less than the supply voltages).

The way we change the Gain (K) and the saturation voltage is by double-clicking the Integrator block. When you do this, you'll observe the following window:




The Gain of 1000 is changed in Gain (K).

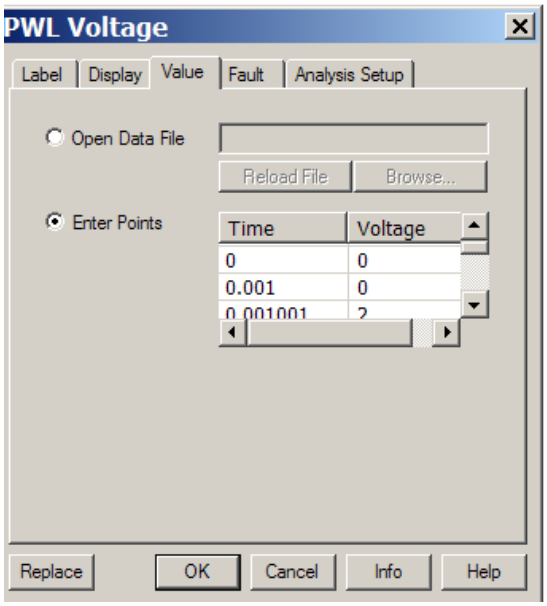
– Vsat is the same as the **"Output voltage lower limit (VL)"**

+ Vsat is the same as the **"Output voltage upper limit (VU)"**

The initial voltage (Vinitial) is set in **"Output initial condition (VOIC)"**

The error signal source (**V2**) is called a **Piecewise Linear Voltage** (PWL) which produces the type of waveform such as the one needed in this lab experiment.

When you double-click V2 you'll observe the following window:




To produce the needed waveform, enter the following values:

| Time | Voltage | Description |
|---|---|---|
| 0 | 0 | At 0 s, the voltage is 0 |
| 1e–3 | 0 | At 1 ms, the voltage is still 0 |
| 1.001e–3 | 2 | At 1.001 ms, the voltage jumps to 2 V |
| 5e–3 | 2 | At 5 ms, the voltage is still 2 V |
| 5.001e–3 | – 4 | At 5.001 ms, the voltage goes to – 4 V |
| 8e–3 | – 4 | At 8 ms, the voltage is still – 4 V |
| 8.001e–3 | 0 | At 8.001 ms, the voltage goes to 0 V |
| 10e–3 | 0 | At 10 ms, the voltage remains at 0 V |

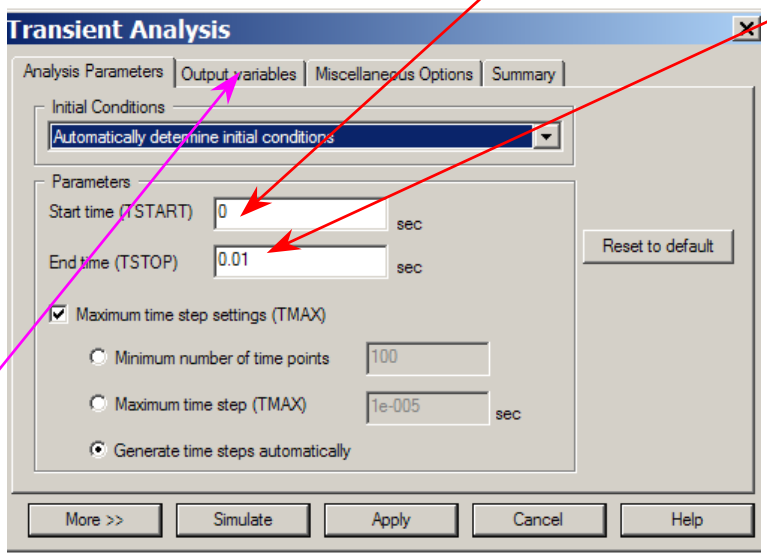
## Circuit Operation

**DO NOT** turn on the power switch. In this lab experiment you are going to use the Transient Analysis:

**Simulate → Analyses → Transient Analysis**

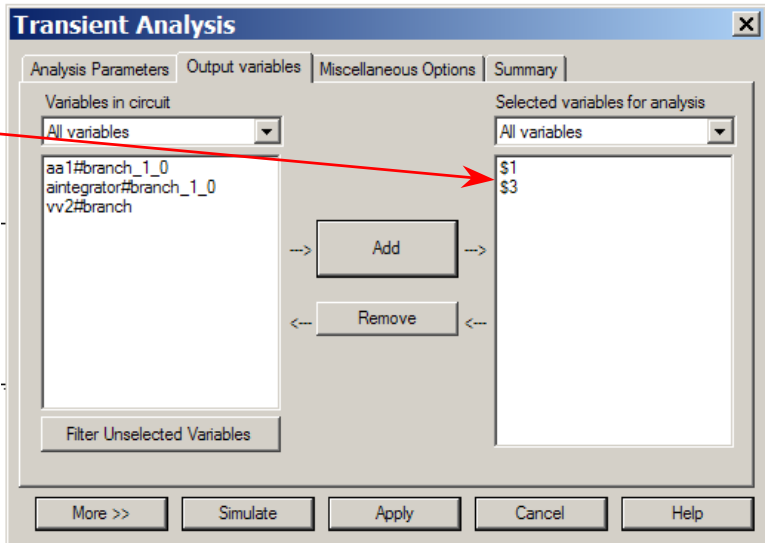
Make sure that you enter the following parameters:

Click on the Analysis Parameters tab. The Start time is set to 0 sec and the end time is set to 0.01 sec (10 msec).




Click on the Output variables tab.

Add nodes 1 (input) and 3 (output).




***Note: These numbers can be different in your circuit depending on the order that the components have been selected, so make sure that you add the proper nodes.***

Next, click on Simulate. You should observe the waveforms depicted in Figure 7-12. The red waveform is the input and the blue waveform is the output.

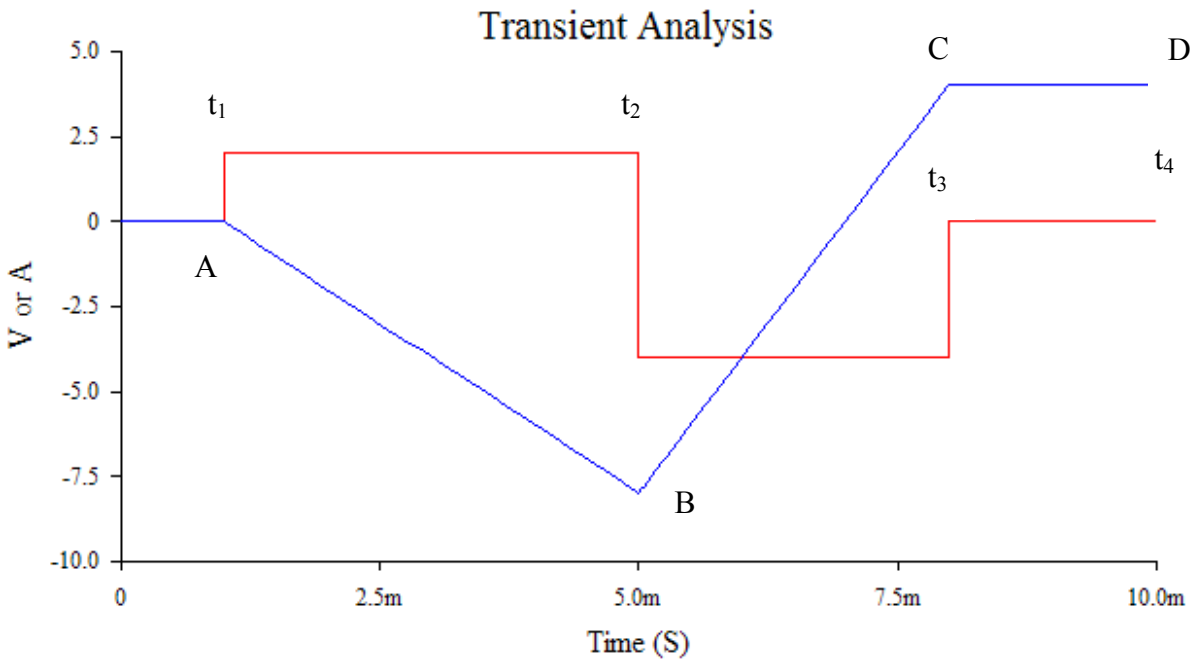



**Figure 7-12. Waveforms of the Inverting Integral Controller using Multisim**

## Questions

1. Does the output waveform obtained in Multisim match the calculated one? Explain any discrepancies.

2. Rearrange the circuit so that we use the non-inverting output and verify its behavior. What do you observe in the output waveform?

3. Change RC to 0.5 ms and compare the measured output waveform with the calculated one. Explain why the output waveform is distorted.

4. The input waveform illustrated in Figure 7-13 is applied to an inverting integrator with: RC = 2 ms and a supply of ± 15 V.

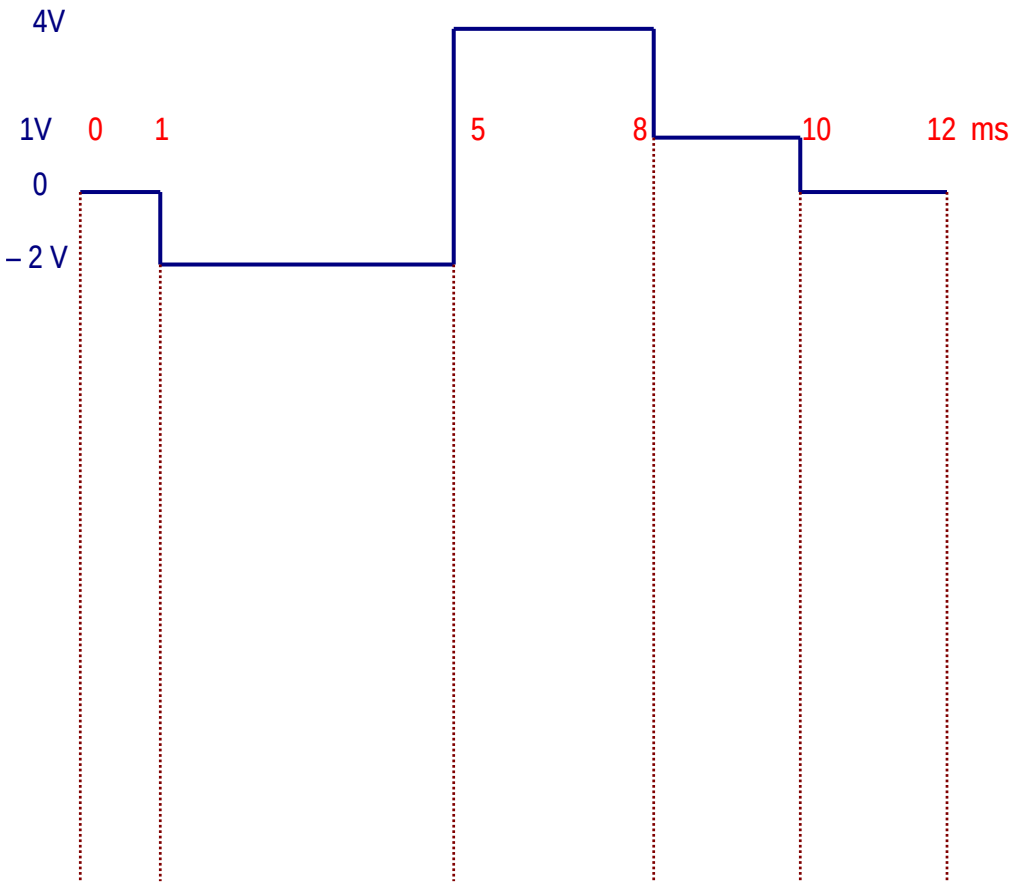



**Figure 7-13. Input waveform for question 4.**

a. Draw the expected output waveform below the input signal in Figure 7-13 indicating its amplitude.

b. Draw the appropriate circuit using Multisim and observe the output waveform. Does the observed output waveform match the one you drew in step 4a? Explain any discrepancies.

5. The input waveform shown in Figure 7-14 is applied to a non-inverting integrator with: RC = 5 ms and a supply of ± 18 V.

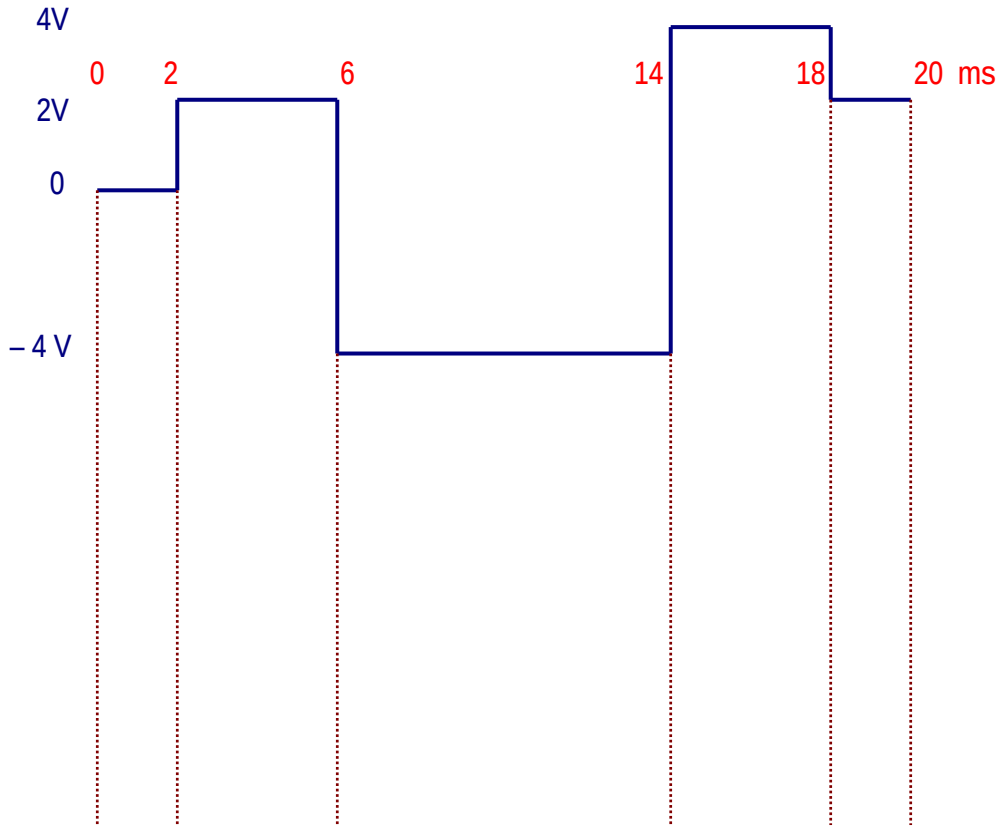



**Figure 7-14. Input waveform for question 5.**

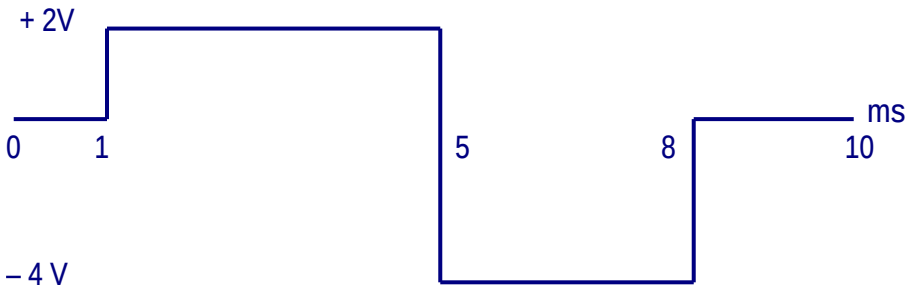
a. Draw the expected output waveform below the input signal in Figure 7-14 indicating its amplitude.
b. Draw the appropriate circuit using Multisim and observe the output waveform. Does the observed output waveform match the one you drew in step 5a? Explain any discrepancies.

**Conclusions.** What are your conclusions regarding the Integrator Controller using Multisim?

## Advanced Design

**If you want to earn extra credit, try the following design.**

The input waveform shown in Figure 7-1 is applied to a Proportional-Integral (PI) controller. The integral controller has an RC constant of 1 ms and a supply of ± 16 V and the proportional controller has a gain of –1. Test your solution using Multisim and compare your output waveform with the one obtained in Multisim. For your convenience, Figure 7-1 is shown below.



# LAB 8 – DERIVATIVE CONTROLLER

## Objectives

1. Build, test, and analyze an inverting derivative controller using Multisim.
2. Predict the expected output waveform of an inverting derivative controller given a specific input waveform and verify it using Multisim.
3. Design a non-inverting derivative controller given a certain time constant and voltage supplies using Multisim.

## Preliminary Information

Read "Controllers with Multisim" to gain an understanding of process control and how to simulate controllers using Multisim.

The Derivative Controller delivers a signal proportional to the rate of change (*dv*/*dt*) of the error signal. When the controlled variable coincides with the reference point (set point), the derivative signal is zero; conversely, when the value of the controlled variable changes, the derivative signal is large. The result is a quick controller response to load changes.

Next, we will mathematically analyze the Derivative Controller when we apply the signal shown in figure 8-1.

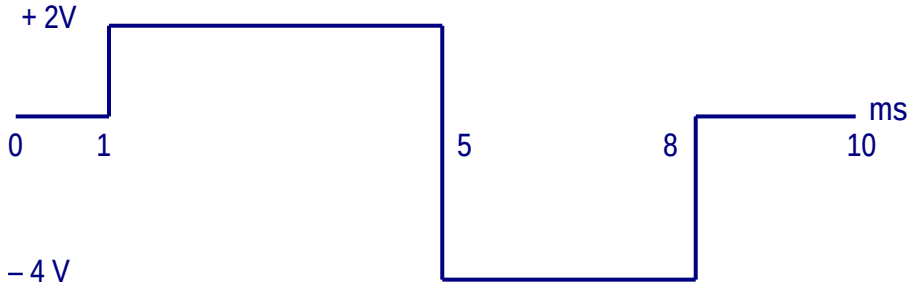



**Figure 8-1. Signal applied to the input of the Derivative Controller.**

The Derivative Controller's block is shown in figure 8-2 and equation 1 is its corresponding output equation.

$$\text{Vout} = \frac{\text{d}}{\text{dt}}\text{Vin} \qquad (1)$$


**Figure 8-2. Block Diagram of a Derivative Controller**

For the analysis, we will divide the input signal in four functions that we will call: $f_1(x)$, $f_2(x)$, $f_3(x)$, and $f_4(x)$ as indicated in Figure 8-3.

Then: $Vin = f_1(x) + f_2(x) + f_3(x) + f_4(x)$ (2)

Each one of these four functions is based on the graph of figure 8-3, in the following way:

$f_1(x): \; y = 0, \quad 0 < x < 1$ (3)

$f_2(x): \; y = 2, \quad 1 < x < 5$ (4)

$f_3(x): \; y = -4, \quad 5 < x < 8$ (5)

$f_4(x): \; y = 0, \quad 8 < x$ (6)

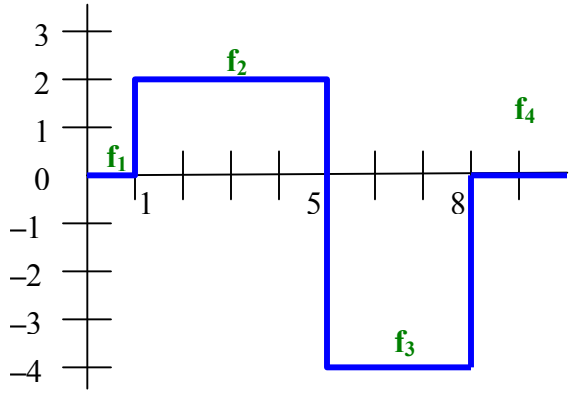



**Figure 8-3. Input signal divided into four segments for analysis purposes.**

Deriving both sides of equation 2:

$$\frac{d}{dt}Vin = \frac{d}{dt}f_1(x) + \frac{d}{dt}f_2(x) + \frac{d}{dt}f_3(x) + \frac{d}{dt}f_4(x) \qquad (7)$$

Deriving equations 3, 4, 5, 6 separately, we have:

$$\frac{d}{dt}f_1(x) = \frac{d}{dt}0 = 0 \qquad (8)$$

$$\frac{d}{dt}f_2(x) = \frac{d}{dt}2 = 0 \qquad (9)$$

$$\frac{d}{dt}f_3(x) = \frac{d}{dt}-4 = 0 \qquad (10)$$

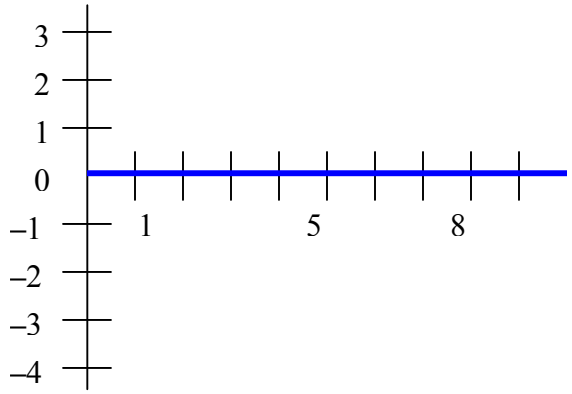
$$\frac{d}{dt}f_4(x) = \frac{d}{dt}0 = 0 \qquad (11)$$

Replacing equations 8, 9, 10 and 11 in equations 7 and 1, we have:

$$Vout = \frac{d}{dt}Vin = 0 + 0 + 0 + 0 = 0 \qquad (12)$$

Equation 12 indicates that the function results in a straight line with zero slope that is coincident with the x axis. The graph of equation 12 is indicated in figure 8-4.

