

## Exercise E – LTspice, Diodes and MOSFETs

**Aim** — To learn how to use the circuit simulation software LTspice for the design and analysis of electrical circuits.

### Objectives

1. To learn how to use LTspice Schematics to set up circuit models.
2. To gain experience with using the component library, and in particular to be able to specify the following components: resistors, capacitors, inductors, DC and AC voltage sources, diodes, metal oxide semiconductor field effect transistors (MOSFETs).
3. To learn how to use a variety of simulation modes by carrying out a series of exercises: steady state DC and AC; transient DC and AC; small-signal analysis; frequency response.
4. To gain experience with the post-processing facilities of LTspice in order to view and analyse the results of your simulations.
5. To reinforce the IA Analysis of Circuits lecture course by comparing LTspice simulation results with analytical and experimental results.

### Introduction

In the IA Analysis of Circuits and Digital Circuits courses you will learn about the properties of circuit components, how they are connected together to form circuits which perform a variety of useful functions, and how such circuits are analysed. Whilst it is important to have a good understanding of the principles of electrical engineering, in practice the methods of circuit analysis you will be taught have limited applicability. For example: the analysis of circuits with many circuit components results in large systems of equations to solve; for transient studies these equations can turn out to be systems of coupled differential equations; non-linear circuit elements such as diodes and metal oxide semiconductor field effect transistors (MOSFETs) can result in the equations becoming analytically intractable. For these reasons it is common to find that electrical engineers use circuit simulation packages, of which LTspice is an example.

The aim of this exercise is for you to gain experience using LTspice, so that you can apply what you learn here to the design and analysis of the final design project you will be building in the Integrated Electrical Project (IEP). Examples will be used to analyse diodes and DC MOSFETs with experimental analysis to compare with modelling results.

The handout starts by explaining how to use the LTspice Schematic editor in order to define circuits, and how to define component values within the editor (Model definition). It then continues to show how LTspice can be used to perform a variety of types of analysis: DC, AC,

frequency response, transient, bias point and small-signal analysis (Model simulation). Some of the facilities for obtaining useful output from the simulations are explored (Post-processing). All of this is achieved via a number of exercises, which have been synchronised with the IA Analysis of Circuits lectures so that you can work through them over the Xmas break.

You can quickly generate lots of files whilst running LTspice, and so you are advised to create an LTspice folder, perhaps with sub-folders for the various exercises. It is also a good idea to save your work to a memory stick.

Completing all 13 parts of this exercise in the handout will ensure that you gain experience with simulating all the circuit components, and all the modes of circuit simulation that you will meet in the IA Analysis of Circuits course.

Six marks of standard credit are available on successful completion of Exercise E, three for the LTspice work and three for the kitset experiments. A mark-up will be conducted in the Lent term. You should upload your completed IEP workbook for both parts of the exercise onto Moodle and be prepared to demonstrate your understanding of these exercises.

### Kitset experimental components

After several of the LTspice simulations is a corresponding experiment using the ADLP2000 kitset components. There will focus on the diodes and MOSFETs in the kit shown in the Figure 0 below. Please note these components are fairly robust, but care should be taken handling the MOSFET as they are prone to damage from static electric charge.

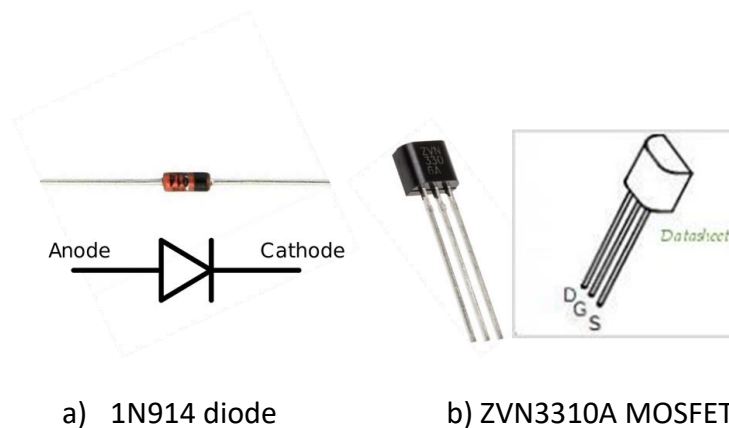


Figure 0. New components from the kitset

### Exercise 0 LTspice Tutorial – The potential divider (redux)

**Theory** — Consider the circuit shown in Fig. 1, consisting of two resistors in series connected to a 10 V DC power supply. Determine the current flowing in the circuit and the voltage across the load resistor,  $V_L$ .

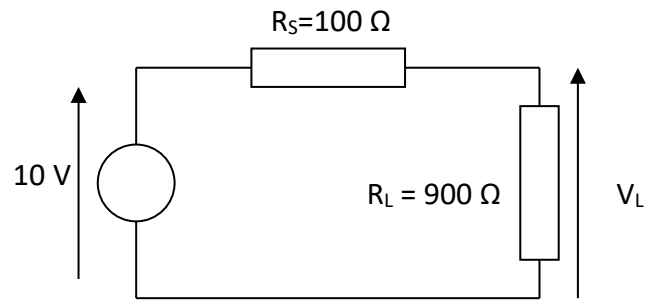


Fig. 1 Potential divider circuit







Current = ..... mA

Voltage across  $R_L$  = ..... V

Any LTspice simulation may be regarded as three operations: circuit definition; circuit simulation; results processing (this may be trivial/non-existent in the case of very simple circuits such as this one!)

Circuit definition is achieved via the LTspice Schematic editor. This provides a convenient graphical user interface (GUI) for this purpose, and also provides the interface for specifying the circuit simulation to be carried out.

### LTspice simulation

1. Open LTspice.
2. Open a new file – **File, New Schematic.**
3. Get the components you will need. For resistors, capacitors, inductors and diodes you will see shortcuts on the top toolbar ( , ,  and  respectively). For other components (as well as the aforementioned ones) click . A new window will appear. This gives access to a huge library of pre-defined components. For example, if you use the scrollbar to scroll across you will come to **res**. Click on this and a box will appear with the circuit symbol for a resistor (old-fashioned one!) and some text to confirm that it is indeed a resistor. If you now click on OK the box disappears and you will be able to place resistors on the main window by clicking the left-hand mouse button. When you have enough resistors click the right-hand mouse button to halt the procedure. You will also need a **voltage** source — follow the same procedure as you did for resistors except click on **voltage** in the component list. Finally you will need a ground for your circuit. There is a shortcut for this on the top toolbar, click on  and then place your ground on the main window. At this stage don't worry about where exactly you have put all your components. If everything has gone to plan then your main window should look something like Fig. 2 below.

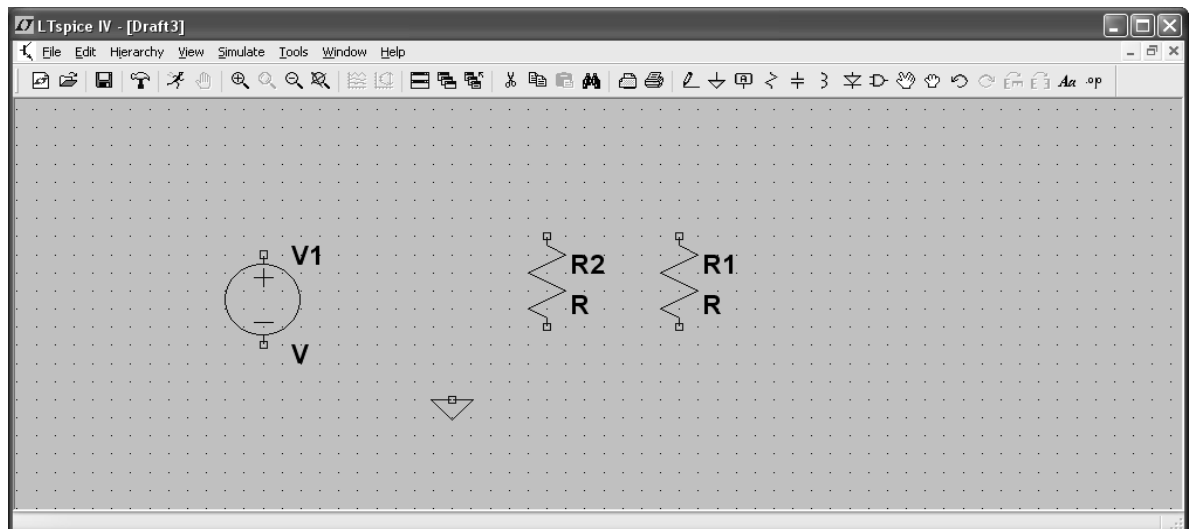






Fig. 2 LTspice screen after step 3

4. Now you need to orientate your components and wire them up into a circuit. Usually some sort of square formation works well. Accordingly, select one of your resistors.

To do this, click on  from the top toolbar, and then click on the resistor. Now enter Ctrl+r – this rotates the resistor so that it is horizontal. Alternatively, select the component you want to rotate and click on the  symbol from the top toolbar. Click

on the left mouse button. You can click on all of your components whilst  is highlighted and drag them to where you want them. Now wire up your circuit. To do

this click on  from the top toolbar and align the cross-hairs that appear with one of the component terminals. Trace a path with the mouse in order to link the component with the next component in the circuit. Click the left mouse button if you need to put a right-angled bend into the wire. Also click the left mouse button when you reach the next component. You can repeat this process until all components form a circuit. Finally attach your ground point using the wiring tool. Notice that this creates a node, which appears as a square dot on your circuit. This square dot indicates that there is an electrical connection. In LTspice wires can cross each other without an electrical connection existing — in that case there is no square dot. You should now have something like Fig. 3 below.