**Figure 8-4. The blue line shows the output waveform based on equation 12**




Observing figure 8-3, notice that at the junction points of functions $f_1$ and $f_2$ (where x = 1), $f_2$ and $f_3$ (where x = 5), $f_3$ and $f_4$ (where x = 8); there is discontinuity of the total function because there are two values in the y-axis for a single value in the x-axis. For example:

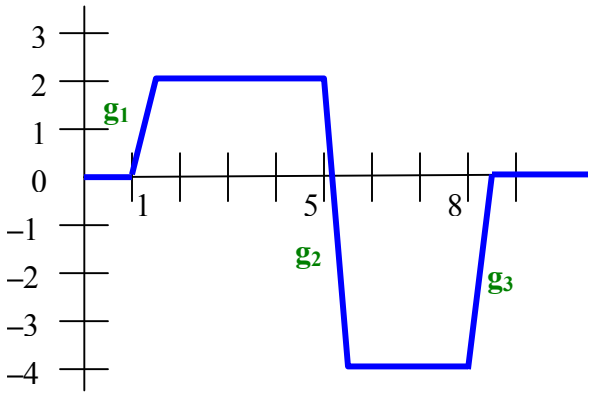
For x = 1 we have y = 0
y = 2

For x = 5 we have y = 2
y = – 4

For x = 8 we have y = – 4
y = 0

Multisim works only with continuous functions; therefore, to resolve the discontinuity issue of the total function we should modify the waveform slightly, in the following way:

**Figure 8-5. Waveform drawn with exaggeration to show the modifications in relationship to the original signal of figure 8-3. The functions are only indicated in the discontinuity points.**




In figure 8-5, we have increased the x-axis by the thousandth between the end and beginning of the following function. This incremental increase should be the possible minimum so that the original signal does not change.

Now, we will analyze the output of the Derivative Controller at the intersection points of the functions that we call $g_1(x)$, $g_2(x)$, $g_3(x)$. We will find the derivative of each of these new functions in the following way:

**Deriving the function $g_1(x)$.**

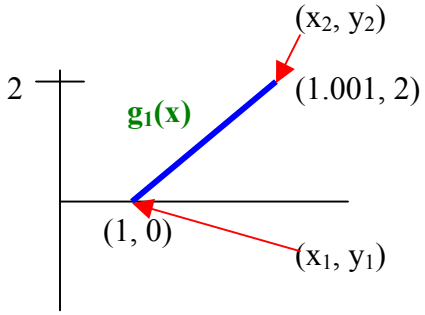



**Figure 8-6. Enlarged view of the function $g_1(x)$ with the initial and final points.**

$y = mx + b$ (1) equation of the straight line

Applying equation 1 to the function $g_1(x)$, we have: $g_1(x) = mx + b$ (2)

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (3)$$ definition of the slope of a straight line

Substituting the values of figure 8-6 in equation 3, we have:

$$m = \frac{2 - 0}{1.001 - 1} = 2000 \quad (4)$$

Substituting equation 4 for m in equation 2:

$g_1(x) = 2000x + b$ (5)

Differentiating both sides of equation 5:

$$\frac{d}{dx} g_1(x) = \frac{d}{dx}(2000x + b) \quad (6)$$

$$\frac{d}{dx} g_1(x) = 2000 \quad (7)$$

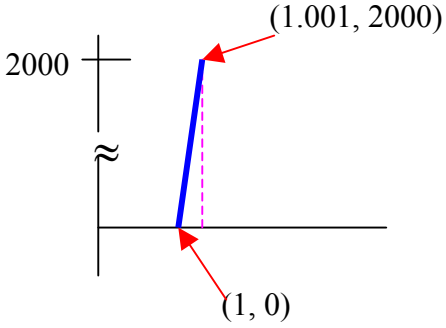



**Figure 8-7. Graph of the derivative of function $g_1(x)$**

***Note:*** ***It is not necessary to find the value of the constant b, because the derivative of a constant is zero.***

The graph of equation 7 is a straight line with a positive slope of 2000. Remember that every number can be thought of as having some arbitrary unit value for its denominator. Therefore, to graph equation 7 we should start from the initial point (1, 0), a unit in the horizontal axis, and end 2000 units up to the vertical axis because the slope is positive.

***Note:*** ***The word unit, for the layout of the straight line should be understood in the following way: if we agree that the distance 2000 is equal to two centimeters, then the unit will be 1/2000 = 0.0005 cm = 0.005 mm***

## Deriving function $g_2(x)$

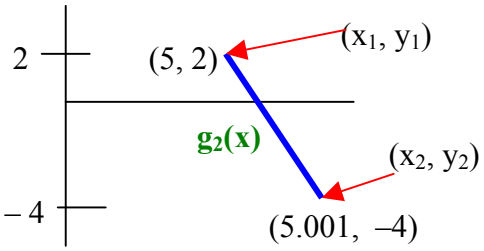



**Figure 8-8. Enlarged view of the function $g_2(x)$ with the initial and final points.**

$y = mx + b$ (1) equation of the straight line

Applying equation 1 to the function $g_2(x)$, we have: $g_2(x) = mx + b$ (2)

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (3) \quad \text{definition of the slope of a straight line}$$

Substituting the values of figure 8-8 in equation 3, we have:

$$m = \frac{-4-2}{5.001-5} = -6000 \quad (4)$$

Substituting equation 4 for m in equation 2:

$$g_2(x) = -6000x + b \quad (5)$$

Differentiating both sides of equation 5:

$$\frac{d}{dx} g_2(x) = \frac{d}{dx}(-6000x + b) \quad (6)$$

$$\frac{d}{dx} g_2(x) = -6000 \quad (7)$$

Since the second member of equation 7 has a negative sign, it corresponds to a straight line with a negative slope. In this case, we should start from the initial point (5, 0), a unit in the horizontal axis, and end 6000 units down to the vertical axis, as shown in figure 8-9.

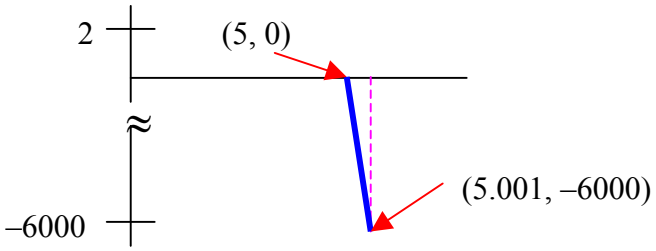
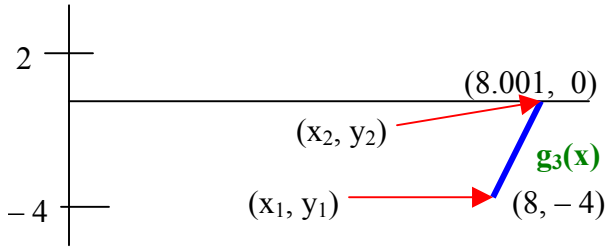



**Figure 8-9. Graph of the derivative of the function $g_2(x)$**

## Deriving the function $g_3(x)$

**Figure 8-10. Enlarged view of the function $g_3(x)$ with the initial and final points.**




To derive the function $g_3(x)$, we will proceed in the same way as we did for the functions: $g_1(x)$ and $g_2(x)$.

$y = mx + b$ (1) equation of the straight line

Applying equation 1 to the function $g_3(x)$, we have: $g_3(x) = mx + b$ (2)

$$m = \frac{y_2 - y_1}{x_2 - x_1} \quad (3)$$ definition of the slope of a straight line

Substituting the values of figure 8-10 in equation 3, we have:

$$m = \frac{0-(-4)}{8.001-8} = 4000 \quad (4)$$

Substituting equation 4 for m in equation 2:

$g_3(x) = 4000x + b$ (5)

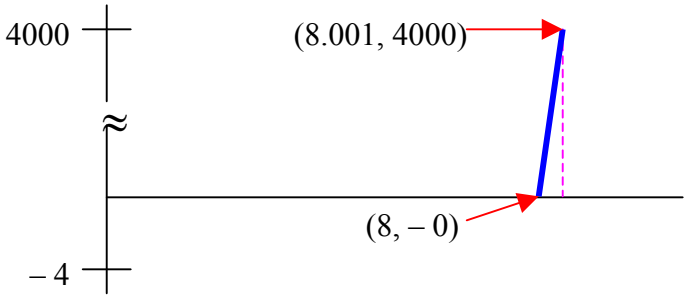
Differentiating both members of equation 5:

$$\frac{d}{dx} g_3(x) = \frac{d}{dx}(4000x + b) \quad (6)$$

$$\frac{d}{dx} g_3(x) = 4000 \quad (7)$$

The graph of equation 7 is a straight line with a positive slope that starts from the initial point (8, 0) in the horizontal axis and ends 4000 units up to the vertical axis, as indicated in figure 8-11.

**Figure 8-11. Graph of the derivative of the function $g_3(x)$**



The final graph of the output of the Derivative Controller is the sum of figures: 8-4, 8-7, 8-9 and 8-11.

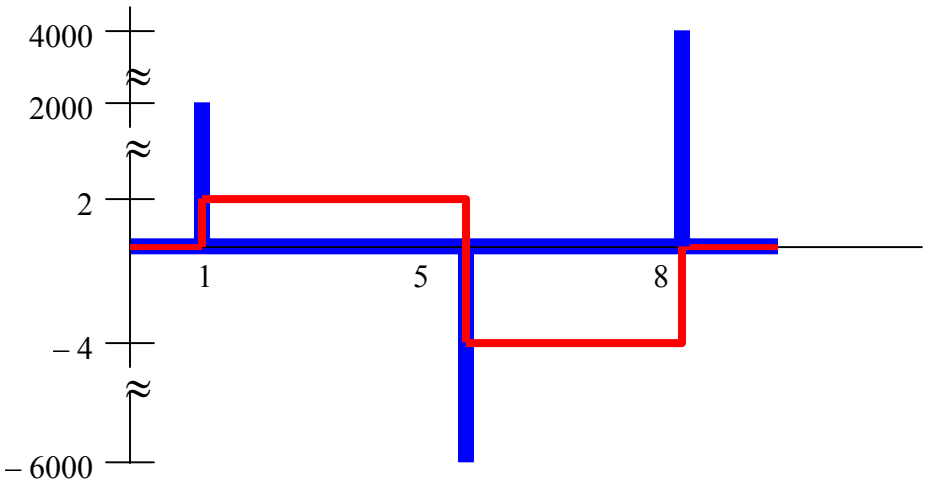
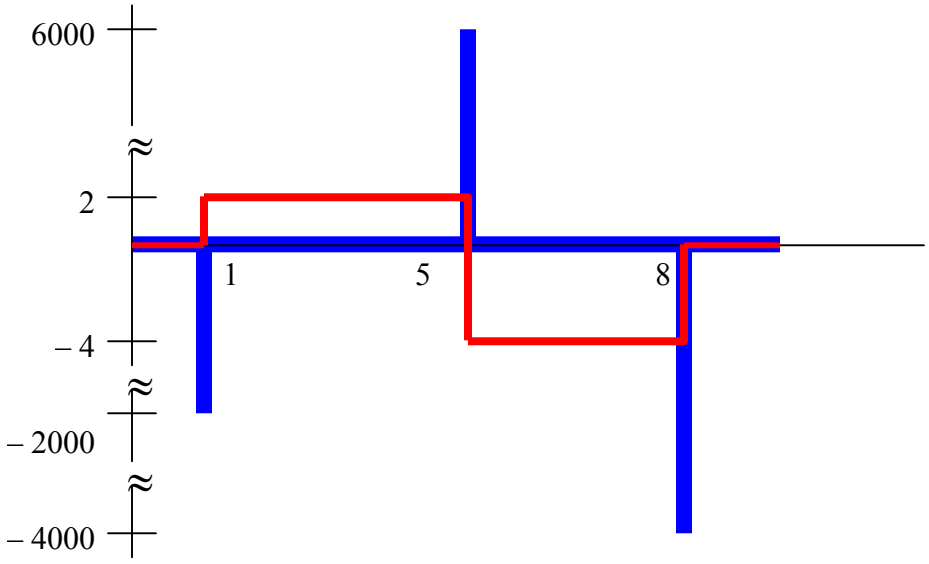



**Figure 8-12. The blue waveform is the output of the Derivative Controller using the non-inverting output. The red waveform corresponds to the input signal**

The peaks demonstrate the behavior of the quick response of the Derivative Controller to the changes of the input signal.




In a real circuit, the maximum value of the peaks is limited by the +Vcc and – Vcc applied to the Derivative Controller.

## Problem Statement

Figure 8-14 is the intended input signal. Notice that this is the same input signal that you used in the Integral Controller in Lab 7. As you did with the integral controller, you need to draw the expected output indicating its amplitude.

The Derivative Controller has a time constant RC = 0.5 ms and uses a supply of ± 12 V.

We assign points at the beginning of each segment as shown in Figure 8-14. These points are: $t_0$, $t_1$, $t_2$, $t_3$ and $t_4$.

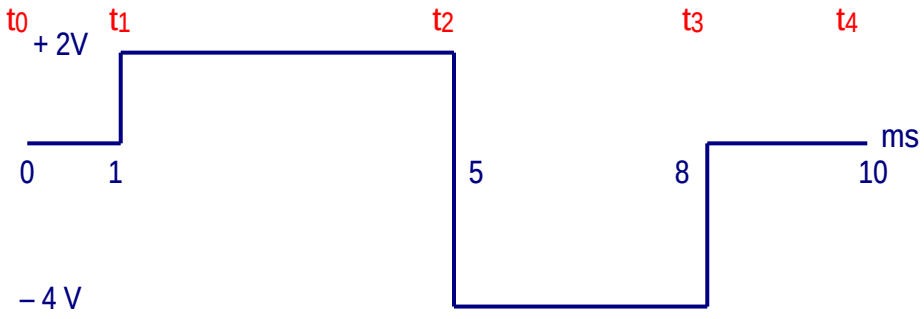



**Figure 8-14. Input waveform applied to a Derivative Controller**

## Expected Output Waveform

The output equation of an inverting derivative controller is:

$$Vo = -RC\frac{dVin}{dt} \qquad (8\text{-}1)$$

A close inspection of Figure 8-14 and using equation 8-1 we notice that the derivative controller will react every time the input signal changes direction.

For example, at 1 ms ($t_1$) the input signal goes from low to high. In this case, the output signal will go from 0 to – Vsat almost instantaneously as shown at point A in Figure 8-15. At 5 ms ($t_2$) the input signal goes from high to low. At this point, the output signal will go from 0 to + Vsat almost instantaneously as shown at point B in Figure 8-15, and so on.

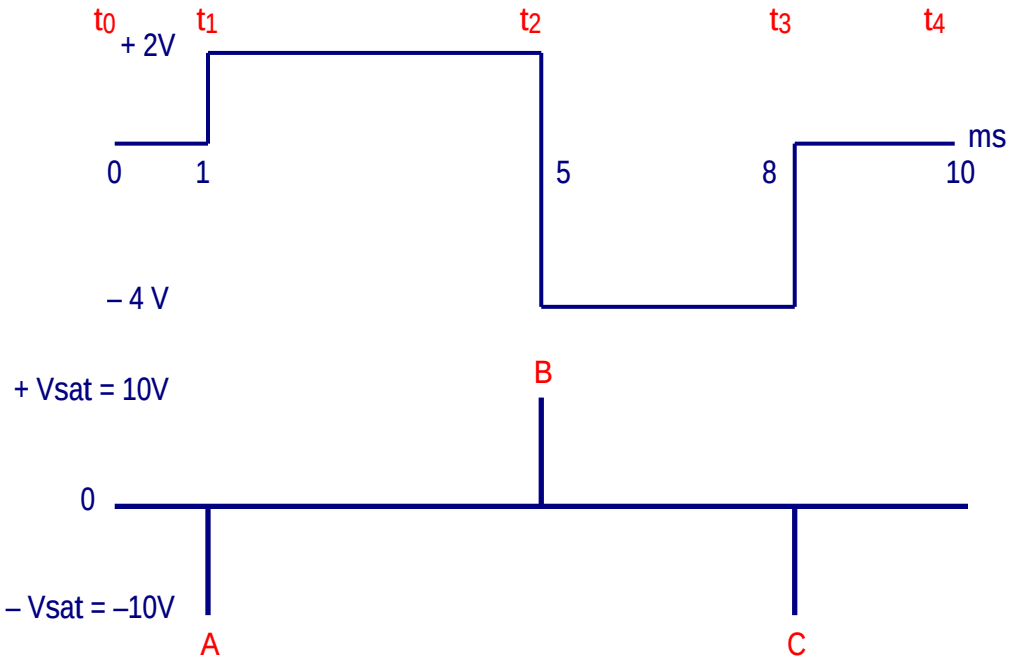



**Figure 8-15. Output waveform of an Inverting Derivative Controller**

With this controller the saturation voltage is ± 10 V (typically, it is 2 V less than the supply voltages).

## Procedure/Tasks Using Multisim

1. Build the circuit shown in Figure 8-16 using Multisim.

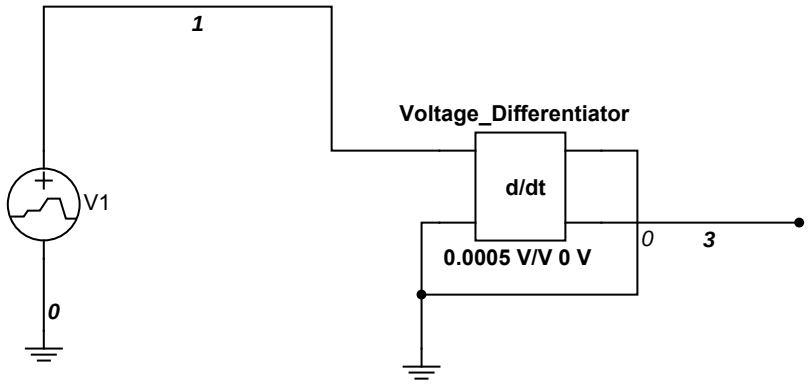



**Figure 8-16. Inverting Derivative Controller**

The Derivative block is the Voltage Differentiator block in Multisim.

## Voltage Differentiator Block

In Multisim you can have an inverting differentiator or a non-inverting differentiator as indicated in Figure 8-17. In this case, we are using an inverting differentiator.

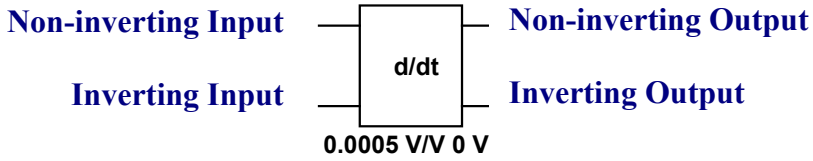



**Figure 8-17. Differentiator block in Multisim**

For the inverting differentiator we can use the non-inverting input and the inverting output or the inverting input and the non-inverting output. In either case, the unused input and output must be grounded. In this laboratory experiment, we'll use the non-inverting input and the inverting output.

The output equation of a differentiator was indicated previously in equation 8-1 and is repeated here for convenience.

$$Vo = -RC\frac{dVin}{dt} \qquad (8\text{-}1)$$

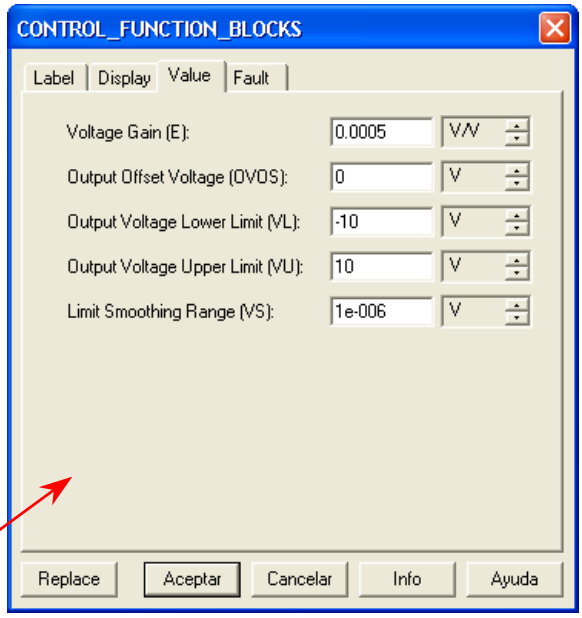
If we make RC = K = Gain then we have:

$$Vo = -K\frac{dVin}{dt} \qquad (8\text{-}2)$$

According to the problem statement RC = 0.5 ms; thus, K = 0.5 ms = 0.0005 (0.0005 V/V in the Differentiator Block).

In addition, the supply is ± 12 V. This means that the saturation voltage is ± 10 V (± 2 V of the supply voltages).

The way we change this saturation voltage is by double-clicking the Differentiator block. When you do this, you will observe the following window:




The Gain of 0.0005 is changed in Voltage Gain (E).

– Vsat is the same as the "Output Voltage Lower Limit (VL)"

+ Vsat is the same as the "Output Voltage Upper Limit (VU)"

Error signal source (V1) is called a Piecewise Linear Voltage (PWL). This signal source produces the type of waveform needed in this lab experiment. This was described previously in Lab 7 – Integrator Controller. Here, we are using the same input waveform.

## Circuit Operation

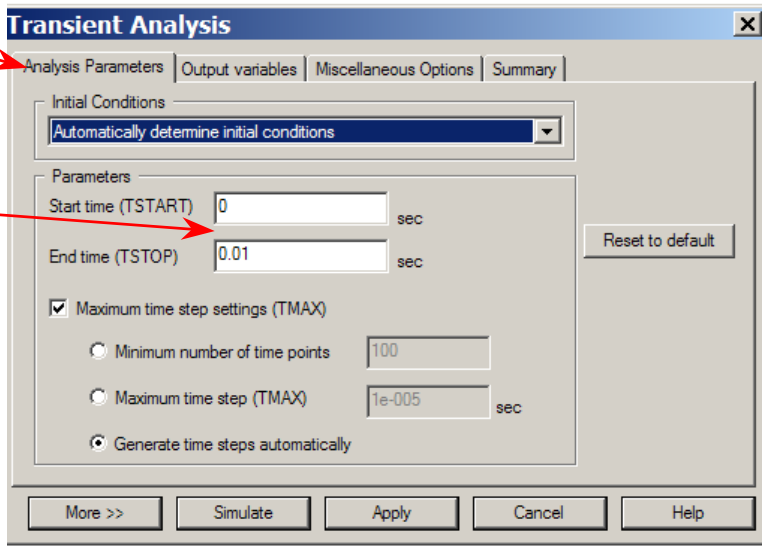
**DO NOT** turn on the power switch. In this lab experiment, you are going to use the Transient Analysis:

**Simulate → Analyses → Transient Analysis**

Make sure that you enter the following parameters:

Click on the Analysis Parameters tab.