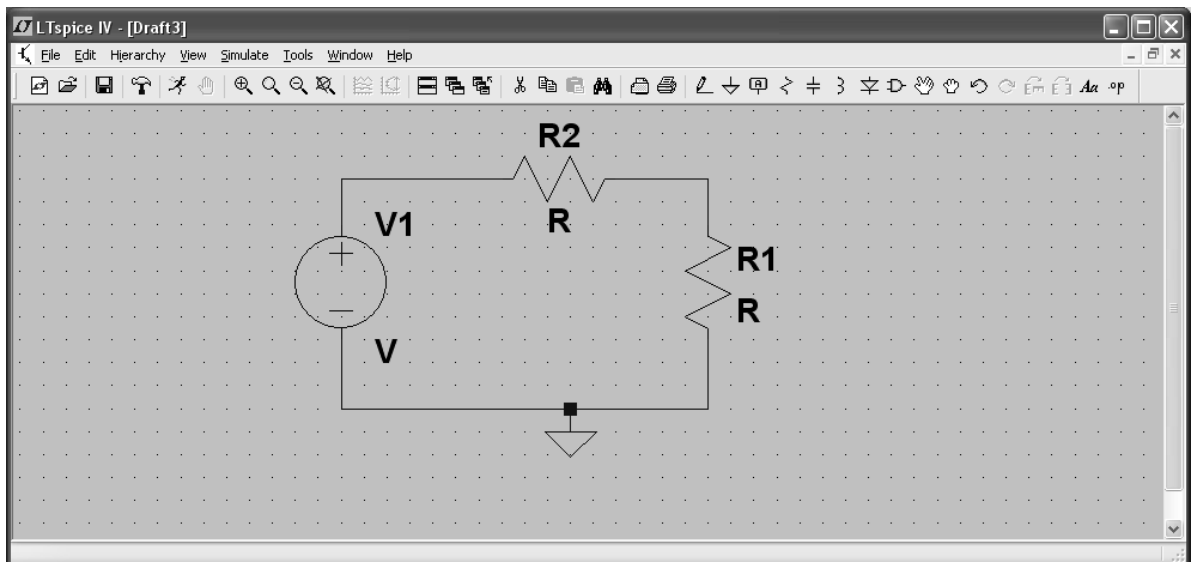


Fig. 3 LTspice screen after step 4


5. Now you need to give all your components values. For a resistor, point at the resistor and right click. A dialogue box will appear and you simply type the value of the resistor in the **Resistor[Ω]**: box. There are boxes for the resistor tolerance and power rating but these can be left blank. LTspice allows some useful abbreviations as follows:

p – pico =  $10^{-12}$       n – nano =  $10^{-9}$       u – micro =  $10^{-6}$       m – milli =  $10^{-3}$   
 k – kilo =  $10^3$       meg – mega =  $10^6$       g – giga =  $10^9$

Thus, to enter a 1 MΩ resistor for example, you could either enter 1000000 or 1meg.

For the voltage source, right click on it and a dialogue box will appear. You type the value of the DC voltage into the **DC value[V]**: box, and optionally a series resistance too. Notice that LTspice displays only the main window voltage source and series resistance values.

6. At this point your circuit is fully defined and so the next step is to simulate it. The only relevant simulation here is a DC analysis, known in LTspice as DC operating point. Click **Simulate, Edit Simulation Cmd** from the top toolbar and a dialogue box will appear. Note the various analysis modes available. Select **DC op pnt** and click **OK**. At this point the more basic side of LTspice is revealed. You will see that a box has appeared that you can drag around your schematic. Left click and in that box you will see the text **.op**. This is called an LTspice directive – originally this program was a purely command-line driven circuit simulator with no GUI, and as you have just seen, relics of this history still remain!

7. Now click **Simulate, Run**, or use the top toolbar shortcut . Following this you will see a window appear containing the results of the simulation. If all is well, your screen should now look something like Fig. 4 below.