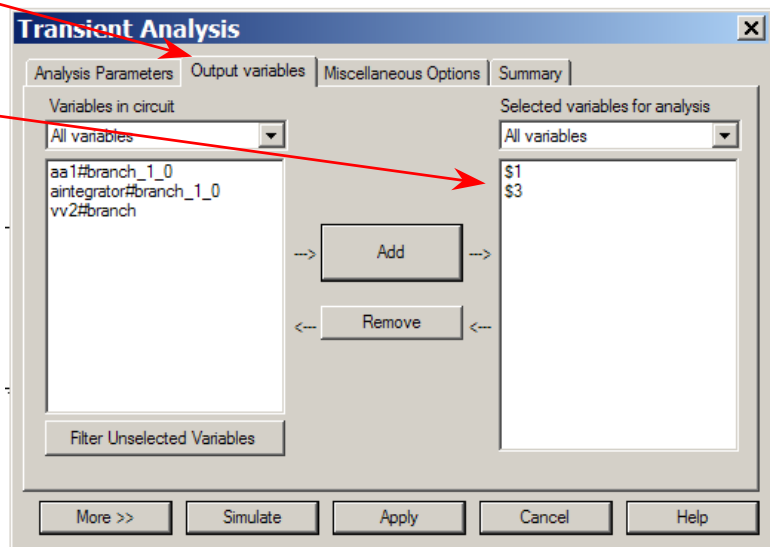
The Start time is set to 0 sec and the End time is set to 0.01 sec (10 msec)




Click on the Output variables tab.

Add nodes 1 (input) and 3 (output).

***Note: These numbers can be different in your circuit depending on the order that the components have been selected. Therefore, make sure that you add the proper nodes for your circuit.***




Next, click on Simulate. You should observe the waveforms depicted in Figure 8-18.

The red waveform is the input and the blue waveform is the output.

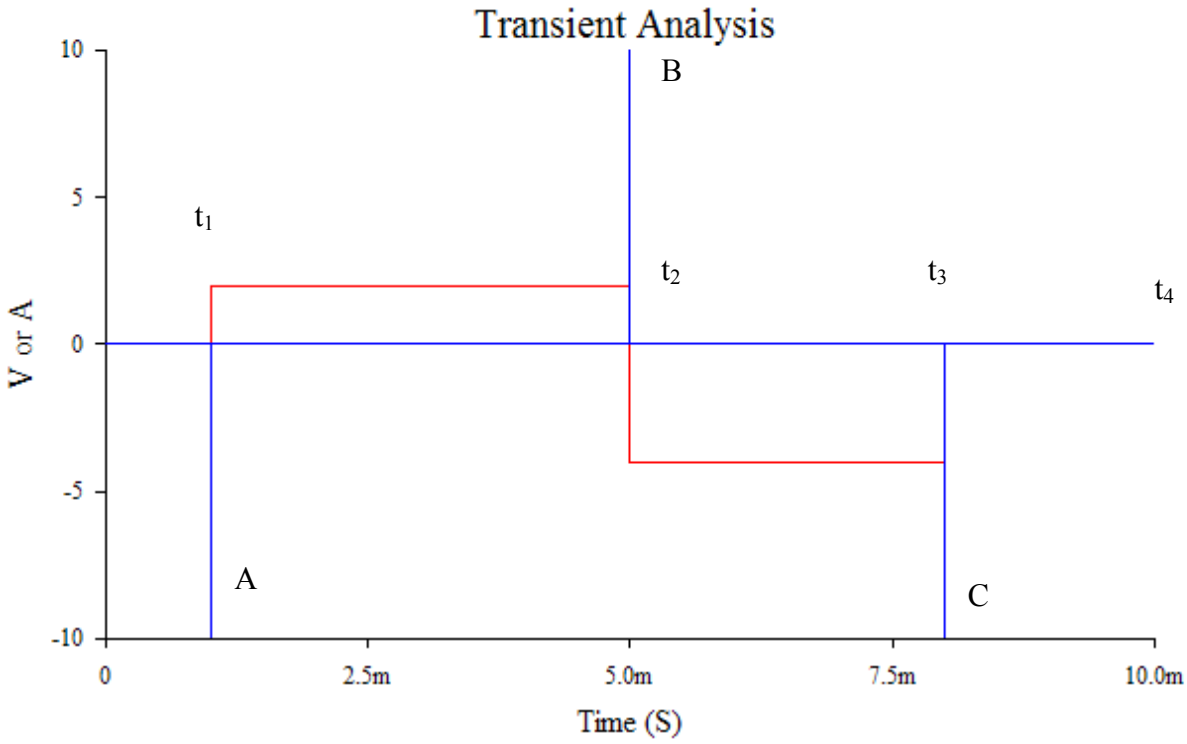



**Figure 8-18. Waveforms of the Inverting Derivative Controller using Multisim**

The red waveform is the input and the blue waveform is the output. In a differentiator, the output reaches saturation every time the input signal goes from 0 to a positive direction or to a negative direction. Since we are using the inverting output, we notice the following:

At $t_1$ the input changes from 0 to + 2 V. The output reaches negative saturation, –10 V in this case (Point A).

At $t_2$ the input changes from + 2 V to – 4 V (negative direction). The output reaches positive saturation, + 10 V in this case (Point B).

At $t_3$ the input changes from – 4 V to 0 V (positive direction). The output reaches negative saturation, – 10 V in this case (Point C).

## Questions

1. Does the output waveform obtained in Multisim match the calculated one? Explain any discrepancies.

2. Rearrange the circuit so that we use the non-inverting output and verify its behavior. What do you observe in the output waveform?

3. Change RC to 2 ms and compare the measured output waveform with the calculated one. What changes (if any) do you observe on the output waveform?

4. The input waveform illustrated in Figure 8-19 is applied to an inverting differentiator with RC = 2 ms and a supply of ± 15 V.

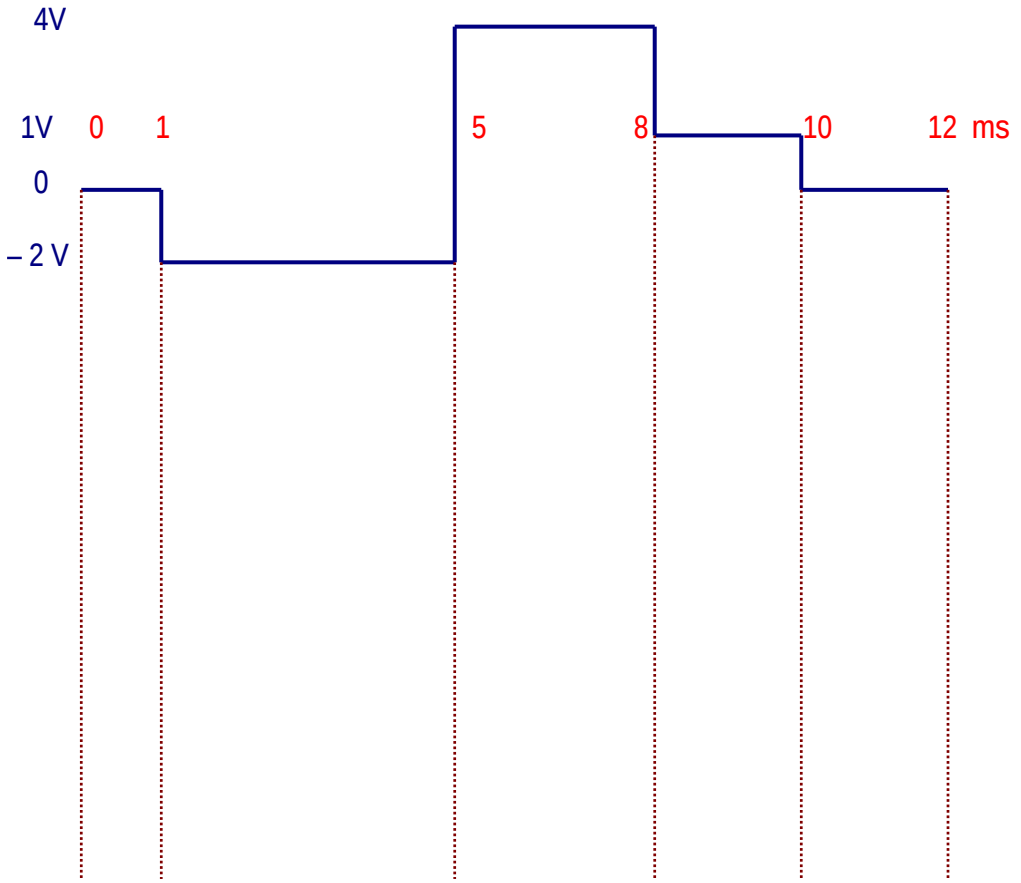



**Figure 8-19. Input waveform for question 4.**

a. Draw the expected output waveform below the input signal in Figure 8-19 indicating its amplitude.

b. Draw the appropriate circuit using Multisim and observe the output waveform. Does the observed output waveform match the one you drew in step 4a? Explain any discrepancies.

5. The input waveform shown in Figure 8-20 is applied to a non-inverting differentiator with RC = 5 ms and a supply of ± 18 V.

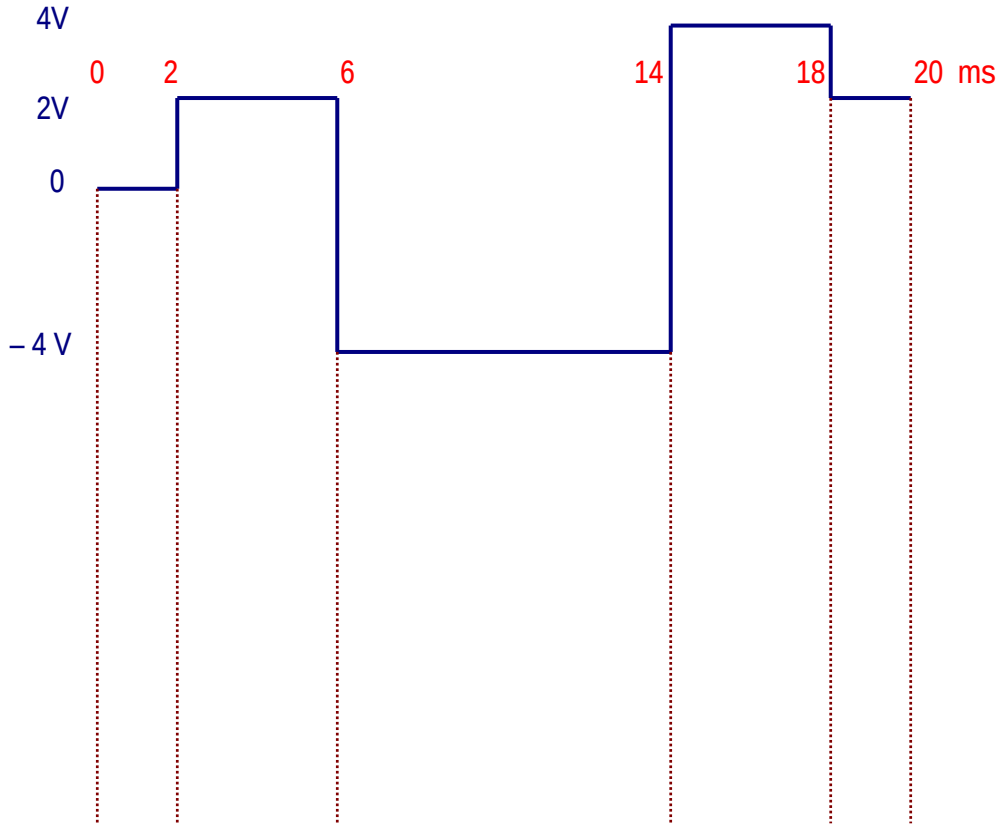



**Figure 8-20. Input waveform for question 5.**

a. Draw the expected output waveform below the input signal in Figure 8-20 indicating its amplitude.

b. Draw the appropriate circuit using Multisim and observe the output waveform. Does the observed output waveform match the one you drew in step 5a? Explain any discrepancies.

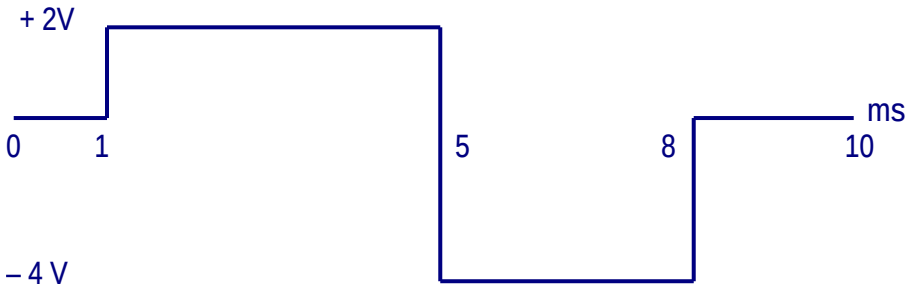
## Conclusions

What are your conclusions regarding the Derivative Controller using Multisim?

## Advanced Design

**If you want to earn extra credit, try the following design.**

The input waveform shown in Figure 8-14 is applied to a Proportional-Derivative (PD) controller. The derivative controller has an RC constant of 1 ms and a supply of ± 16 V and the proportional controller has a gain of –1. Test your solution using Multisim and compare your output waveform with the one obtained in Multisim. For your convenience, Figure 8-14 is shown below.



# LAB 9 – PID CONTROLLER

## Objectives

1. Build, test, and analyze a Proportional-Integral-Derivative (PID) controller using Multisim.
2. Predict the expected output waveform of a PID controller given a specific input waveform and then verify it using Multisim.
3. Design a non-inverting PID controller given a certain time constant and voltage supplies using Multisim.

## Preliminary Information

The **ON/OFF Controller**, also called All/Nothing, works satisfactorily when the process has a slow reaction speed and has a minimum delay time.

When a quick response to the changes in the dependent variables or a precise control of the controlled variable is required, the control ON/OFF is not the best choice.

For example, in a production process employing the continuous use of heat, the valve of the final control should be located in some position in between fully open or fully closed to maintain the temperature at a given reference point or set point. The controller that executes this task is known as a Proportional Controller.

This **Proportional Controller** provides a signal proportional to the error, the greater the error the larger the error signal.

The **Integral Controller** provides a ramp waveform output in response to an accumulated error over time such that the correction of the error is executed progressively.

The **Derivative Controller** reacts quickly to the changes in the process control due to the effect of interferences or forces which cause the process to deviate from a desired set point.

The **PID** or Proportional-Integral-Derivative Controller combines the advantages and disadvantages of each controller allowing correction of the error signal in the most efficient way.

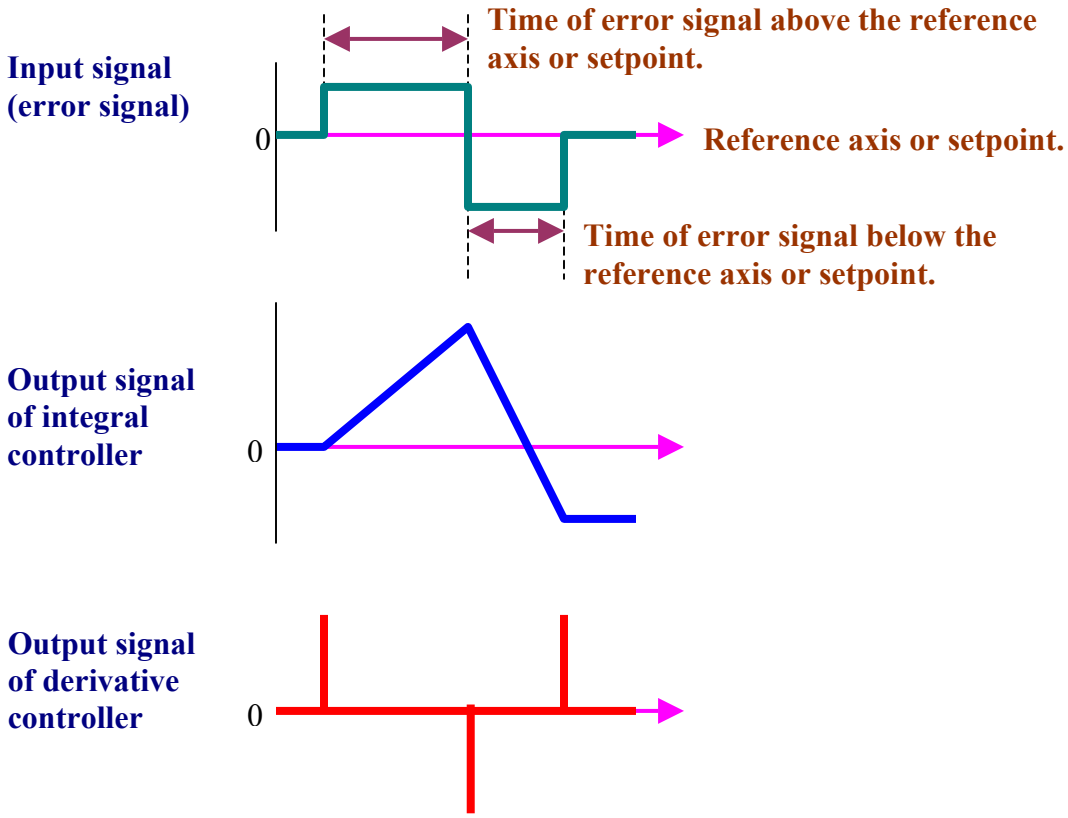



**Figure 9-1. Output signals of each Controller.**

Next, we will analytically determine the output waveform of the PID Controller. We will take into account the following considerations:

1. The three controllers (Proportional, Integral and Derivative) will have unity gain.

2. We will use circuits with non-inverting output to take advantage of the calculations carried out in labs 7 and 8.

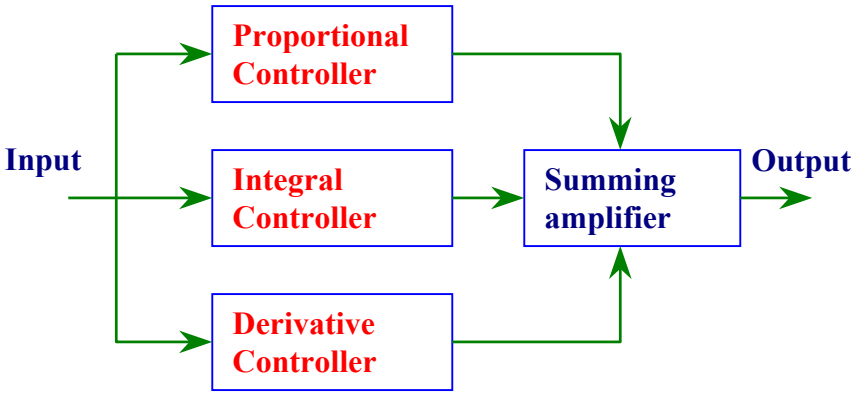



**Figure 9-2. Connection of the PID Controller.**

3. We will connect the controllers as shown in figure 9-2.
   The summing amplifier will be a circuit with unity gain and non-inverting output. The output of the summing amplifier will be the combined actions of the three controllers.

4. The controllers will have a supply of +Vcc = +12V and –Vcc = –12V. Since in the internal circuitry there is a loss of two volts, the maximum positive voltage or +Vsat will be +10V and the maximum negative voltage or –Vsat will be –10V. This implies that the peaks of the output of the Derivative Controller will be limited to +10V to –10V.

5. The input signal is equal to the output of the Proportional Controller because we are using a circuit with unity gain and non-inverting output as seen in figure 9-3.

**Summary of the controllers' waveforms.**

The following summarizes the waveforms taken from the preliminary information of laboratories 7 and 8.

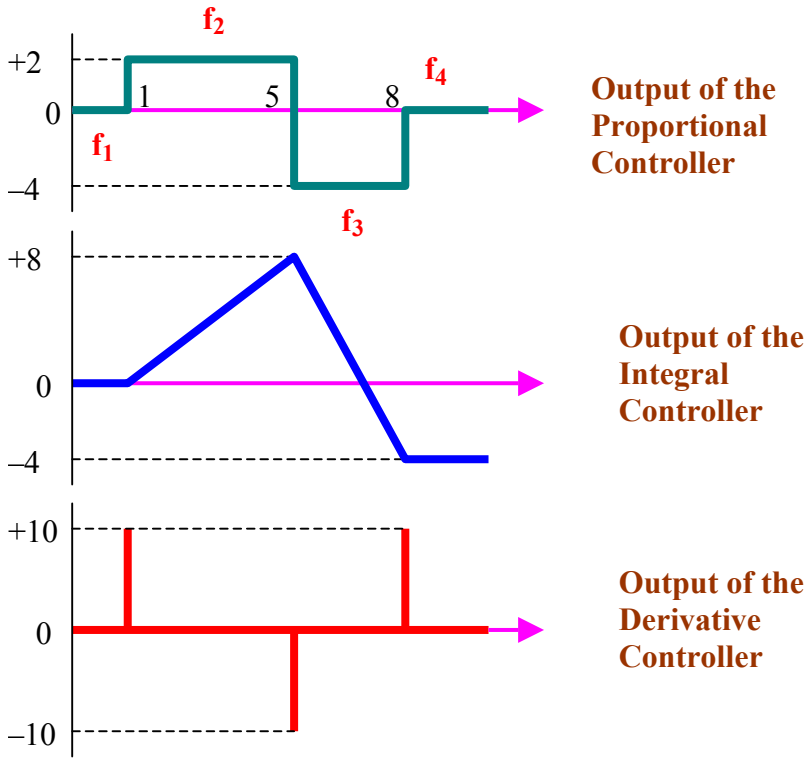



**Figure 9-3. Output signals of each controller for analysis purposes.**

## Waveform of the Proportional Controller

For this analysis, we will use a circuit with unity gain and non-inverting output. Therefore, the output signal of the Proportional Controller will be identical to the input signal. With these explanations, we will proceed to the input signal and its mathematical definition of Lab 7.

Observing the graph of figure 9-4, each function is defined in the following way:

$f_1(x)$ : $y = 0$ , $0 < x < 1$

$f_2(x)$ : $y = 2$ , $1 < x < 5$

$f_3(x)$ : $y = -4$ , $5 < x < 8$

$f_4(x)$ : $y = 0$ , $8 < x$

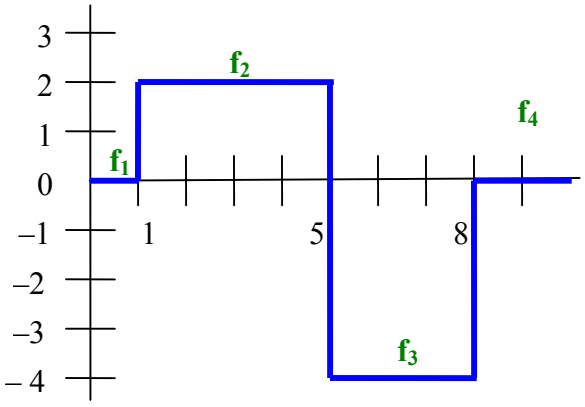



**Figure 9-4. Output of the proportional Controller**

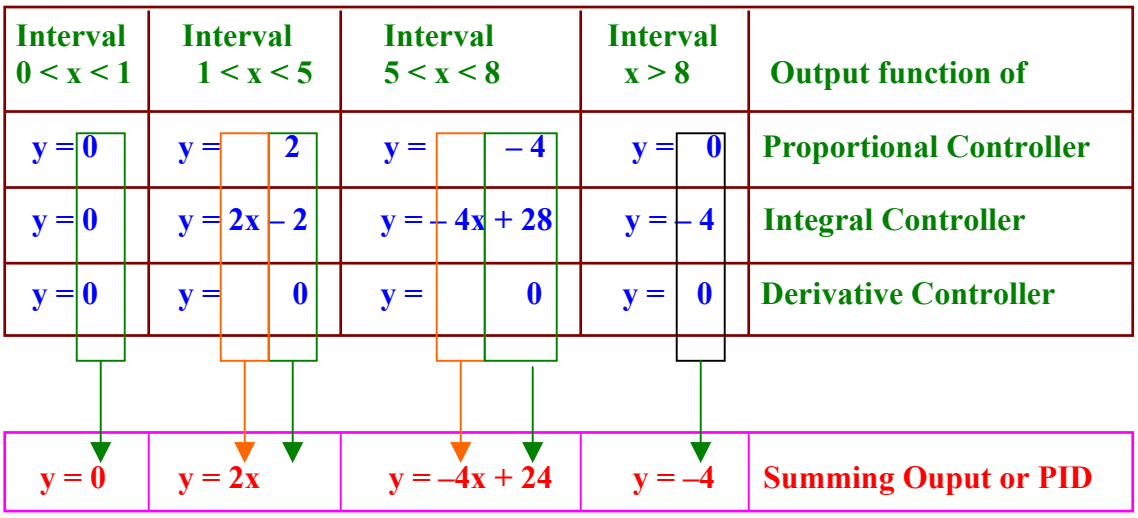
## Summary of the controllers' functions

To find the output functions of the summing amplifier or output of the PID Controller, we separately add the variables and the constants of the functions generated by each controller in each interval. They are presented as follow:

| Interval $0 < x < 1$ | Interval $1 < x < 5$ | Interval $5 < x < 8$ | Interval $x > 8$ | Output function of |
|---|---|---|---|---|
| $y = 0$ | $y = 2$ | $y = -4$ | $y = 0$ | Proportional Controller |
| $y = 0$ | $y = 2x - 2$ | $y = -4x + 28$ | $y = -4$ | Integral Controller |
| $y = 0$ | $y = 0$ | $y = 0$ | $y = 0$ | Derivative Controller |

| | | | | |
|---|---|---|---|---|
| | | | | Summing Ouput or PID |

Algebraically summing the output function of each controller results in the following:




Next, we evaluate these functions with the values of the intervals, to find the initial and final points of their corresponding line.

**Interval 0 < x < 1**

The output function of the PID is: y = 0

For x = 0 → y = 0 → (0, 0) initial point
For x = 1 → y = 0 → (1, 0) final point
} **m line**

**Interval 1 < x < 5**

The output function of the PID is: y = 2x

For x = 1 → y = 2 (1) = 2 → (1, 2) initial point
For x = 5 → y = 2 (5) = 10 → (5, 10) final point
} **n line**

**Interval 5 < x < 8**

The output function of the PID is: y = – 4x + 24

For x = 5 → y = – 4 (5) + 24 = 4 → (5, 4) initial point
For x = 8 → y = – 4 (8) + 24 = – 8 → (8, – 8 ) final point
} **p line**

**Interval x > 8**

The output function of the PID is: y = – 4

For x = 8 → (8, – 4)

These coordinates are the initial point of a line parallel to the x-axis that we will call **q**.

We draw the lines in the calculated intervals that correspond to the partial graph of the PID output.

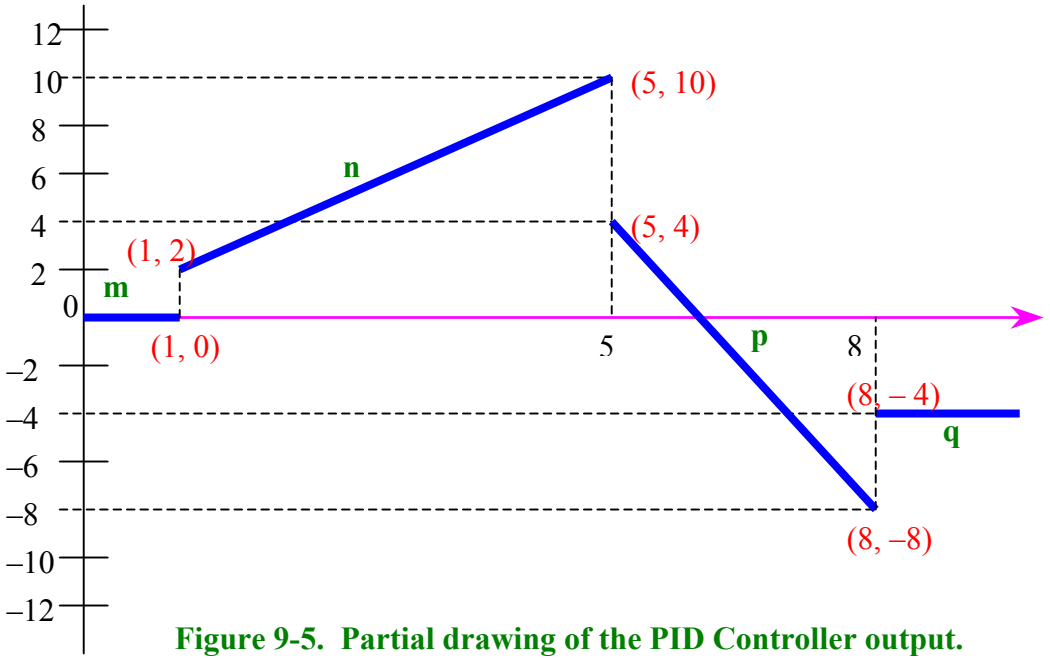



**Figure 9-5. Partial drawing of the PID Controller output.**

To complete the graph we should find the response of the PID Controller at the junctions of the functions, they are at the points: 1, 5, 8 of the x-axis.

Recall in Lab 8 - Derivative controller, the discontinuity of the total controller output function. For that reason, we decided to distort the input signal slightly into three new functions that we designate as: $g_1(x)$, $g_2(x)$, and $g_3(x)$. These functions are defined as follows:

$g_1(x) = 2000x + b$

$g_2(x) = -6000x + b$

$g_3(x) = 4000x + b$

The output functions of the Derivative Controller are:

$$\frac{d}{dx} g_1(x) = \frac{d}{dx}(2000x + b) = 2000 \Rightarrow y = 2000$$

$$\frac{d}{dx} g_2(x) = \frac{d}{dx}(-6000x + b) = -6000 \Rightarrow y = -6000$$

$$\frac{d}{dx} g_3(x) = \frac{d}{dx}(4000x + b) = 4000 \Rightarrow y = 4000$$

At point 3 of the considerations outlined previously, we said the controllers have a supply of +Vcc = +12V and –Vcc = –12V. However, due to the loss in the internal circuitry, the outputs were limited to +Vsat = +10V and –Vsat = –10V. Therefore, the functions of the derivative controller: $g_1(x)$, $g_2(x)$, and $g_3(x)$ will be redefined with these value limits in the following way:

$$y = 10$$
$$y = -10$$
$$y = 10$$

The output functions of the Integral Controller from the input signal function are:

$$\int f_2(x)dx = \int 2\,dx = 2x + b$$

$$\int f_3(x)dx = \int -4\,dx = -4x + b$$

$$\int f_4(x)dx = \int 0\,dx = 0$$

We will evaluate each one of these functions to find their corresponding value of the b constant. To accomplish this we will use the graph of figure 9-6 taken from Lab 7 that corresponds to the output of the Integral Controller.

For $\int f_2(x)dx$ : $y = 2x + b$

Calculating b for coordinates (1, 0) of the junction of the straight lines a and b

$$y = 2x + b \;\Rightarrow 0 = 2(1) + b \;\Rightarrow\; b = -2$$

Then **y = 2x – 2**

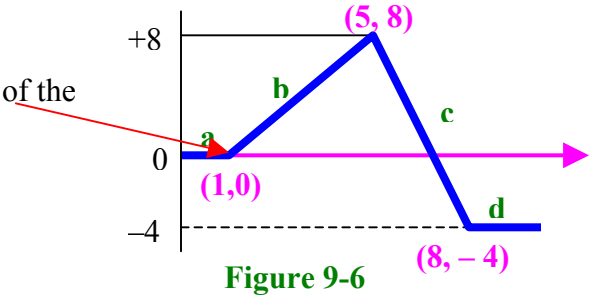


**Figure 9-6**

For $\int f_3(x)dx$ : $y = -4x + b$

Calculating b for coordinates (5, 8) of the junction of the straight lines b and c

$$y = -4x + b \;\Rightarrow 8 = -4(5) + b \;\Rightarrow\; b = 28$$

Then **y = – 4x + 28**

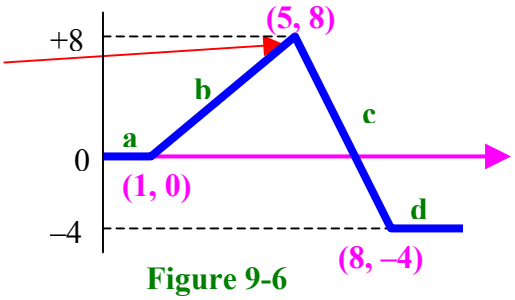


**Figure 9-6**

For $\int f_4(x)dx = \int 0\,dx = 0$ : $y = 0$

Summarizing, the output functions of the Integral Controller with b constant are:

$y = 2x - 2$
$y = -4x + 28$
$y = 0$

## Graph of the output response of the PID controller at points 1, 5, 8

In figure 9-7 below figure 9-5 is repeated with the addition of the output signal of the Proportional Controller for analysis purposes.

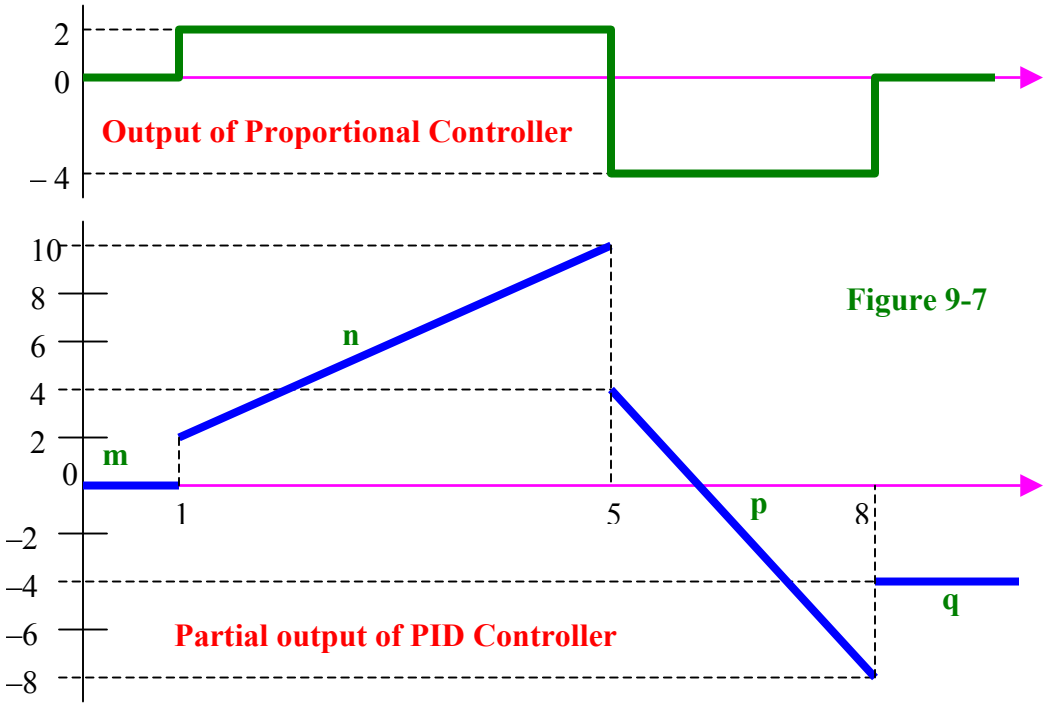



Figure 9-7

## Junction of lines m, n

In figure 9-7 PID Controller output, we join line m at point (1, 0) to line n at point (1, 2) with an additional line. Following point (1, 2) the output function of the Proportional Controller is y = 2.

At the point (1, 0) we will calculate the coordinates of the following points by adding the output functions of the three controllers. They are:

| | |
|---|---|
| $y = 2$ | **Output of the Proportional Controller** |
| $y = 2x - 2$ | **Output of the Integral Controller** |
| $y = 10$ | **Output of the Derivative Controller** |
| $y = 2x + 10$ | **Output of the summing amplifier or PID output** |

At the point (1,0). Replacing values for x = 1

$y = 2x + 10 = 2\,(1) + 10 = 12$ → the final point is: (1, 12)

The junction of the lines m and n will be obtained by drawing a line that we will call u whose initial and final points are (1, 0) and (1, 12) respectively. In a real circuit, the maximum value of the peaks is limited by the +Vcc and – Vcc applied to the PID Controller.

### Junction of the lines n, p

In figure 9-7 PID Controller output, we join line n at point (5, 10) to line p at point (5, 4) with an additional line. Following point (5, – 4) the output function of the Proportional Controller is y = – 4

At the point (5, 10) we will calculate the coordinates of the next point by adding the output functions of the three controllers, in the following way:

| | |
|---|---|
| $y = -4$ | **Output of the Proportional Controller** |
| $y = -4x + 28$ | **Output of the Integral Controller** |
| $y = -10$ | **Output of the Derivative Controller** |
| --- | |
| $y = -4x + 14$ | **Output of the summing amplifier or PID controller** |

At the point (5, 10). Substituting x = 5 into the above equation gives the following:

$y = -4x + 14 = -4\,(5) + 14 = -6$ → therefore, the following point is: (5, – 6)

The junction of lines n, p will be obtained by drawing a line that we'll call **v** whose initial and final points are (5, 10) and (5, – 6) respectively.

### Junction of lines p, q

In figure 9-7 PID Controller output, we join line p at point (8, – 8) to line q at point (8, – 4) with an additional line. Following point (8, – 4) the output function of the Proportional Controller is y = 0.

Then, at the point (8, –8) we will calculate the coordinates of the next point by adding the output functions of the three controllers, in the following way:

| | |
|---|---|
| $y = 0$ | **Output of the Proportional Controller** |
| $y = 0$ | **Output of the Integral Controller** |
| $y = 10$ | **Output of the Derivative Controller** |
| --- | |
| $y = 10$ | **Output of the summing amplifier or PID controller** |

At the point (8, –8).

y = 10 → Therefore, the following point is: (8, 10)

The junction of the lines p, q will be obtained by drawing a line that we'll call w whose initial and final points are (8, – 8) and (8, 10) respectively.

Adding calculated lines u, v, w to figure 9-5 results in the complete output response of the PID Controller as shown in figure 9-8.

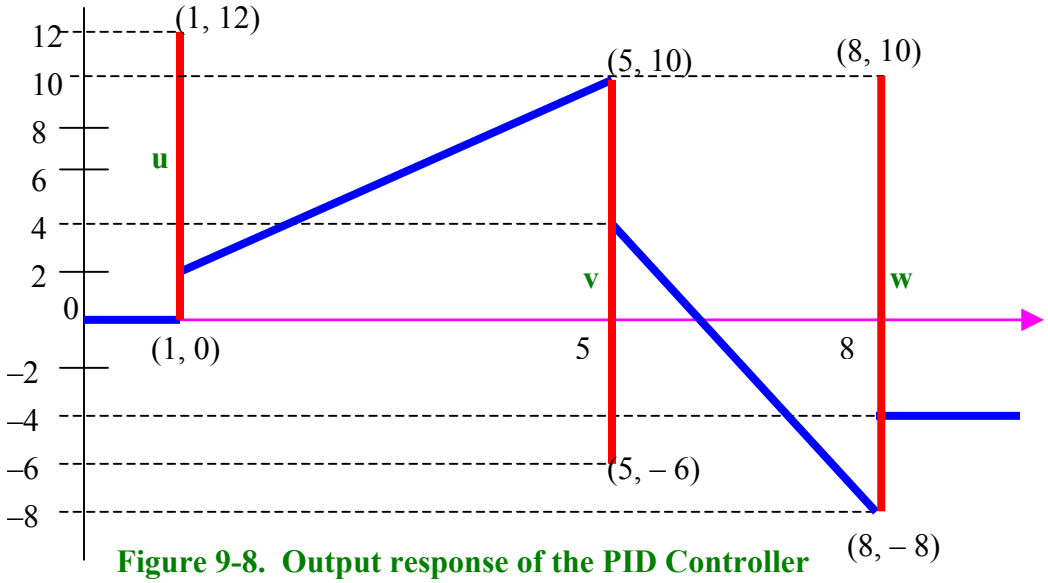



**Figure 9-8. Output response of the PID Controller using circuits with non-inverting outputs.**

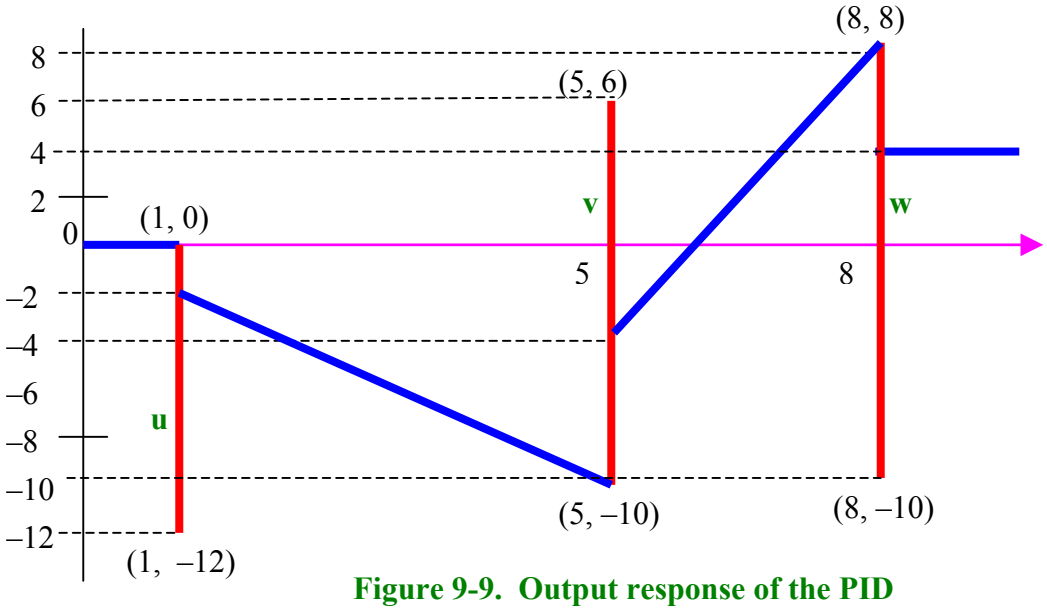



**Figure 9-9. Output response of the PID Controller using circuits with inverting outputs.**

## Problem Statement

A controller input error signal is shown in Figure 9-10. Notice that this is the same input signal that was used in the Integral Controller and the Derivative Controller, Labs 7 and 8 respectively. Design a PID Controller with the following specs:

Proportional controller: Gain = –1

Integral Controller: RC = 1 ms

Non-inverting Derivative Controller: RC = 0.5 ms

Summing Amplifier: Gain = –1

Voltage supply = ± 12 V

All three stages must be in parallel

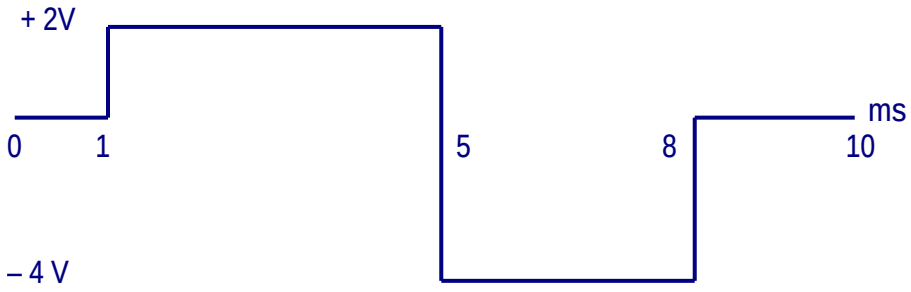



**Figure 9-10. Input waveform applied to a PID Controller**

## Solution

According to the problem statement, the block diagram of the PID controller is as indicated in Figure 9-11

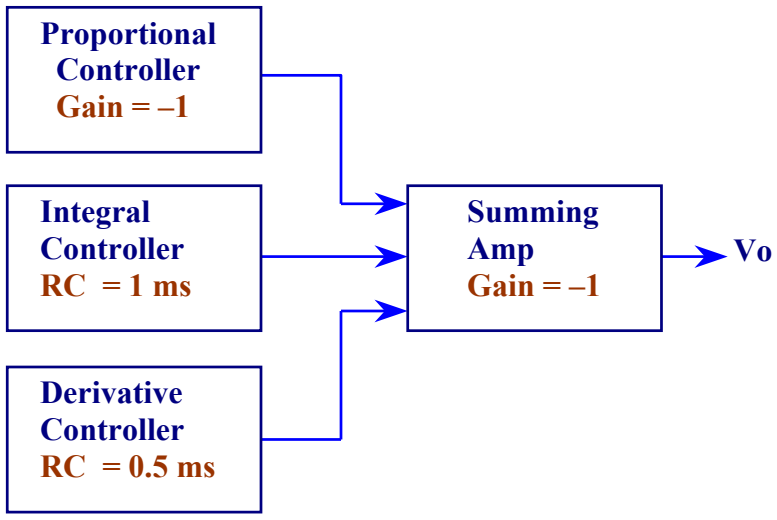



**Figure 9-11. Block diagram of the PID controller.**

You are already familiar with the Integral Controller and the Derivative Controller. The Proportional Controller is simply an inverting amplifier. The statement of the problem calls for a gain of –1. The summer adds the outputs of the three controllers and has a gain of –1.