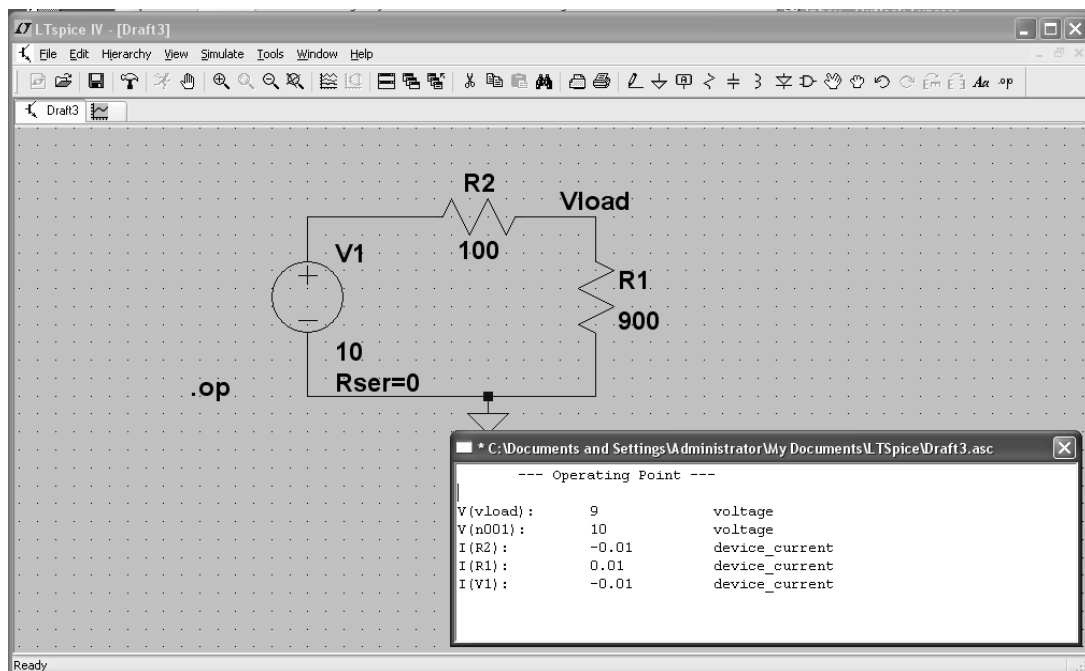



Fig. 4 LTspice window after step 7

Again, a legacy of the history of LTspice is revealed, namely a rather confusing text-based set of results, saying things like:

```
V(n002): 9          voltage
```

This means that the voltage at node 2 of the circuit is 9 V. But what is meant by node 2? LTspice internally allocates node numbers to all nodes in the circuit (ground excluded) but doesn't reveal to the user what this allocation is. Fortunately you can label nodes in the circuit to make interpreting these results more straightforward.

8. Close the results window and now select . A dialogue box appears and you can enter a name for a node. So, since you want to know what the load voltage is type **Vload** into the box labeled 'ABC'. Click **OK** and then attach the resulting box to the top of the load resistor in your circuit. Simulate the circuit again – this time the line that said:

```
V(n002): 9 voltage
```

has been replaced by:

```
V(vload):9 voltage
```

Much clearer! Notice that currents are described in a much more straightforward fashion — since currents flow through components they are referenced by the component names that appear on your circuit e.g.

```
I(R1): 0.01 device_current
```

means the current flowing through resistor R1.

9. Check your results — do the values printed on the screen for the circuit current and load voltage agree with your calculations? .....
10. Finally save your circuit — **File, Save As.**

## Exercise 1 Diode characterisation

The diode has the circuit symbol shown in Fig. 5(a). It is a semiconductor device which conducts current when the voltage of the anode is positive with respect to that of the cathode, but not when the cathode is positive with respect to the anode. These situations are termed **forward biased** and **reverse biased** respectively. Furthermore, in the forward biased case the voltage across the diode needs to reach a certain value known as the **forward voltage drop** before any substantial current flows. But after that the diode voltage increases very little with further increased current through it. There is a finite slope,  $dV_D/dI_D$  in this region and this gives rise to what is known as the **on-state resistance** of the diode. An idealised current vs voltage plot for a silicon diode is shown in Fig. 5(b), and from this it is seen that a diode can be most simply characterised by its forward voltage drop and its on-state resistance.

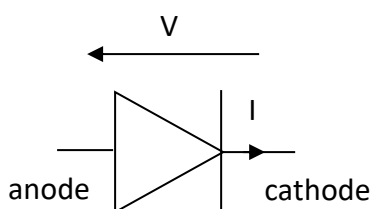


Fig. 5(a) Diode symbol and terminal names

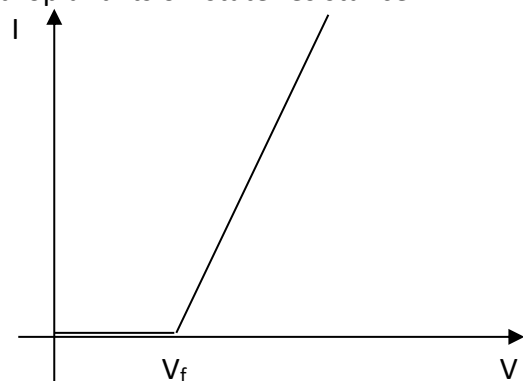


Fig. 5(b) Idealised I vs V characteristic

### LTspice simulation

LTspice can be used to plot these characteristics using the idea of the DC Sweep. The DC Sweep allows you to set up a DC voltage source (or current source) so that it varies between a lower and upper limit and with a certain increment. It will then plot the variable of your choice against the swept variable.