Figure 9-12 shows the input waveform, the outputs of the three controllers and the output of the PID controller. The analysis of figure 9-12 is as follows.

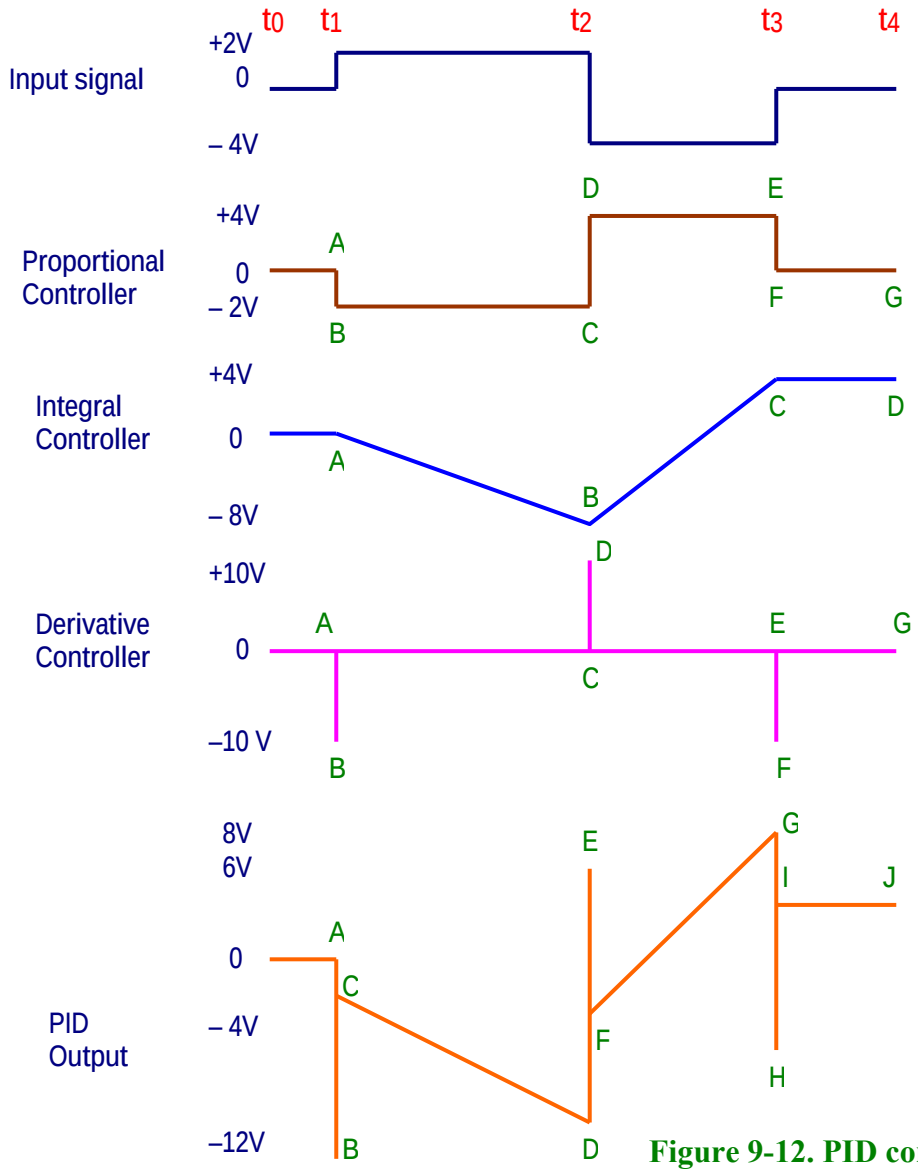



**Figure 9-12. PID controller output**

Since the gain of the proportional controller is –1 its output signal is the inverted equivalent of the input signal. The output of the integral controller was determined in Lab 7 and the output of the derivative controller was determined in Lab 8. For illustration purposes, we have established reference points on each of the output signals of the three controllers. These reference points are shown in order beginning from left to right. For example, let us start with the output from the proportional controller. At time $t_1$ we have two voltage levels: 0 V (point A) and –2 V (point B). At time $t_2$ we have again two voltage levels: –2 V (point C) and 4 V (point D). At time $t_3$ we

have two voltage levels: + 4 V (point E) and 0 V (point F). At time $t_4$ we have only one voltage level: 0 V (point G).

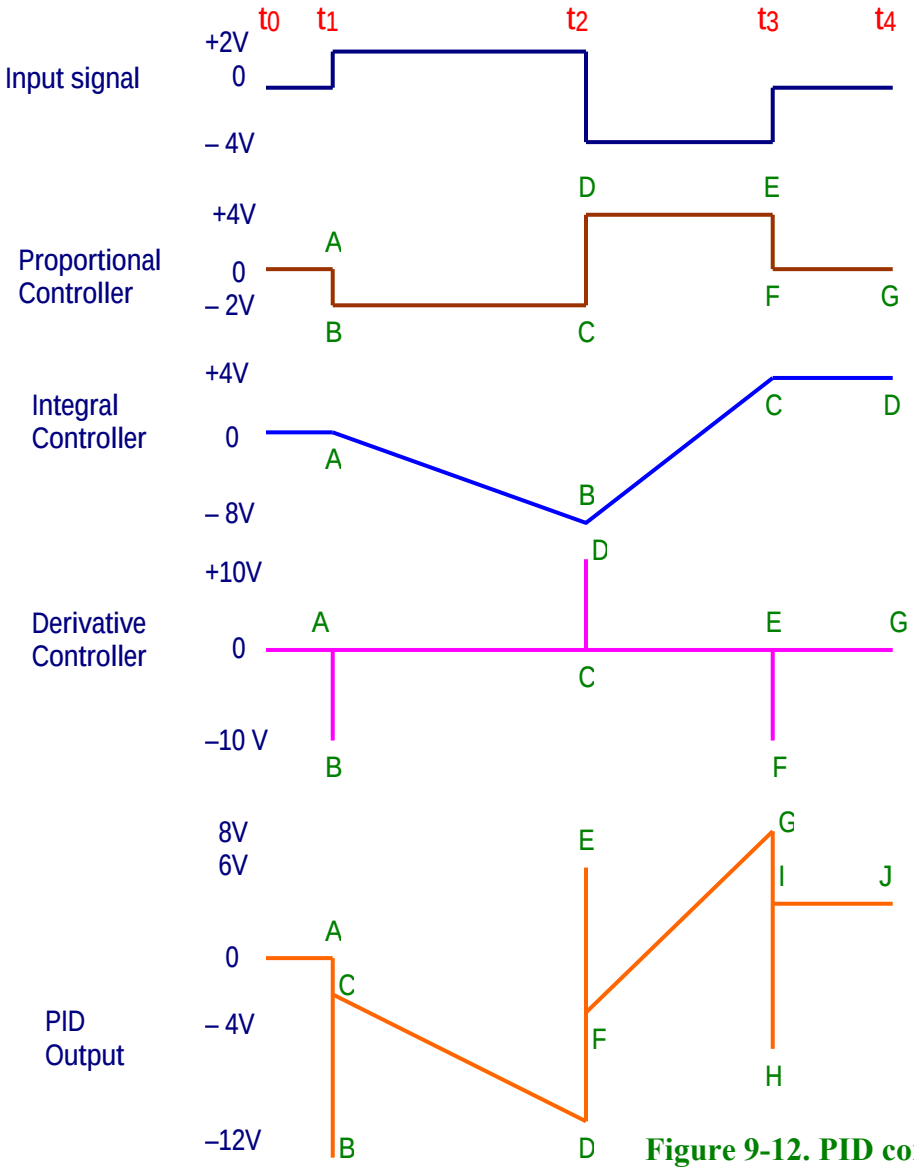



**Figure 9-12. PID controller output**

Follow the same procedure for the outputs of the Integral Controller and the Derivative Controller. The output of the PID controller is calculated next.

**PID Point A** = Prop Point A + Integral Point A + Derivative Point A = 0 + 0 + 0 = 0 V

**PID Point B** = Prop Point B + Integral Point A + Derivative Point B = –2 + 0 + (–10) = –12 V

**PID Point C** = Prop Point B + Integral Point A + Derivative Point A = –2 + 0 + 0 = – 2 V

***Note:*** ***The output of the Derivative controller goes to – 10 V at point B and then quickly returns to 0 V at point A again.***

**PID Point D** = Prop Point C + Integral Point B + Derivative Point C = - 2 + (– 8) + 0 = –10 V

**PID Point E** = Prop Point D + Integral Point B + Derivative Point D = + 4 + (– 8) + 10 = + 6 V

**PID Point F** = Prop Point D + Integral Point B + Derivative Point C = + 4 + (– 8) + 0 = – 4 V

***Note:** The output of the Derivative controller goes to + 10 V at point D and then quickly returns to 0 V at point C again.*

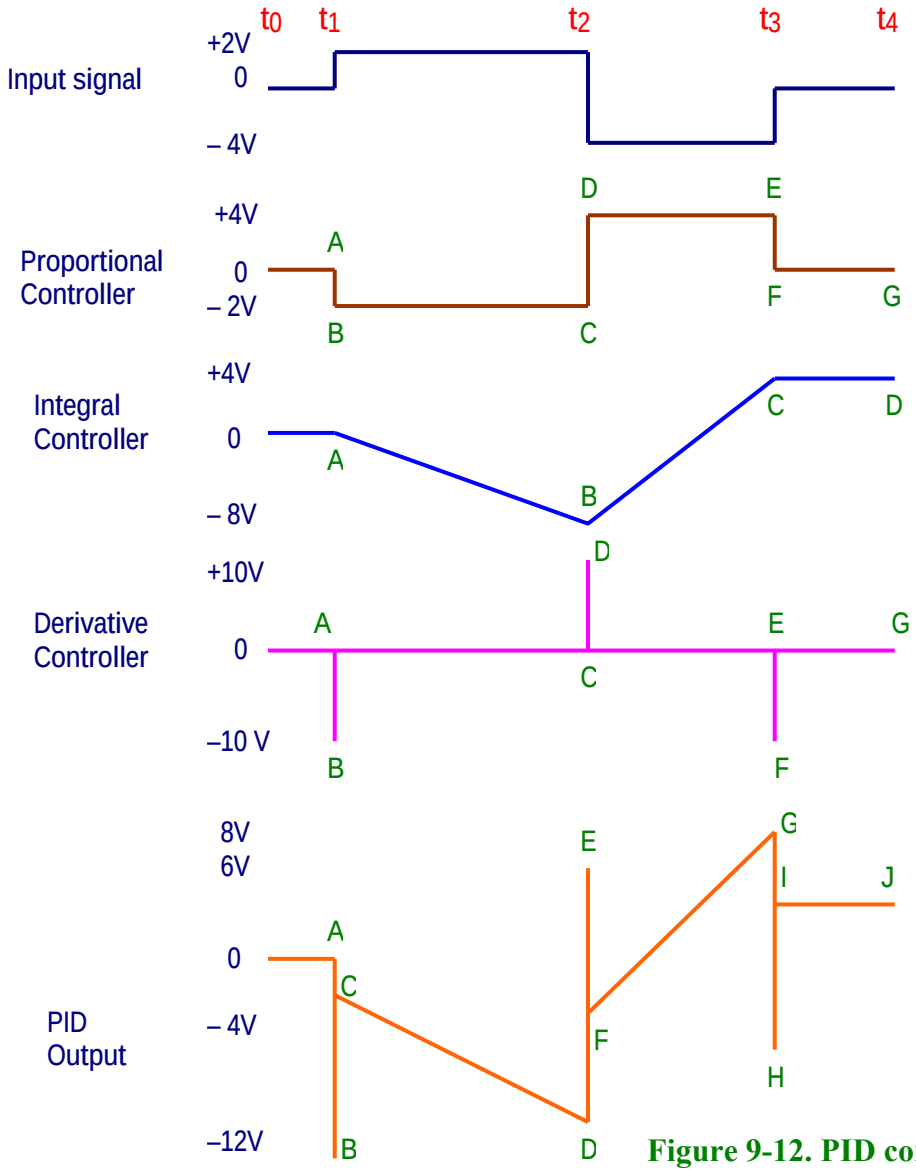



**Figure 9-12. PID controller output**

**PID Point G** = Prop Point E + Integral Point C + Derivative Point E = + 4 + 4 + 0 = + 8 V

**PID Point H** = Prop Point F + Integral Point C + Derivative Point F = 0 + 4 –10 = – 6 V

**PID Point I** = Prop Point F + Integral Point C + Derivative Point E = 0 + 4 + 0 = + 4 V

***Note:** The output of the Derivative controller goes to –10 V at point F and then quickly returns to 0 V at point E again.*

**PID Point J** = Prop Point G + Integral Point D + Derivative Point G = 0 + 4 + 0 = + 4 V

## Procedure/Tasks Using Multisim

1. Build the circuit shown in Figure 9-13 using Multisim.

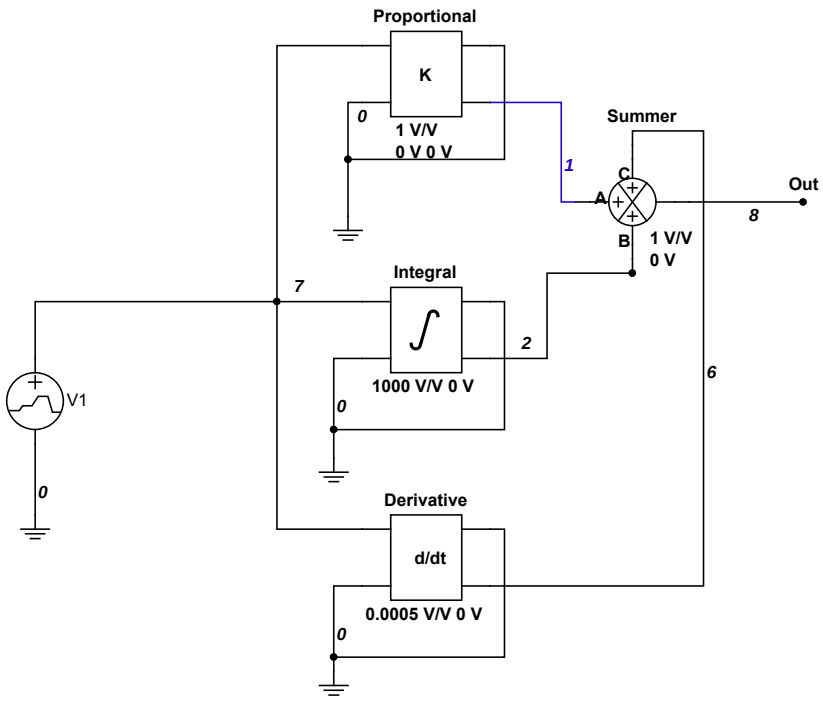



**Figure 9-13. PID controller using Multisim**

We discussed the Integrator controller and the Derivative controller in Labs 7 and 8 respectively.

**As a review:**

The output equation of an integrator is Vout = –V*t/RC + C.

Or Vout = – K * V * t + C, where K = Gain = 1/RC.

The design calls for RC = 1 ms; thus, K = 1/RC = 1/0.001 s = 1000 (1000 V/V in the Integrator Block).

The Derivative block is the Voltage Differentiator block in Multisim.

The output equation of a Differentiator is Vout = –RC dVin/dt

Or Vout = –K * dVin/dt, where K = Gain = RC.

The design calls for RC = 0.5 ms; thus, K = 0.5 ms = 0.0005 (0.0005 V/V in the Differentiator Block).

The proportional controller is simply an inverting amplifier. The statement of the problem calls for a gain of –1. This is achieved by setting the gain to 1 V/V and using the signal from the inverting output as shown in Figure 9-14. The inverting input and non-inverting output are grounded.

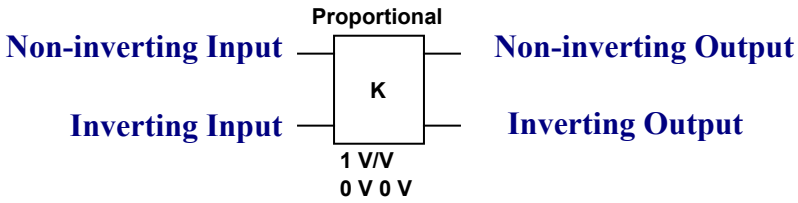



**Figure 9-14. Proportional block in Multisim**

The PID output is obtained by summing the outputs of the three controllers: proportional, integral and derivative. The function block that performs this sum is called a summer in Multisim. Notice that the inputs to this summer block are nodes 1, 2 and 6 and the output is obtained at node 8 (Out). The gain of this summer is 1 or 1 V/V.

The supply is ± 12 V. This means that the saturation voltage is ± 10 V (2 V less than the supply voltages).

## Circuit Operation

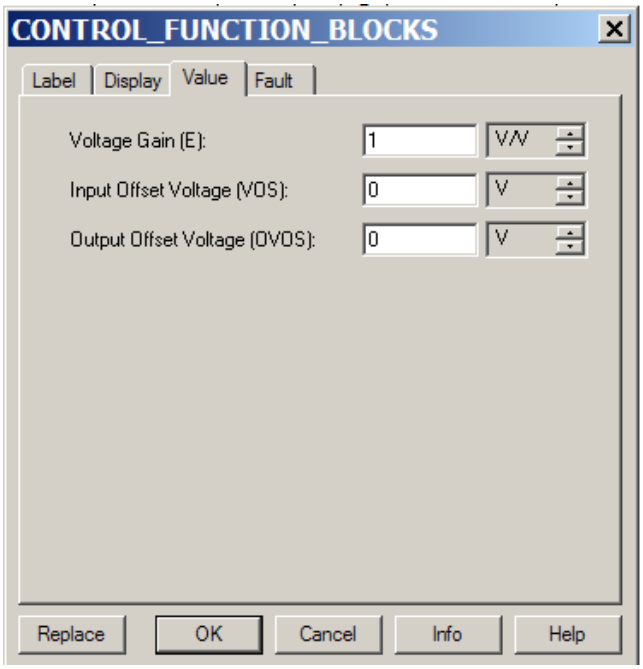
**DO NOT** turn on the power switch. In this lab experiment, you are going to use the Transient Analysis:

**Simulate → Analyses → Transient Analysis**

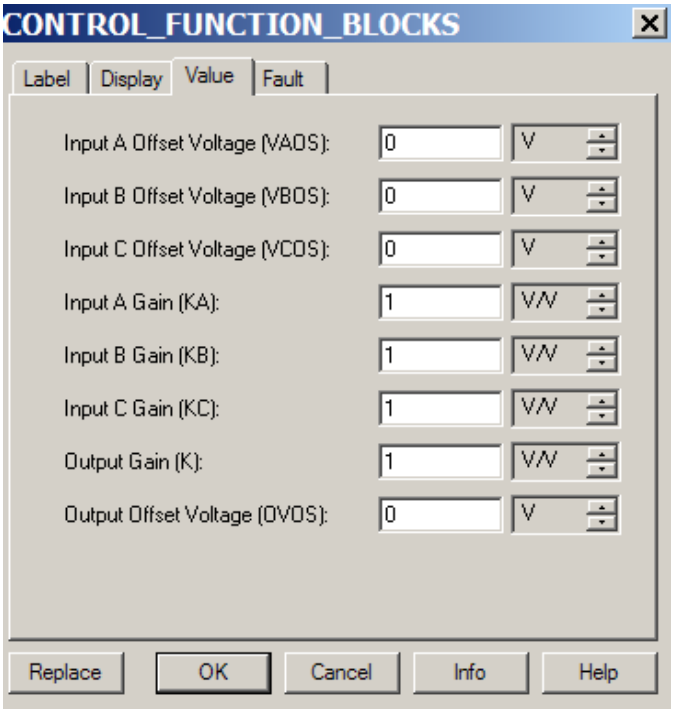
Follow the procedure outlined in Labs 7 and 8 to setup the gain of the Integral block and the Derivative block.

Set the gain of the Proportional block as follows: double-click on the Proportional block and click on the Value tab.

Set the Voltage Gain (E) to 1 V/V.

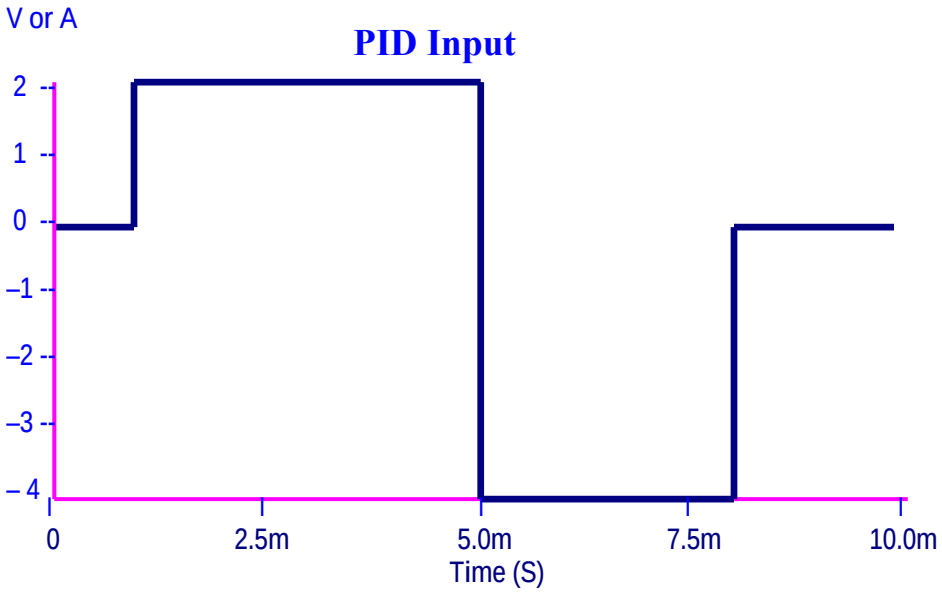


Now double-click on the summer block and click on the Value tab. Set the values as shown in the figure below.