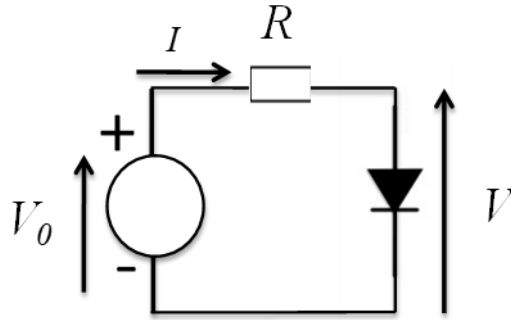


Fig. 6 Diode characterisation circuit

E1. Set up the circuit shown in Fig. 6. Select the **1N914** silicon diode by right-clicking on the diode in your schematic, **Pick New Diode** and then click on **1N914** from the list of diodes, then click **OK**. Set  $R$  to  $470\ \Omega$  and the DC voltage source ( $V_0$ ) to  $0\text{V}$ .

Set up the simulation; go to **Edit Simulation Command** and select **DC Sweep** and enter **V1** as the **Name of 1<sup>st</sup> source to Sweep** assuming this is the default name for the voltage source in your LTspice schematic. Set **Type of Sweep** to **Linear**, **Start Value** to **0**, **Stop Value** to **5** and **Increment** to **100 mV**. Click on **OK** and place the resulting LTspice directive somewhere on your schematic. **Run** the simulation.

Select the voltage across the diode ( $V$ ) and the current through the diode  $I(D1)$  to view, and observe that the diode current only starts to become significant when the diode voltage exceeds a certain value, around  $0.7\text{ V}$ . You may find it helpful to zoom in around this value on your plot. To find the diode resistance you could measure the slope of the characteristic and take its reciprocal. But you can make LTspice do the hard work for you. If you right click on  **$I(D1)$**  in the results window a dialogue box will appear into which you can enter expressions which are functions of  $I(D1)$  to plot. LTspice has a mathematical library of functions which is reasonably extensive. Type  **$d(I(D1))$**  into the **Expression Editor** – this is LTspice syntax for finding and plotting the derivative of  $I(D1)$ . This will give you a graph showing the conductance of the diode vs voltage (conductance is the reciprocal of resistance). Notice that the plot has zero slope for small voltages (ie infinite resistance) but a finite and approaching constant slope at  $V = 3\text{ V}$ .

- Capture a plot showing the voltage across the diode and the current through diode as a function of the swept voltage.
- From the plot estimate the diode resistance the diode forward voltage drop ( $V_f$ ) from where the diode starts to conduct and current flows.

E2. Now build the circuit in Figure 6 with  $V_0 = V_{\text{USB}}$  from your USB power pack,  $R = 470\ \Omega$  and the 1N914 diode which can be identified via the tiny writing on its body as in Figure 0. It should be easy to identify as there are 4 of them in the kitset. Note the orientation of the diode so



that it is forward biased. Set the Picoscope to measure DC on probe A at x 10 and set the amplitude to 10 V and set a Measurement for the Max value of Probe A.

- Measure  $V_{USB}$ , then reduce the Amplitude to 2 V and measure the forward voltage drop across the diode  $V_f$  and calculate the current passing through it. Does it match your simulation from E1?
- Repeat the experiment above with the 1N4001 diode and the Red, Green and Yellow LEDs. Note that on all the LEDs the Cathode is the wire closest to the flat spot on the side of the LED (also the shorter of the two legs on the LED). Can you explain why the forward bias voltages are different?
- Repeat the experiment using the Infrared Red LED (the slightly pink tinted one, not the black ones). How can you tell that it is emitting light?
- Replace the IR LED with the Red LED and add a second LED in parallel with it. What happens to the forward voltage and current? Does anything else change?

E3. Now reverse the polarity of the diode D1 using the move cursor and then rotating the symbol by 180 degrees so the diode is now pointing upwards.

- Repeat the simulation in E1. To estimate the leakage current and the reverse bias diode resistance.

E4. There are other types of diodes which can be simulated in LTspice. One of them is the 1N5817 Schottky barrier diode. A Schottky diode is made from a junction between a semiconductor such as silicon and a metal such as molybdenum, platinum, chromium or tungsten. This changes the characteristics of the diode, reducing its capacitance which means it can operate at higher frequencies. It also has an effect on the forward biased voltage drop across the diode.

- In the circuit of Figure 6, change the diode in LTSpice to the 1N5817 (note the snazzy symbol) and repeat the measurements in step 2.
- How does your measurement compare with the specifications of the diode that can be found online? Can you adjust your model to confirm this figure?

E5. Another common form of diode is the Zener diode. In this diode the p-n junction is engineered so that the breakdown voltage, when reverse biased, is reduced from 100's of volts down to a specific value defined as its Zener breakdown voltage. This will typically be in the range of 3 to 30 V. To test a Zener diode, set up the circuit in Figure 6, but with the diode reverse biased (pointing upwards).

- Swap the diode to the 1N750 Zener diode (even snazzier symbol) and increase the maximum sweep voltage to 8 V.

- Capture a plot showing the voltage across the diode and the current through diode as a function of the swept voltage and measure the Zener voltage of the diode. Why is the current through the diode negative?
- Can you give an example where this sort of diode might be useful?

E6. In the kit there is a 1N4729 Zener diode (labelled C3V6). Set up the circuit in Figure 6 and use Probe A to measure the forward bias voltage across the diode (Amplitude 5 V, DC mode, x10) and also set up Probe B to measure the supply voltage  $V_{USB}$  (Amplitude 10 V, DC mode, x10). Set R at 470  $\Omega$ .

- From the data sheet for this diode find the Zener breakdown voltage and measure the forward biased voltage drop across the diode and calculate the current. Is it what you expected?
- Now reverse the polarity of the Zener diode and measure the reverse bias voltage across the diode and  $V_{USB}$  for resistor values 1500, 1000, 470, 100, 68, and 47  $\Omega$ . Plot the current through the diode as function of the voltage across it. How does  $V_z$  measured compare to the data sheet? How does the shape of the plot compare to your LTSpice simulation?

### Exercise 3 The diode as a rectifier

The fact that a diode will only conduct when it is forward biased means it has some very interesting properties in AC circuits. One of the most useful is using the diode as a rectifier to convert AC into DC voltages and is the basis behind many different forms of power supplies that take 240 V mains 50 Hz AC voltages and then convert them to a lower DC voltage such as 5V. In the AC power section of the 1A lectures you will learn how a transformer can be used to step down an AC voltage to a lower value using a pair of mutually coupled inductors that have different numbers of turns in each coil. This can be used to step 240 V AC down to 10 V AC and then the diode is used to rectify the AC into DC. The circuit in Figure 7 shows how a diode can be used to rectify an AC signal.

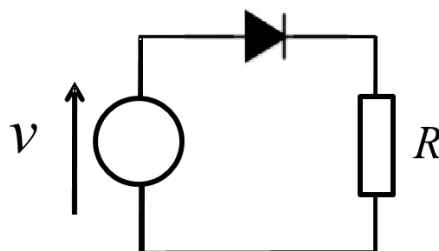


Figure 7 single diode rectifier

E7. The circuit in Figure 7 can be simulated in LTSpice using an AC sinusoidal voltage source and then analysed using the **transient** mode (**.tran**). Start with a blank schematic and import a voltage source, diode and resistor. Right click on the voltage source symbol to set its properties. Select **SINE** from the list of **Functions** and set the **DC offset** to 0 v, the **Amplitude**

to 2 V and the **Freq** to 50 Hz then click OK. Select the **1N914** diode and set resistance to be 1 k $\Omega$ . Right click on the background and select **Edit Simulation Command**. Select the **Transient** tab and set the **Stop time** to 50 mSec.

- Now run the simulation and capture a plot showing the input voltage and the voltage across the resistor.
- Measure the peak voltage and the width of the sinusoidal pulses. Are they what you expected?

E8. Now add a capacitor to the circuit in Figure 7 in parallel to the resistor. Set its value to 10  $\mu$ F.

- Capture a new plot of the voltage across the resistor. What is this capacitor doing?
- An important metric in rectification is ripple which is a measurement of the remaining AC signal that resides on top of the DC component. Investigate what happens as you increase the value of the capacitor.
- Roughly estimate what value of capacitor would you need to reduce the AC ripple to around 10 % of the overall DC voltage? Capture a plot to show your result.
- Using the same capacitor value, reduce the value of the load resistance R to 100  $\Omega$  to see it's the effect on the size of the ripple. Capture a plot but this time include the current through the load as well.
- Find the new capacitance required to reduce the ripple back to around 10 % with the new load. Capture a plot showing your solution which includes the voltage across the load resistor and the current through the diode. Comment on your results.

E9. Unplug your USB power supply from the breadboard and build the circuit shown in Figure 7 using the 1N914 diode and R = 1 k $\Omega$ .

- The AC voltage source should be the AWG set to 50 Hz sinusoid, with 2 V Amplitude.
- Set Probe A to 2 V Amplitude, DC mode, x10.
- Set the Timebase to 5 mSec/div and the Trigger to Auto with the Trigger Threshold set at 500mV to get a stable trace.
- Capture the trace on the screen and measure the peak voltage. How does this compare to the LTSpice simulation?
- Modify your LTSpice model to take into account the source impedance of the AWG. Does this improve things? Show the new schematic and simulated voltage across the diode.
- Reverse the polarity of the diode. Capture a trace to show how this effects the output voltage.

E9a – (optional challenge). In examples sheet 3 Q4 you investigated the full wave rectifier using 4 Si diodes. Using the 4 x 1N914 diodes build the circuit using a 1 k $\Omega$  resistor as the load. The challenge is to capture the fully rectified output waveform from this circuit. Hint – you need to measure the voltage across the load and not with respect to ground.