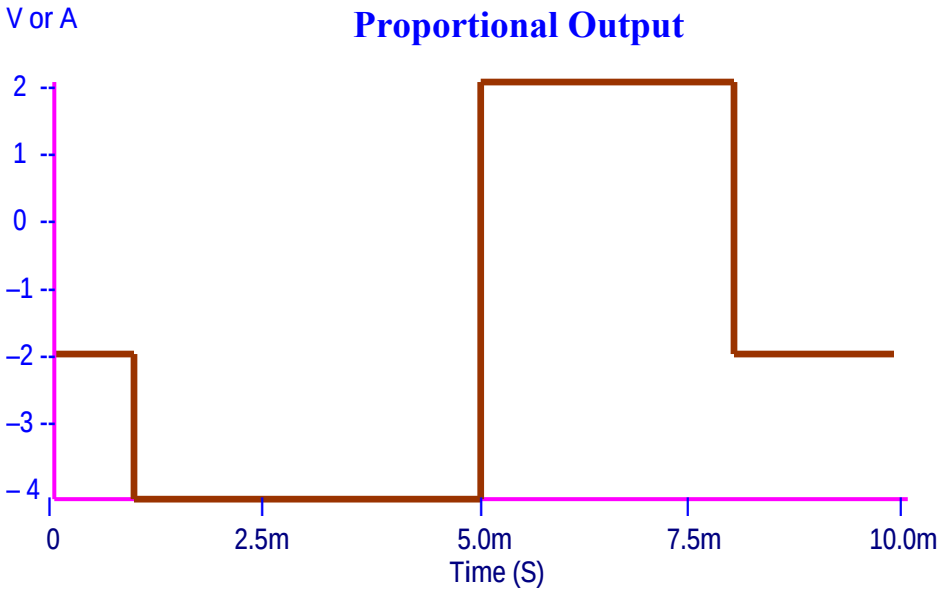



Using the procedure for Transient Analysis as indicated in labs 7 and 8 you will obtain the following outputs for each controller and the PID output.

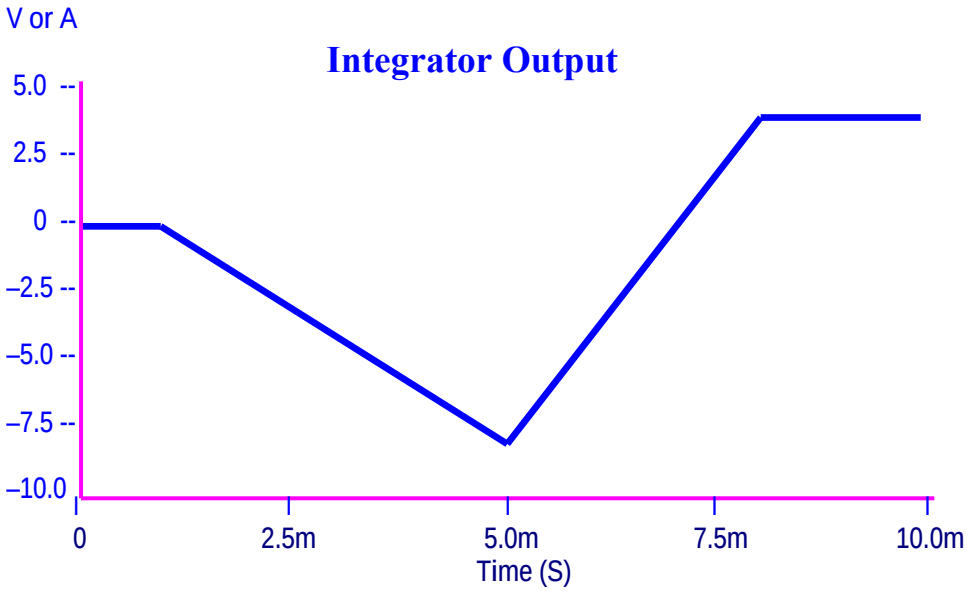
The PID input is the same as the error input signal used in labs 7 and 8.

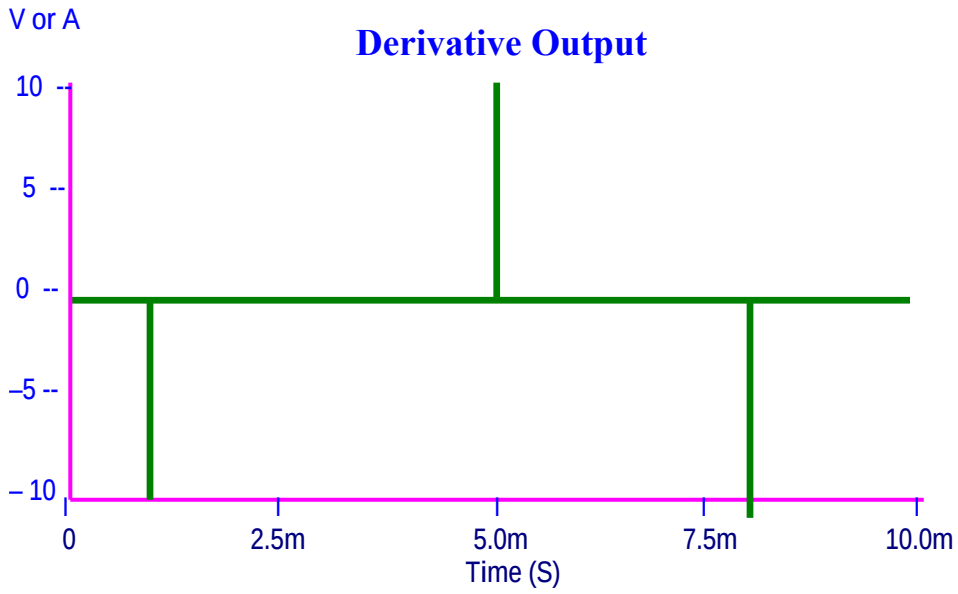


The output of the proportional controller is depicted below. Notice that the output is exactly the opposite of the PID input

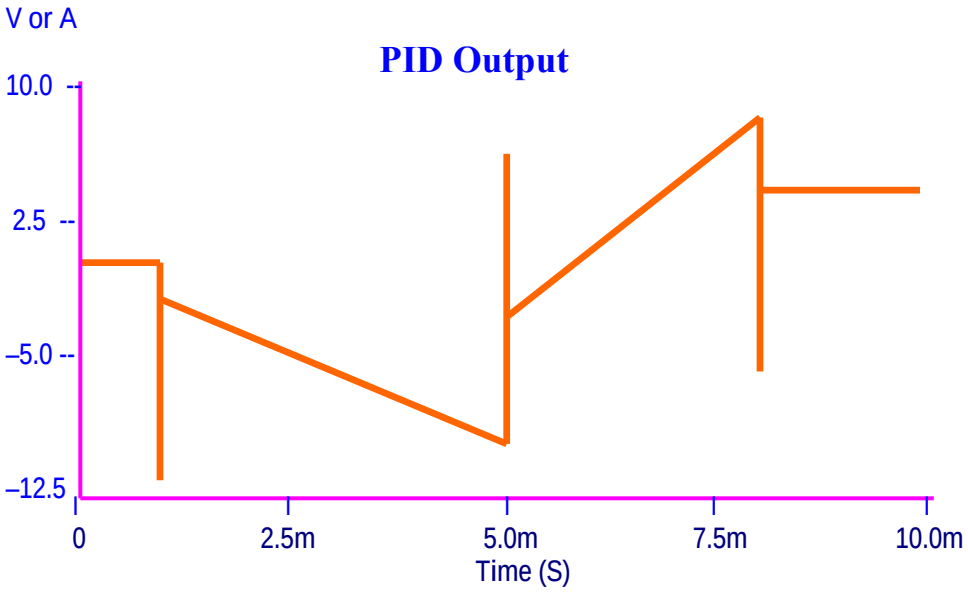



The outputs of the Integral block and Derivative block are shown below. They are exactly the same as you saw in labs 7 and 8.






Finally, the outputs of the three controllers are added in the summer block. The resulting output is depicted below.



The addition of the controllers' outputs and the PID output can be easily understood by connecting a 4-channel oscilloscope as shown in figure 9-15.

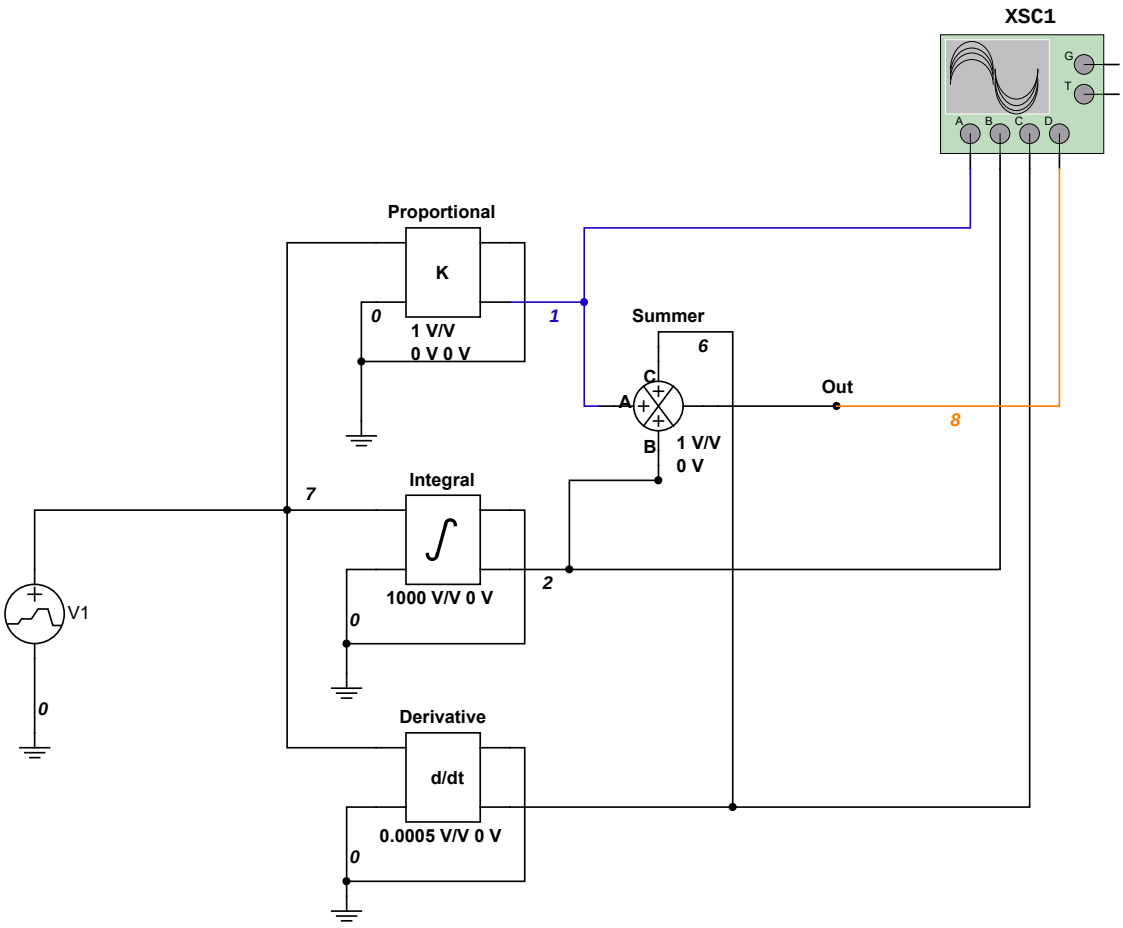
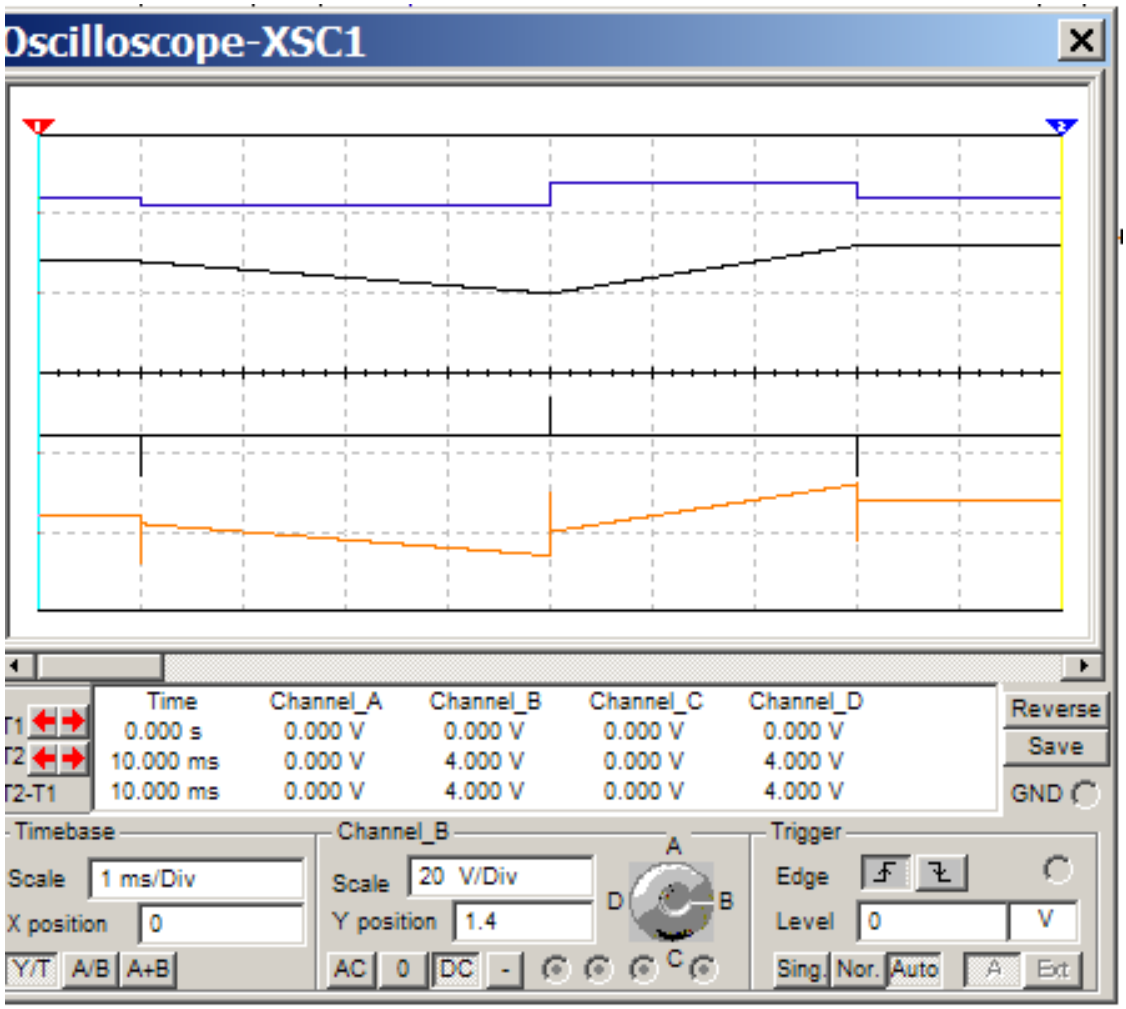



**Figure 9-15. PID controller outputs observed using an oscilloscope.**

The three controllers' outputs and the PID output will be as seen below.




The top signal is the proportional output, the second from the top signal is the integrator output, and the third signal from the top is the derivative output. Finally, the bottom signal is the PID output. A close inspection of the oscilloscope settings shows that the time base scale is 1 ms/Div and the vertical scale has been set to 20 V/Div.

## Questions

1. Does the output waveform of the PID controller obtained in Multisim match the calculated one? Explain any discrepancies.

2. Change the gain of the summer to –1 in Figure 9-13. Draw the expected waveform in Figure 9-16 and compare it with the one obtained in Multisim.

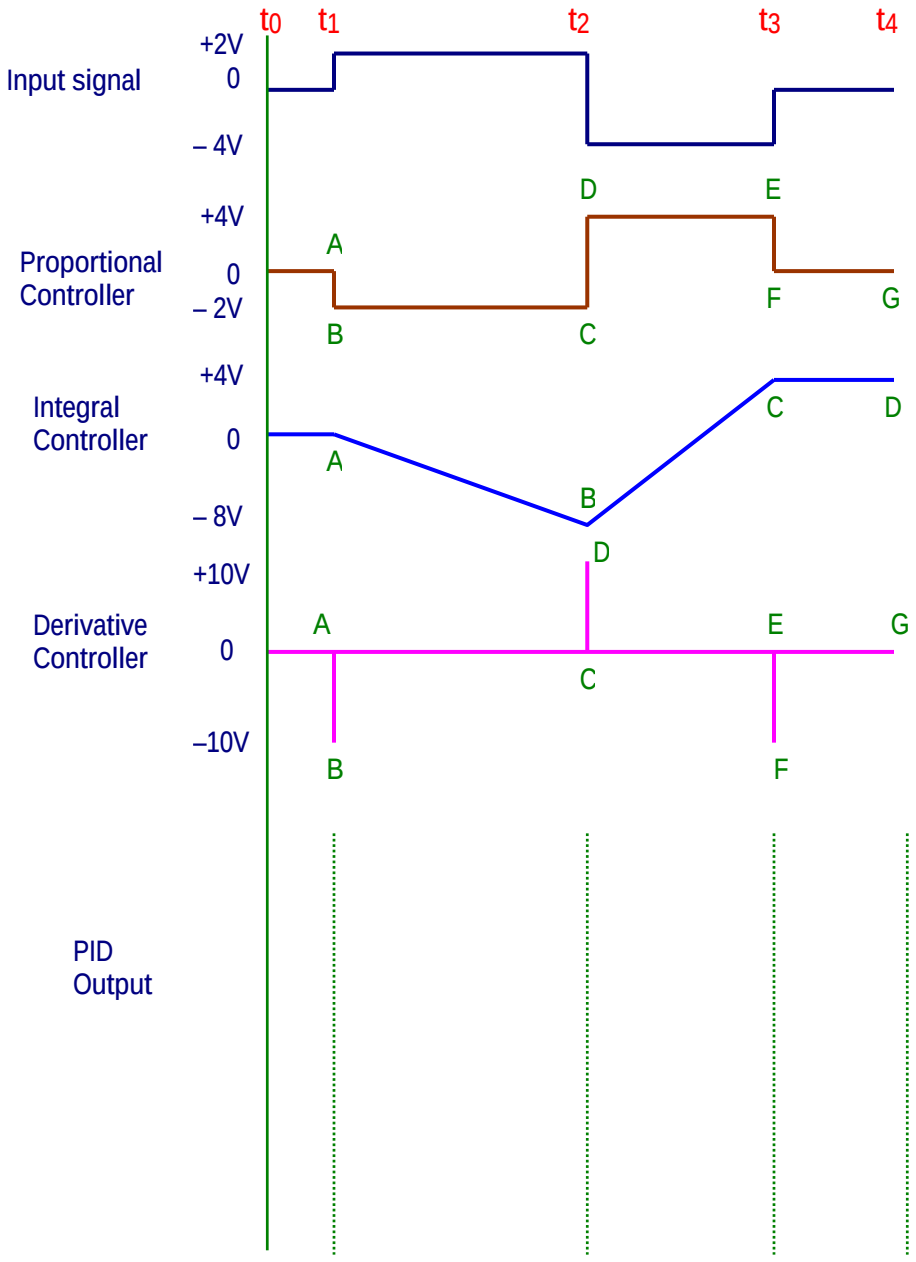



**Figure 9-16. PID output with summer gain = –1.**

## Conclusions

What are your conclusions regarding the PID Controller using Multisim?

## Advanced Design

**If you want to earn extra credit, try the following design.**

The input signal shown in Figure 9-10 is applied to a PID controller with the following specs:

Proportional controller: Gain = 2
Integral Controller: RC = 2 ms
Derivative Controller: RC = 1 ms
Voltage supply = ± 15 V
All three stages must be in parallel

a. Draw the expected PID output waveform in Figure 9-17.

b. Verify your design using Multisim.