## Exercise 4 Field effects transistors

### Theory

FETs and MOSFETs are three terminal devices, and the three terminals are known as the gate, drain and source as shown in Fig. 8(a) below, and abbreviated G, S and D respectively in subsequent figures. By controlling the voltage at the gate with respect to the source, the current flowing from drain to source may be controlled. This effect is caused by the gate voltage controlling the resistance of the drain-source channel, and is frequently expressed using families of curves in which the drain current,  $I_D$ , is expressed as a function of the drain-source voltage,  $V_{DS}$ , for a variety of values of gate-source voltages,  $V_{GS}$ . This means that the drain current and source current are always the same and are often called  $I_{DS}$ . In the first exercise you will use LTspice to derive these characteristics for the enhancement mode MOSFET ZVN3310A.

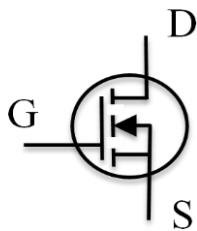


Fig. 8(a) Circuit symbol for the n-channel MOSFET

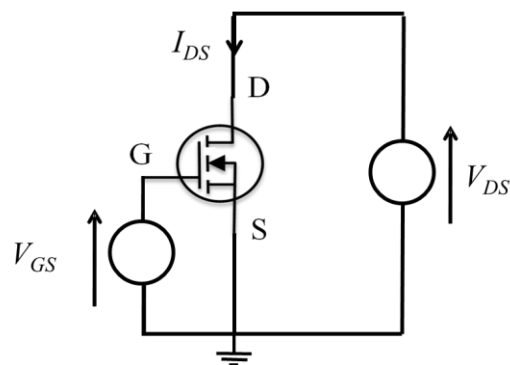


Fig. 8(b) MOSFET test circuit

### LTspice simulation

E10. Using the LTspice schematic editor set up the circuit shown in Fig. 8(b). The ZVN 3310A MOSFET is not in the LTspice library so load the schematic file ZVN3310A.asc (which can be found on Moodle) and then set up the circuit. Don't forget the ground point and set both sources to 0 V.

Next set up the simulation. To do this, use the **DC Sweep** which you used to obtain the diode characteristics in the previous exercise. In this case you need to vary two quantities,  $V_{DS}$  and  $V_{GS}$ . In the **DC sweep** menu set the first source to be  $V_{DS}$  by typing the name of the  $V_{DS}$  voltage source in your schematic into the **Name of 1<sup>st</sup> source to Sweep** box. Set the **Start Value**, **Stop Value** and **Increment** to **Linear, 0, 5 and 0.1** respectively. Now click on **2<sup>nd</sup> source** and set **Name of 2<sup>nd</sup> source to Sweep** to the name of the voltage source in your schematic which supplies  $V_{GS}$ . Set the **Start Value**, **Stop Value** and **Increment** to **1.8, 2.4 and 0.2** respectively.

- **Run** the simulation and view the results. It is the variation of  $I_D$  with  $V_{GS}$  and  $V_{DS}$  you are interested in, so right click on the plot and select **View, Visible Traces** then select **I<sub>x</sub>(M1:D)**. You will then see the family of  $I_D$  vs  $V_{DS}$  curves for different values of  $V_{GS}$ . If you want to see which colour relates to which  $V_{GS}$  value select **View, Step Legend**.

- Repeat the simulation, but this time sweep  $V_{GS}$  from 2 to 5 V in 1V steps and  $V_{DS}$  from 0 to 10 V in 0.1 V steps. What do you notice that is different?

## Exercise 4 Biasing a MOSFET amplifier

### Theory

In order to operate a MOSFET as an amplifier, it must be DC biased. Biasing is the art of selecting resistors and DC power supply voltages in order to achieve a certain operating point for the transistor. This operating point should be in the saturation region of the transistor's operating characteristics. This means that the region to the left of the knee of all the  $I_D$ - $V_{DS}$  characteristics must be avoided. Also, the gate-source voltage must be positive for an enhancement mode MOSFET.

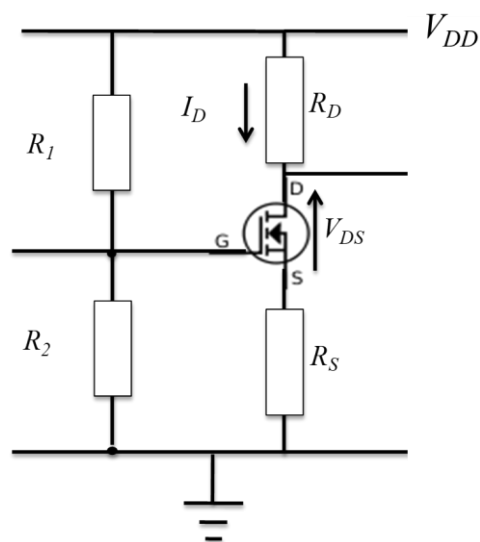


Fig. 9 Self-biased MOSFET amplifier circuit

The amplifier we want to bias is shown in Figure 9. This is a self-biasing amplifier, and it works because the potential divider holds the gate at a voltage  $V_G$  (remember that no gate current can flow in a MOSFET). The current  $I_D$  passes through  $R_S$  creating a voltage at the source  $V_S = I_D R_S$ .  $V_{GS}$  is then  $V_G - V_S$ . The voltage at the drain is then defined as  $V_D = V_{DS} + V_S$ . Often a good operating point is one which is in the middle of the  $V_{DS}$  range. If we look at the plots taken in E7 as well as the data sheet that can be found on Moodle, we can see that ZVN3310A device can operate at high voltages and currents. This is not ideal for designing amplifiers with small signals and low currents, however we can select an approximate suitable operating point at  $V_{GS} = 2.2$  V,  $V_{DS} = 2$  V and  $I_D = 15$  mA assuming a supply voltage of  $V_{DD} = V_{USB} = 5$  V.

E11. Copy the first plot you took in E10 ( $V_{GS}$  from 1.8 to 2.4V). Mark the operating point on your plot and sketch the load line. Now calculate the values of  $R_D$ ,  $R_S$ , and  $R_1$  for this operating point to give maximum voltage swing at the Drain ( $V_D$ ) and using a value of  $R_2 = 1$  M $\Omega$ . Can you explain this choice of  $R_2$ ?

E12. Set up a schematic of the amplifier in Figure 9, with  $V_{DD}$  set at 5V DC. It is useful to add labels at the gate and drain of the MOSFET to define  $V_G$  and  $V_D$  nodes respectively. To set up the simulation go to **Edit Simulation Command** and select the tab **DC op pnt**. This should leave a **.op** directive on your schematic. Run the simulation and verify your design and the choice of operating point. The result should appear in a new window which you can highlight and copy to the clipboard with the **ctrl C** command.

E13. The final test to build the circuit and test the biasing and operating point. An important factor is to adjust the resistor values to match the limited set available in the ADALP2000 kitset. The main limitation is the choice of  $R_S = 33\ \Omega$  as this is not available in the kit. If we adjust the values of the resistors then we shift the operating point, which is ok as long as it stays in the safe zone. There is no substitute for trial and error to see what best fits. The nearest value for  $R_S$  which could work is  $47\ \Omega$  which will raise that voltage at the source, which we can compensate for by increasing  $V_G$ . The value of  $R_D$  can be met using a  $100\ \Omega$  and a  $68\ \Omega$  in series and we can alter the ratio of  $R_1$  and  $R_2$  by reducing  $R_1$  to  $670\ k\Omega$  which can be made from  $200\ k\Omega$  and a  $470\ k\Omega$  resistors in series.

- Given the iteration suggested above, update your LTSpice model of the biased ZVN3310A MOSFET with the new resistor values. Run the simulation and replot the operating point and load line. Is this revision acceptable?
- Now build the circuit on your breadboard being as neat as possible (see Figure 10). Try and avoid components crossing. Measure the voltages, including  $V_{USB}$ , using Probe A, DC mode on x10, 5 V Amplitude and compare them with your LTSpice values.

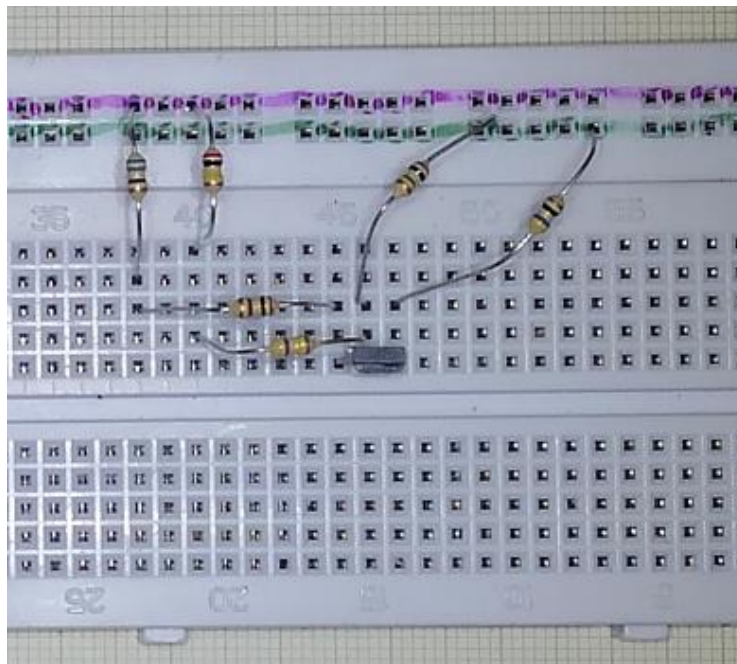


Figure 10 – Sample layout of biased MOSFET circuit

- Don't lose this circuit or the model as we will be continuing with it next term.