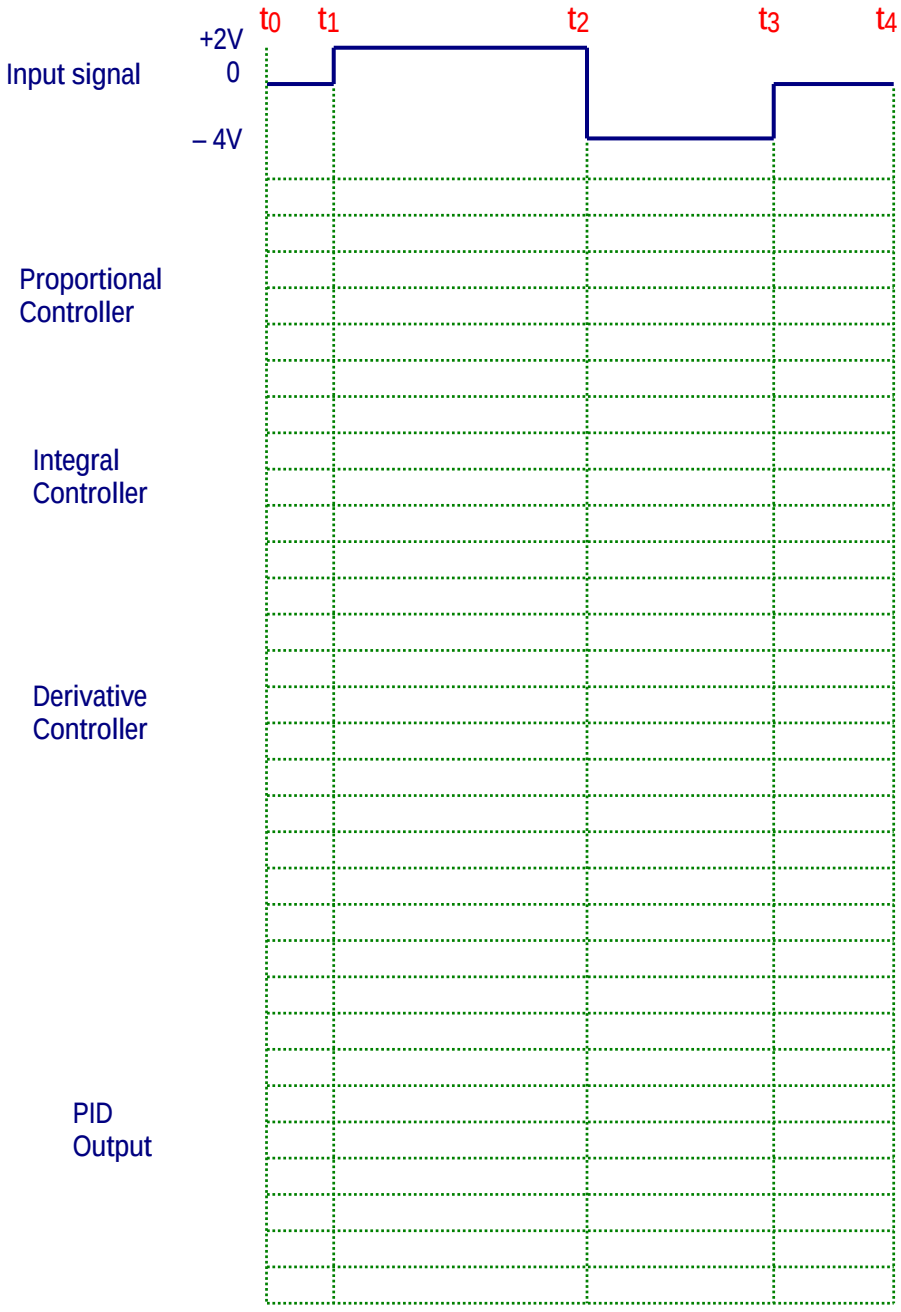



**Figure 9-17.**

# LAB 10 – CLOSED–LOOP SYSTEM

## Objectives

1. Build, test, and troubleshoot a closed-loop system using Multisim.
2. Design a closed-loop system to meet certain requirements.
3. Hardwire the closed-loop system of objective 1 and compare the behavior of the hardwired circuit with the one obtained using Multisim.

## Preliminary Information

Most industrial processes are analog, where we regulate one or more of the following variables: position, speed, pressure, force, flow, power, level, temperature, tension or intensity. To accomplish this, we use either individually or in a combination control technologies. For example, a valve can regulate flow with mechanical, pneumatic, and/or electronic technology.

The type of regulation depends on the process. We know what values we should adjust in certain circumstances. However, in some cases the process loses control because of process variations such as an increase in production, appearance of interferences, equipment aging, changes in temperature, and other factors.

The control of the shower head flow is a clear regulation example. To obtain warm water, we open the hot water and cold water valves alternately until achieving the desired temperature; in this case the temperature sensor is the user. If for some reason the water becomes very hot or very cold, we will have to manipulate the hot water and cold water valves again until the desired temperature is achieved.

This method of controlling the warm water through human intervention is referred to as an open loop control and its block diagram is shown in figure 10-1.


**Figure 10-1. Open-loop control system.**

**The open-loop control system** is a relatively quick and simple way of implementing a mean of the process control. However, the accuracy of the system **depends on the calibration carried out by a human operator**. This method, however, is not desirable when:

1. The regulation of the process is complex and it requires a high degree of involvement by a human operator. Human control of complex processes often leads to errors due to fatigue, distraction, and other human-based factors.

2. On occasion, a manual control might require a person with much expertise and/or training and that individual is difficult to replace.

3. In large-scale production, the manual control is expensive because any mistake generates loss on excessive waste of material or reduction in the quality of the product.

For the above considerations, among others, it is preferred to use the closed-loop system as depicted in Figure 10-2.

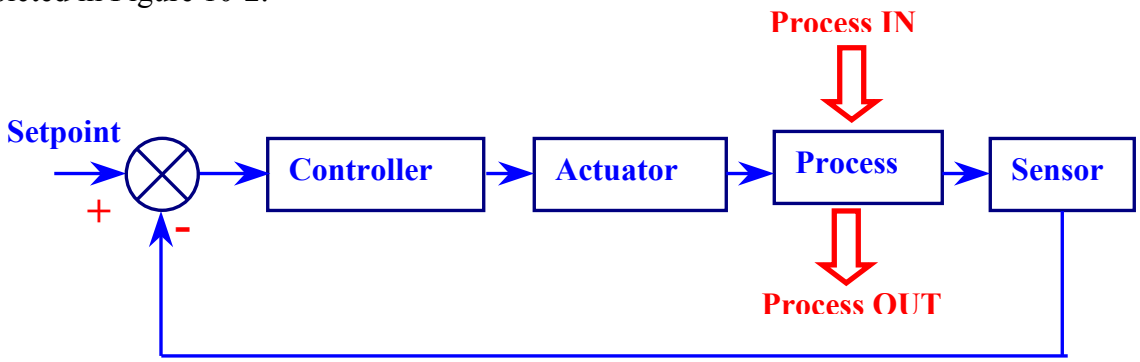



**Figure 10-2. Closed-loop control system.**

Figure 10-2 shows the generic block diagram of a closed-loop control system where the variable to be controlled or output signal is continually measured (sensed) and compared with a reference value or set point. The difference between the controlled signal and set point is referred to as error or system deviation.

The controller executes the necessary corrections on the actuator until the signal difference or error signal is eliminated and the controlled variable equals the reference variable.

**The closed-loop control system uses feedback to reduce the error of the system.** The use of feedback improves stability of the process even though external interferences and internal variations of the parameters of the system are present.

**In industry, two or more dependent or independent closed-loop control systems are usually used to fine-tune the control of a process.**

Figure 10-3 depicts a temperature control system. When the measured temperature deviates from the set point, the temperature controller varies the position of the valve or actuator that regulates the fuel input.

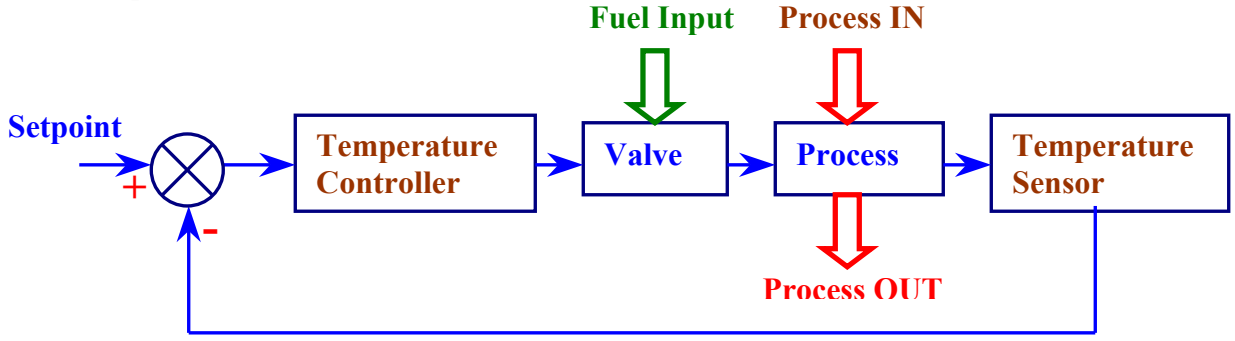



**Figure 10-3. Closed-loop temperature control.**

This closed-loop system adequately provides all the controlled characteristics of the fuel (pressure, viscosity and other characteristics) that keep fuel flow constant.

Those characteristics such as fuel pressure varies that changes the flow through the actuator. With variable fuel pressure, the temperature of the system will also vary requiring the controller to carry out the necessary changes to readjust the actuator.

The changes of fuel pressure are random and delays in the temperature sensing feedback loop slow the correction signals generated by the controller. The continuous changes of pressure will produce a continuous correction on the control valve that sometimes acts when it is no longer necessary because the flow has returned to its normal condition. This results in a system instability that must be accounted in the control system design.

**One likely way of addressing this system instability could be the following:**

We notice in this control system that temperature is regulated by the flow of fuel and the fluctuations of this flow affect the temperature of the process. As a means of possibly improving performance of the system one might consider sensing the flow of the fuel rather than the change in temperature. Due to thermodynamics concerns, temperature changes and the ability to sense such changes is slower than the ability to measure changes in pressure. However, because there is a correlation between fuel pressure variation and temperature changes we can anticipate what a temperature change will be based on fuel pressure variations. This will enable us to adjust the valve to compensate for the variations in fuel pressure before the fuel flow variations affect the controlled temperature.

In this case, we say that the temperature sensor is controlling the primary variable and the flow sensor is controlling the secondary variable. In this way, there are two feedback loops in the process control system as shown in figure 10-4.

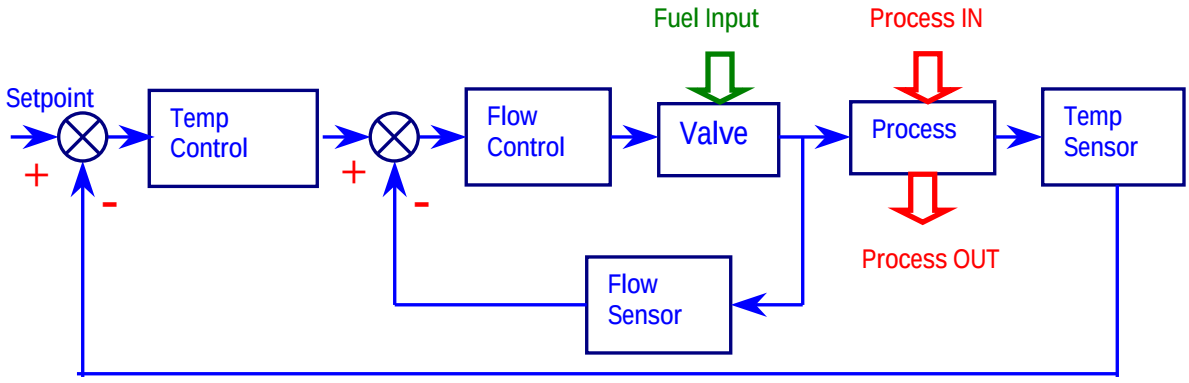



**Figure 10-4. Dual closed-loop control.**

The output signal of the temperature controller acts as reference point for the flow control. The second controller will quickly correct the flow variations due to the random interferences of the fuel and maintain the capacity of the system to control the temperature along with the primary controller at all times.

**In a process where two or more variables are controlled, one variable dependent on another, it is said that the control system is a cascade control.**

## Dead band

To clarify the dead band concept, we will refer to the ON/OFF controller that is used to control the temperature of such processes as ovens in the plastic industry and others.

The controller turns on or off the heating element so that the temperature increases or decreases respectively. This is often accomplished by means of an electromechanical relay or a thyristor, in the following way:

1. When the temperature measured by the sensor is below the Set point, the controller activates the relay or the thyristor and the heating element is on.

2. When the temperature measured by the sensor is above the Set point, the controller disables the relay or the thyristor and the heating element is off.

Figure 10-5 shows graphically the two described steps.

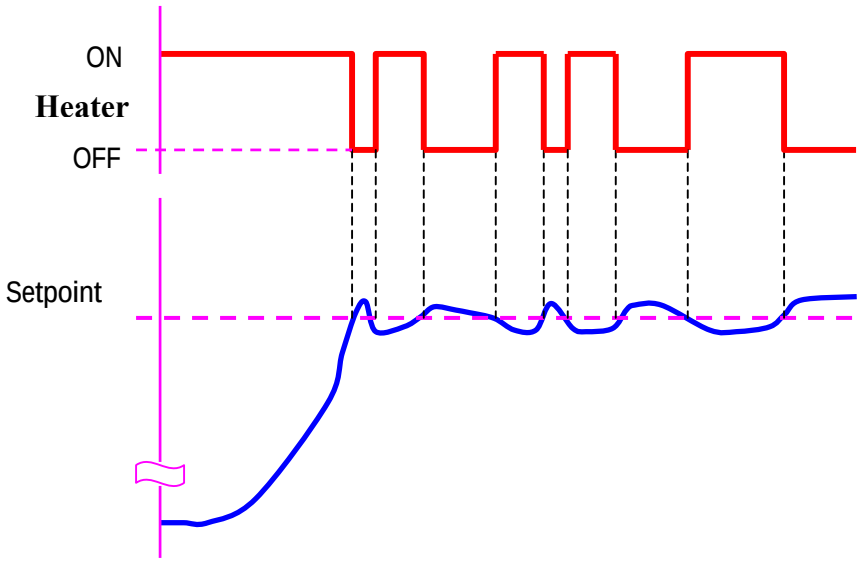



**Figure 10-5. The heating element is activated or disabled due to slight temperature variations.**

In figure 10-5, when the temperature is slightly higher than the set point (for example 0.1 ºC) the controller disables the heating element and when the temperature is slightly smaller than the set point (example 0.1 ºC) the controller will activate the heating element.

The result of this will be the controller continually turning on or off the heating element thereby causing the deterioration of the relay contacts. To avoid this problem, a “dead band” circuit is added to the ON/OFF Controller so that the heating element is activated a few degrees above the set point and is disabled some degrees below the set point. The hysteresis (difference between ON/OFF temperatures) introduced by this dead band not only extends the life of components in the system, but it also prevents system instability in the form of oscillations.

Figure 10-6 shows the output graph of an ON/OFF controller with a dead band of +/– 10 ºC around the set point of 150 ºC. Therefore, when the temperature reaches 160 ºC, the heating is turned off and conversely, at 140 ºC the heater is switched on..

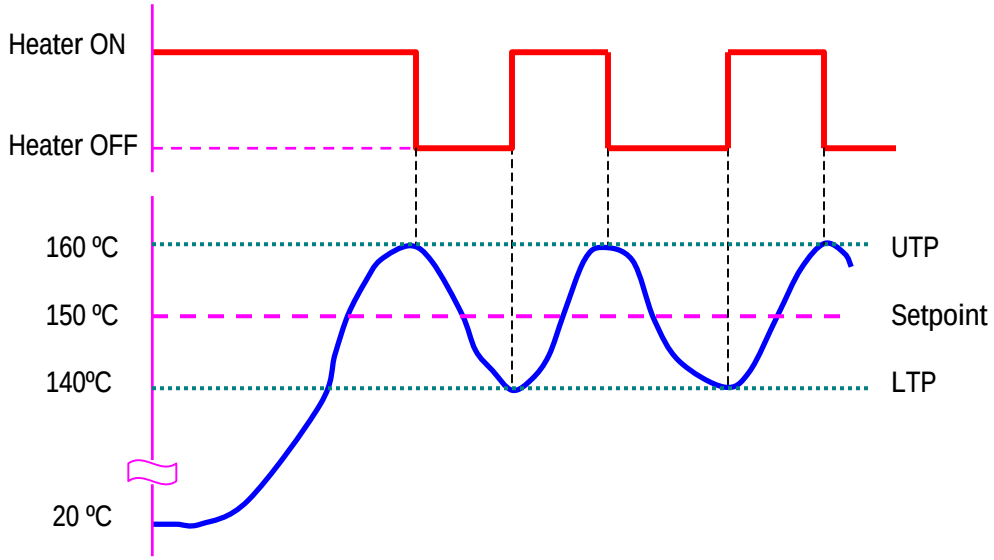



**Figure 10-6. Output graph of the ON/OFF Controller with dead band of 20 ºC.**

**We can say that the "dead band" is the space in which the error signal changes without producing any effect on the controller's output.**

In equation form, the dead band is:

**Dead band = UTP – LTP**

Where: **UTP: Upper Trigger Point.**

**LPT: Lower Trigger Point**.

Replacing values for our example, we have:

Dead band = 160 ºC – 140 ºC = 20 ºC

Figure 10-7 shows an ON/OFF Controller with an additional op-amp circuit that provides the dead band.

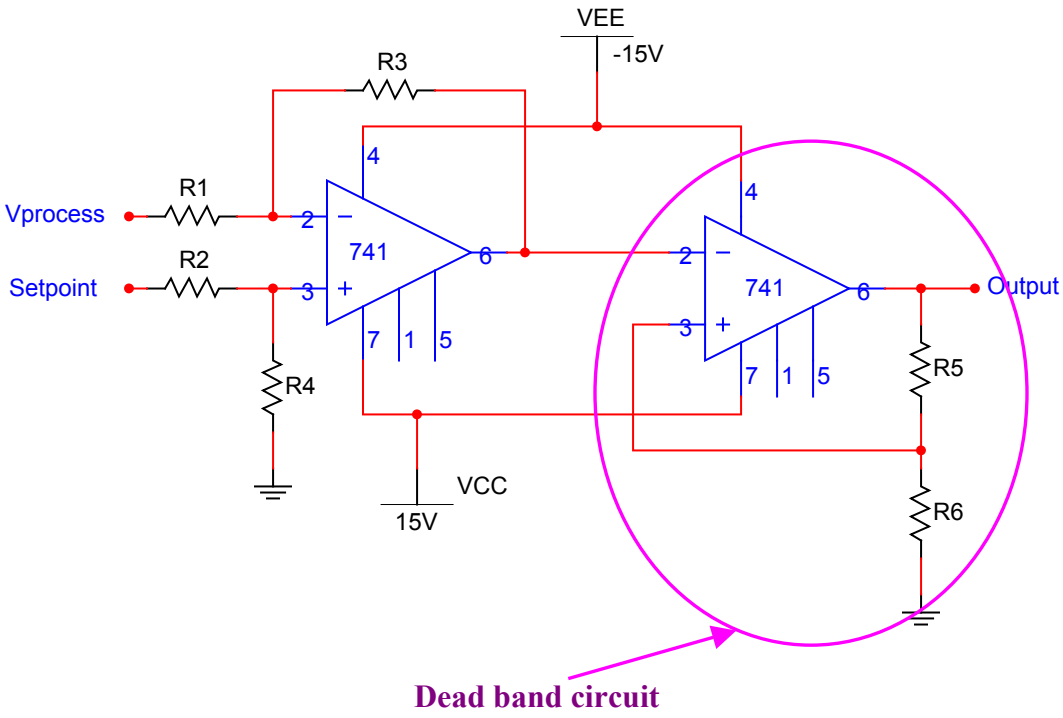



**Figure 10-7. Basic ON/OFF Controller with dead band.**

## Problem Statement

Build an ON/OFF temperature control. This system should control the temperature of a reaction chamber whose internal temperature must be between 250 °C and 450°C throughout the reaction. The RTD to be used has the output response shown in Figure 10-8 that responds to temperatures variation between – 200 ºC to 750 ºC.

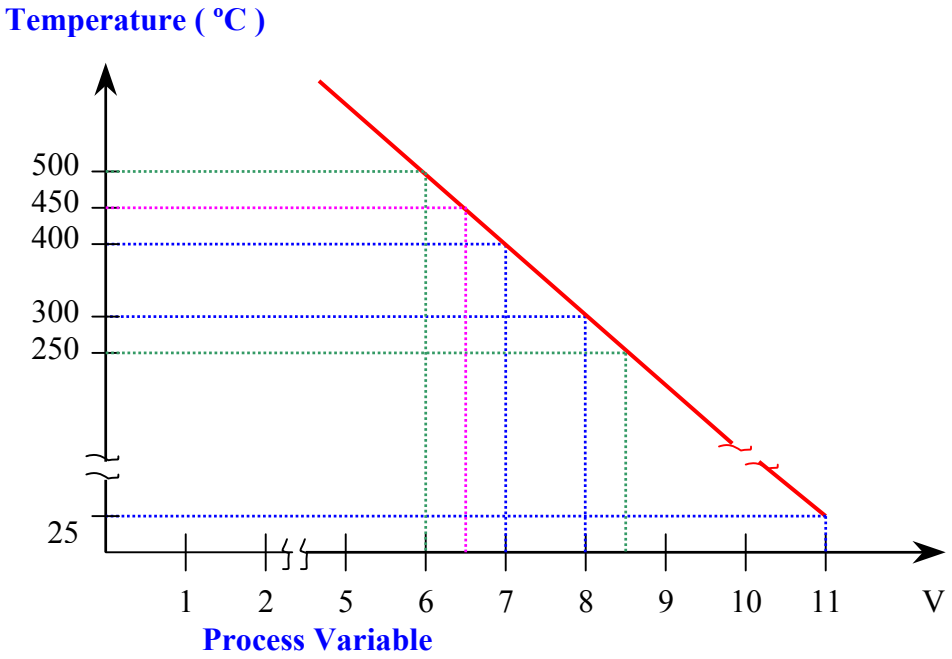



**Figure 10-8. RTD Sensor temperature versus output response.**

Note that the RTD's output response shown is limited to the range from 250 °C to 500 ºC.

## Solution

The circuit shown in Figure 10-9 is an example of implementing the problem statement requirements.

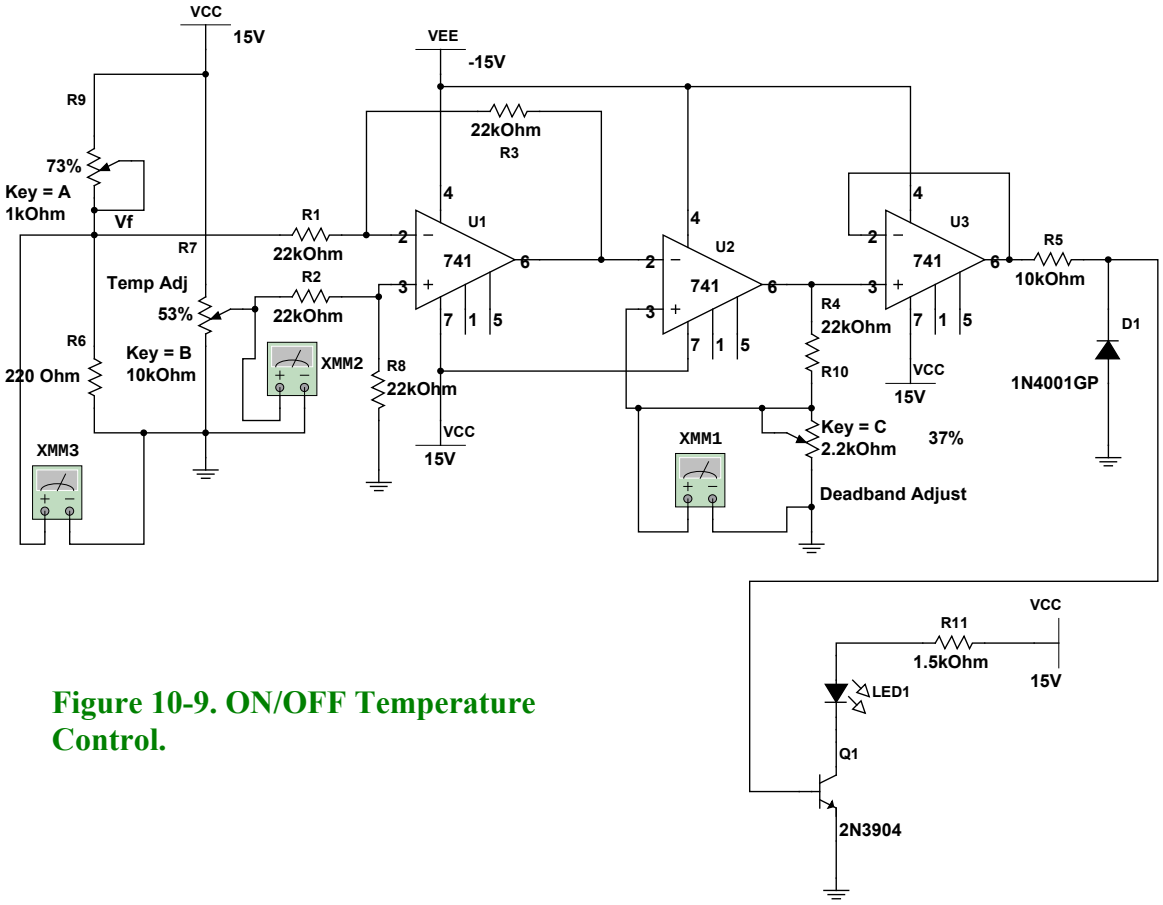



**Figure 10-9. ON/OFF Temperature Control.**

## Circuit Analysis

**$U_1$: Differential amplifier**. The gain is determined by R3/R1 or R8/R2; hence R3 = R8 and R1 = R2. Since resistors R1, R2, R3 and R8 are of the same value, the Gain is equal to 1. The output of the $U_1$ stage is given by the following equation:

$$Vo = VTemp\ Adj - V_f$$

**Resistor R7** controls the set point or desired temperature. $V_f$ is the Vprocess variable. Process-variable feedback is provided by R9 which simulates the RTD sensor assembly suspended inside the reaction chamber.

The **Vprocess variable** or **$V_f$** is determined by the equation: $V_f = [R6/(R6 + R9)]\ (V_{CC})$.

As the temperature increases, the resistance of R9 increases and therefore the Vprocess variable or $V_f$ decreases because the voltage drop across R6 is applied to the inverting input of U1.

Although the temperature control must be between 250 ºC to 450 ºC, the thresholds are established between 300°C and 400°C. This threshold setting accommodates the oven's thermal inertia. This thermal inertia is caused by a heat propagation delay between the heating element and temperature sensor response time.

Because the heater (simulated by an LED in this circuit) is off until the temperature decreases to 300 °C, the temperature will actually fall below this value to approximately 250 °C, due to thermal inertia. Likewise, when the heater is shut off at 400 °C, the temperature will continue to rise to approximately 450 °C. It is clear from this example that thermal inertia results in an on/off overshoot/undershoot which must be corrected by the dead band settings.

From Figure 10-8 the Vprocess variable or $V_f$ for 300 °C is 8 V and for 400 °C is 7 V.

### Vtemp adj

Knowing these values, Vtemp adj (R7) is adjusted to 7.5 V to allow the error voltage to vary – 0.5 V at 300 °C to + 0.5 V at 400 °C.

### Window Comparator ($U_2$)

To maintain the 300 °C to 400 °C temperature range, the comparator must switch when one of these two thresholds (UTP = + 0.5 V and LTP = – 0.5 V) is reached.

Potentiometer R10 (along with R4) controls the values of the Upper Trigger Point (UTP) and the Lower Trigger Point (LTP). The equations for calculating UTP and LTP are:

$$\text{UTP} = \frac{\text{R10}}{\text{R4} + \text{R10}}(+\text{Vsat})$$

$$\text{LTP} = \frac{\text{R10}}{\text{R4} + \text{R10}}(-\text{Vsat})$$

### Buffer ($U_3$)

$U_3$ is a voltage follower that isolates the load from $U_2$ to prevent asymmetrical saturation voltages.

**Diode D1 prevents –Vsat from reaching the base of transistor Q1.**

### System Operation

The circuit of Figure 10-9 is powered with +Vcc = +15V and –Vcc = –15 V. By considering the 2 V internal voltage drop for a 741 op-amp, we see the maximum available output voltages or saturation voltages (Vsat) are:

+Vsat = +13V and –Vsat = –13 V.

The circuit function will be discussed in the beginning with the output and ending with the input, in other words, from right to left. We will assume that the heating element, represented by the LED, is initially off.

Therefore, initially, U3 Vo = –Vsat = –13V → Vout, comparator (U2) Vo = –13 V

–13 V at U2's pin 6, means the voltage drop across R10 will be negative and equal to – 0.5 V. This voltage is applied to U2's pin 3; therefore: voltage on U2 pin 3 = LTP = – 0.5 V.

Process Variable = Vprocess var = + 11 V (See Figure 10-8, Temperature = 25°C)

Verror = $Vo_{,U1}$ = Vtemp adj – Vprocess var = 7.5 – 11 = – 3.5 V

Since the voltage at pin 2 of comparator (U2) = – 3.5 V (Verror) and the voltage at pin 3 of comparator (U2) = – 0.5 V, → Vout, comparator (pin 6) = + Vsat = + 13 V. **Voltage at the non-inverting input is more positive than at the inverting input.**

Thus, the heater is turned ON and the voltage at pin 3 of comparator becomes UTP = + 0.5 V.

From Figure 10-8, as temperature increases the Vprocess var (or $V_f$) decreases and the Verror increases. (Verror = 7.5 V – Vprocess var).

When the temperature is greater than 400 °C, the Vprocess variable becomes less than 7 V. The Verror ($Vo_{,U1}$) becomes greater than + 0.5 V and the Vout, comparator ($Vo_{,U2}$) = –Vsat = –13V. Now, the voltage at the inverting input is greater than the voltage at the non-inverting input. Under these conditions, the heater turns OFF. However, the chamber temperature may continue to rise slightly. At this point the voltage at pin 3 of comparator becomes LTP = – 0.5 V.

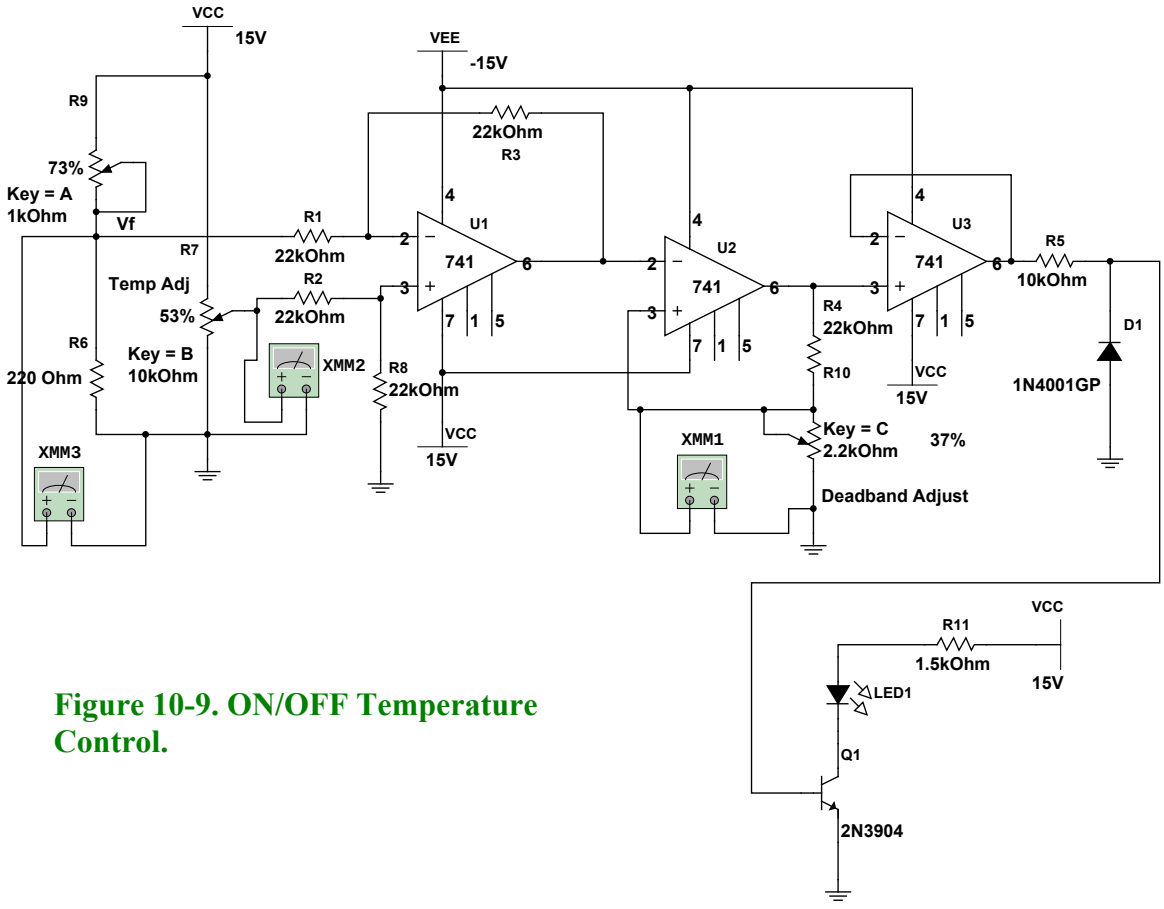



**Figure 10-9. ON/OFF Temperature Control.**

## Procedure/Tasks Using Multisim

Build the circuit shown in Figure 10-9 using Multisim. Use virtual resistors and virtual potentiometers.

1. Turn on the power switch.
2. Press the C or SHIFT-C keys to adjust the voltage at pin 3 of $U_2$ to approximately – 500 mV or + 500 mV.
3. Press the B or SHIFT-B keys to adjust the Vtemp adj (R7 wiper) voltage to approximately 7.5 V
4. Press the A or SHIFT-A keys to adjust the voltage $V_f$ until it is higher than 8 V. When this happens, the heater turns ON (in this case the LED is ON).
5. Now, press the A or SHIFT-A keys to adjust the voltage $V_f$ until it is lower than 7 V. When this happens, the heater turns OFF (in this case the LED is OFF). Notice that the LED remains ON while $V_f$ is adjusted from approximately 8 V to 7 V.
6. Finally, press the A or SHIFT-A keys to adjust the voltage $V_f$ until it is higher than 8 V. When this happens, the heater turns ON (in this case the LED is ON). Notice that the LED remains OFF while $V_f$ is adjusted from approximately 7 V to 8 V.

Steps 5 and 6 demonstrate that the output will not change within the 7 to 8 V window because it is typical of ON/OFF controllers with dead band. See Figure 10-10.

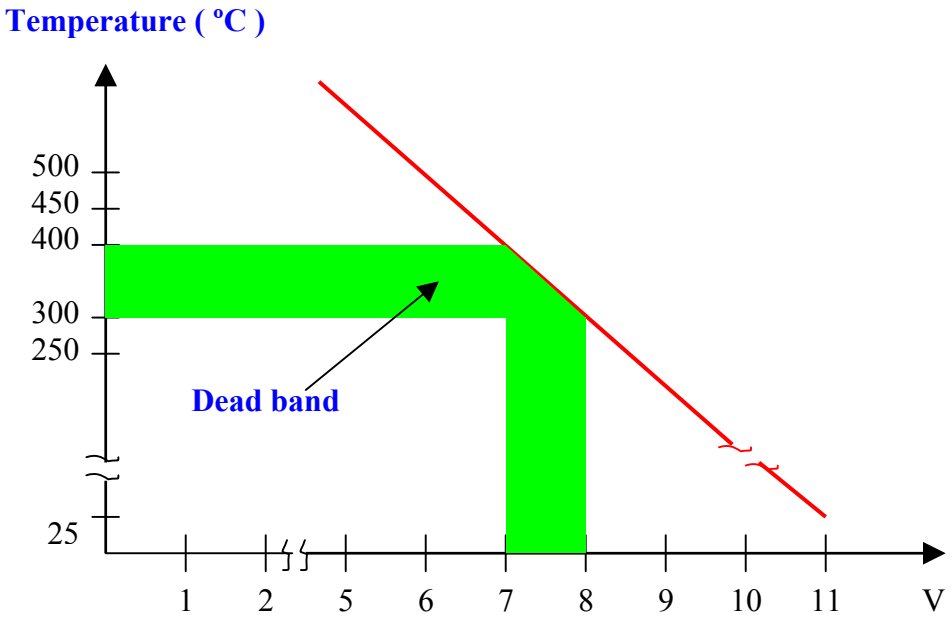



**Figure 10-10. Dead band graph.**

## Questions

1. Replace the three 741 ICs by a single quad op-amp such as the LM324AJ. Note that the LM324AJ has four sections A, B, C and D. Each section will have its own power supply terminals pin 4 (+V) and pin 11 (– V). You only need to connect the respective power supplies to any section. Pin 4 should be connected to VCC and pin 11 should be connected to ground. Does the circuit behave the same? Explain any discrepancies.

2. Perform the necessary adjustments if we need to control the temperature from 250 °C to 500 °C. Test your design. How can you verify that your design is correct?

3. If you want to replace the LED by a 500-W heater what modifications will you do? Test your design using Multisim.

## Procedure/Tasks – Hardwired Circuit

Now that you have tested the circuit to satisfy the problem statement using Multisim, it is time to hardwire the circuit depicted in Figure 10-9 using real components. Repeat what you did in Lab 1, that is, going through a checklist of components and the equipment that you need to complete this task. This will reduce the careless errors and damaging to components.

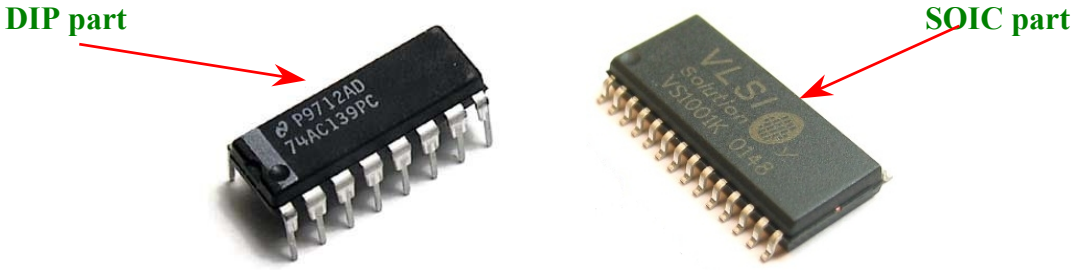
### List of components

| Item # | Component Description | Reference Designators | Quantity |
|---|---|---|---|
| 1 | National ADC0804 (or equivalent) | A1 | 1 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

### Equipment List

It is good practice to have a standard procedure checklist to use in hardwiring circuits. Such a checklist will enable you to minimize errors in the circuit assembly, prevent malfunction, and consequently reduce damage to components. This checklist should always be used, but it is not limited to the following:

- Ensure all components are checked before powering up the circuit for the first time.
- Check your circuit schematic against the data sheet for each component used in the circuit design. Most common problems include the following:
    - Wrong component pinouts, especially between DIP & SOIC parts. For example, it is not uncommon for IC manufacturers to use different pinouts between DIP8 and SOIC8 packages for the same part. Sometimes a circuit designer will layout a schematic using the pinout of a SOIC part and the person doing the wiring of the circuit might use a DIP part. The consequence of this is an incorrectly wired breadboard resulting in faulty circuit operation and possible IC damage.
    - $V_{CC}$, $V_{EE}$ or ground being connected to the incorrect pin on the IC
    - Pins left floating that should be tied to high or low, depending on the part requirements in the data sheet or application notes.




- Ensure that every IC has its own noise decoupling or bypass capacitor (e.g., 0.1 uF) connected directly between the IC's $V_{CC}$ and ground pin and $V_{EE}$ and ground pin. This is especially important when there is high speed digital or high load current circuit combined with sensitive analog circuit (such as a Wheatstone Bridge).
- If the power supply leads to the breadboard are long (i.e., > 6 inches), then a tantalum bypass capacitor of 10 - 33µF should be used between power and ground points where the power supply leads connected to the breadboard. Make sure the polarity of the capacitor is correct and that its working voltage (WV) is at least 1.5 times the maximum voltage used in the circuit. e.g., the working voltage rating of a capacitor is 10V; the supply voltage should be at least 15V.
- Ensure all amplifier feedback loops are as short as possible. Ideally, feedback resistors should be routed from the amplifier's output pin directly to the input pin. This helps to reduce the possible amplifier's feedback loop pick up and amplifies the unwanted noise.
- All component leads should be kept as short as possible. Additionally, signal paths from one IC to another should be kept as short as possible to minimize noise pickup. If a signal

path must be long for whatever the reason, ensure it is routed away from other signal paths to reduce the possible crosstalk.

- Prevent ground loops and loops in power supply connections. On the breadboard, all ground buses and power buses should be terminated at one point. That is, you should not be able to trace a loop from the ground or power supply connection to the breadboard to all component connections and then back to the power or ground source.
- If it is possible, separate analog and digital power supplies and grounds from one another. These should be routed back and terminated at the single power and ground termination point as previously discussed.
- Whenever possible, choose a CMOS part over a high current TTL part (e.g., LMC555 over a TTL 555), unless the design calls for the high current capacities of TTL.

## Procedure/Tasks

1. Apply power to the circuit.
2. Adjust R10 potentiometer until the voltage at pin 3 of $U_2$ to approximately – 500 mV or + 500 mV.
3. Adjust R7 potentiometer until the $V_{temp\ adj}$ (R7 wiper) voltage is approximately 7.5 V.
4. Adjust R9 potentiometer until the voltage Vf is higher than 8 V. When this happens, the heater turns ON (in this case the LED is ON).
5. Now, adjust R9 potentiometer until the voltage Vf is lower than 7 V. When this happens, the heater turns OFF (in this case the LED is OFF). Notice that the LED remains ON while Vf is adjusted from approximately 8 V to 7 V.
6. Finally, adjust R9 potentiometer until the voltage Vf is higher than 8 V. When this happens, the heater turns ON (in this case the LED is ON). Notice that the LED remains OFF while Vf is adjusted from approximately 7 V to 8 V.
7. Replace the three 741 ICs by the LM324.
8. Repeat steps 1 to 6 and compare the results using a single LM324 instead of three 741 ICs.

## Questions

1. Does the hardwired circuit respond the same as the one you tested using Multisim? Explain any discrepancies.

2. Do you get the same results when replacing three 741 ICs by a single LM324 IC? Explain any discrepancies.

## Conclusions.

After completing the closed-loop system both in Multisim and hardwire, what are your conclusions?

# MATERIALS FOR TECH 167 KIT - LAB EXPERIMENTS

| Quantity | Description |
|---|---|
| 1 | Breadboard (Jameco: Part No.: 20722CJ) |
| 1 | DAC 0800 IC LCN – DIP (14904) |
| 1 | ADC 0804 IC LCN – DIP (10153) |
| 5 | 741 IC OPAMP (24539) |
| 1 | LM324AN IC OPAMP (212169) |
| 2 | 2N3904 BJT NPN transistor – In stock |
| 2 | 2N3906BJT PNP transistor – In stock |
| 10 | LEDs, red (Jameco: Part No.: 333171CJ) |
| 1 | 1N4001GP DIODE (35975PS) |
| 1 | 1N4002GP DIODE (76961PS) |
| 1 | 2N5064 SCR (211431PS) |
| 1 | HT-32 DIAC (160098) |
| 1 | SC146B TRIAC (31819) |
| | |
| 1 | 12V Incandescent Lamp (Jameco: Part No.: 209998CJ – T1 wire terminal) |
| 1 | 120V – 15V AC TRANSFORMER (All Electronics, page 55) |
| 1 | 250 V - 2 AMP FUSE (10381) |
| 1 | SPDT SWITCH (Jameco: Part No.: 21910CJ) |
| 1 | DIP SWITCH (Jameco: Part No.: 38842CJ) |
| | |
| 8 | 220 Ω - ¼ W resistor |
| 8 | 330 Ω - ¼ W resistor |
| 7 | 1 kΩ - ¼ W resistor |
| 1 | 1.5 kΩ - ¼ W resistor |
| 4 | 2.2 kΩ - ¼ W resistor |
| 1 | 3.3 kΩ - ¼ W resistor |
| 1 | 3.6 kΩ - ¼ W resistor |
| 1 | 4.7 kΩ - ¼ W resistor |
| 1 | 5.6 kΩ - ¼ W resistor |
| 1 | 10 kΩ - ¼ W resistor |
| 5 | 22 kΩ - ¼ W resistor |
| 2 | 100 kΩ - ¼ W resistor |

**½ W, 3/8" Square, Single Turn Trimming Potentiometers Top Adjustment**

| | |
|---|---|
| 1 | 500 Ω potentiometer (Jameco: Part No.: 254318CJ) |
| 1 | 1 kΩ potentiometer (Jameco: Part No.: 254326CJ) |
| 1 | 2 kΩ potentiometer (Jameco: Part No.: 254334CJ) |
| 2 | 5 kΩ potentiometer (Jameco: Part No.: 254342CJ) |
| 2 | 10 kΩ potentiometer (Jameco: Part No.: 254351CJ) |
| 1 | 100 kΩ potentiometer (Jameco: Part No.: 254393CJ) |
| 1 | 1 MΩ potentiometer (Jameco: Part No.: 254431CJ) |

| | |
|---|---|
| 1 | 0.01 uF capacitor – 25 V |
| 2 | 0.1 uF capacitor – 25 V |
| 1 | 10 uF capacitor – 25 V |
| 1 | 100 pF capacitor – 25 V (15341CC) |
| 1 | 47 pF capacitor – 25 V (15510CC) |