

ECE Tools

In Homework #9, you are to demonstrate your proficiency with two ECE tools. In reality, you probably did that already in homework sets #5 to #8. Any time you are collecting data to validate your design, you're probably using some ECE tool. This lecture covers a few ECE tools and some of the things you can do with them, including

- Oscilloscope
- Multimeter
- Frequency Counter
- Spectrum Analyzer
- Power Quality Meter
- Watt Meter

Multimeters

Multimeters give you a single number. Usually this works best when dealing with DC signals, although they can give you amplitude information for AC signals.

Some things a multimeter can do are:

RLC of circuit components: If you remove the component from the circuit (motor, resistor, inductor, etc.) you can measure the DC resistance, inductance, or capacitance.



With multimeters, you can measure resistance & voltage.
Some multimeters will allow you to measure other things (like temperature, capacitance, inductance)

For example, two 47mH inductors reads as

Inductor	R	L
#1	98.3 Ohms	45.3mH
#2	96.9 Ohms	46.6mH

Two DC servo motors read as

Motor	R	L
#1	120.6 Ohms	13.12mH
#2	103.3 Ohms	13.03mH

Using statistics, you can find

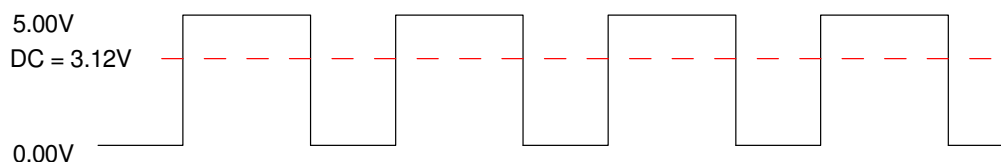
- The mean
- The standard deviation,
- The 90% confidence interval for any given inductor, motor, resistor, (2-sided t-test), and
- The probability that $R > 100$ Ohms (1-sided t-test).

Voltages: Voltages tell you a lot.

- Measuring the power supply tells you what the actual power to the circuit is
- Measuring the voltage V_{ce} for a BJT transistor tells you if the transistor is off / active / saturated.
- Measuring the voltage across a resistor allows you to compute the current through that resistor
- Measuring the DC voltage of a PWM signal tells you the duty cycle.

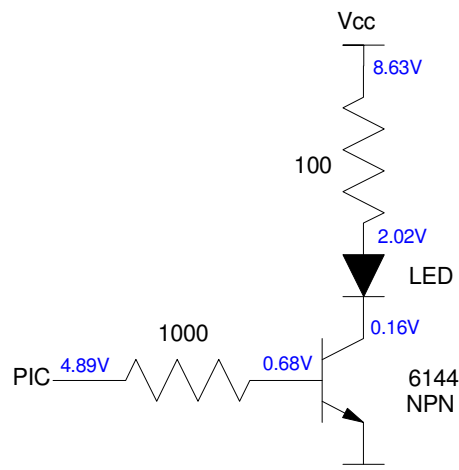
For example, if the PWM signal has a peak of 5.00V and a DC reading of 3.12V, the duty cycle is

$$Duty = \left(\frac{3.12V}{5.00V} \right) = 0.624 = 62.4\%$$



You can determine the duty cycle of a square wave by measuring its DC (average) voltage.

As another example, for the following BJT switch, assume the voltages are:



What this tells you is:

- $I_b = \left(\frac{4.89V - 0.68V}{1000\Omega} \right) = 4.21mA$
- $I_c = \left(\frac{8.63V - 2.02V}{100\Omega} \right) = 66.1mA$
- The transistor is saturated (V_{ce} is about 0.2V)
- The LED has a 1.86V drop across it (it's probably a red LED with $V_f = 1.9V$)
- The gain of the transistor is at least 15.7 (the ratio of I_c / I_b)

(Since the transistor is saturated, you don't know what the gain actually is - just its lower limit.).

You actually can tell quite a bit about this circuit with just voltage measurements.

Multimeters with AC Voltages:

DC voltage tells you the average voltage in the circuit. AC voltage tells you the RMS voltage. Depending upon the multimeter, this could be

- True RMS, or
- The peak-to-peak voltage times $\frac{1}{2\sqrt{2}}$.

For sine waves, the two are the same. For other waveforms, they are different. For example, a +/- 1V square wave has

- 1.000V true RMS, and
- 0.707V as computed as $V_{pp} / 2\sqrt{2}$

With AC readings and sinusoidal inputs, you can

- Measure the input and output of a filter at a given frequency
- Calculate the gain of a filter at a given frequency (gain = output / input)
- Measure the ripple in an AC/DC converter
- Measure the noise on the power supply

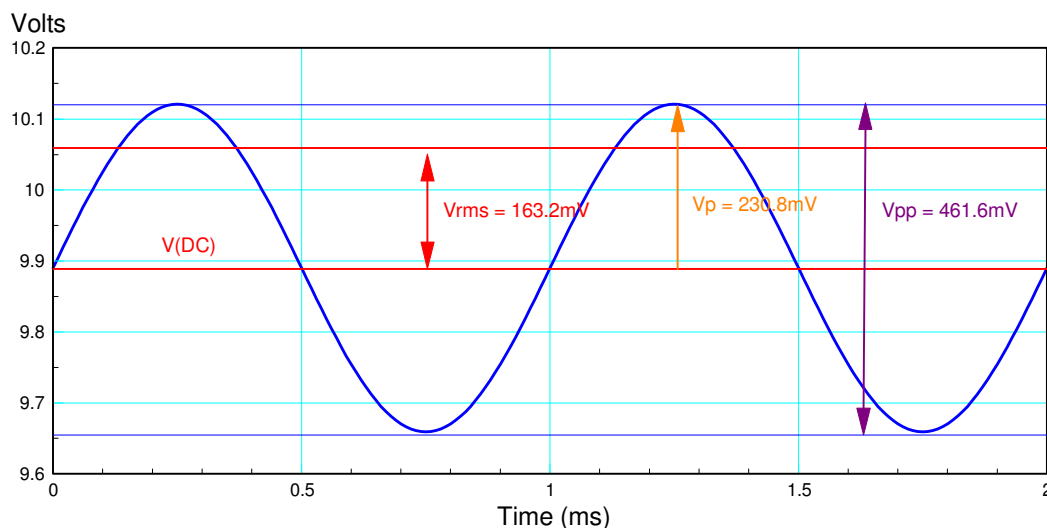
For example, with the following AC to DC converter, assume the readings at the output are:

- 9.89V DC Voltage
- 163.2mV AC Voltage (rms)

What this tells you is, assuming a sinusoidal signal,

- The average of the output is 9.89V
- The ripple is
 - 163.2mVrms
 - 230.8mVp
 - 461.6mVpp

Note that with AC signals, you *need* to specify the units. Vrms, Vp, and Vpp are all different numbers.



Signal based upon DC and AC measurements

Frequency Counter: Depending upon the multimeter, some will be able to read the frequency of an AC waveform or a square wave. Usually, they work better with TTL level square waves.

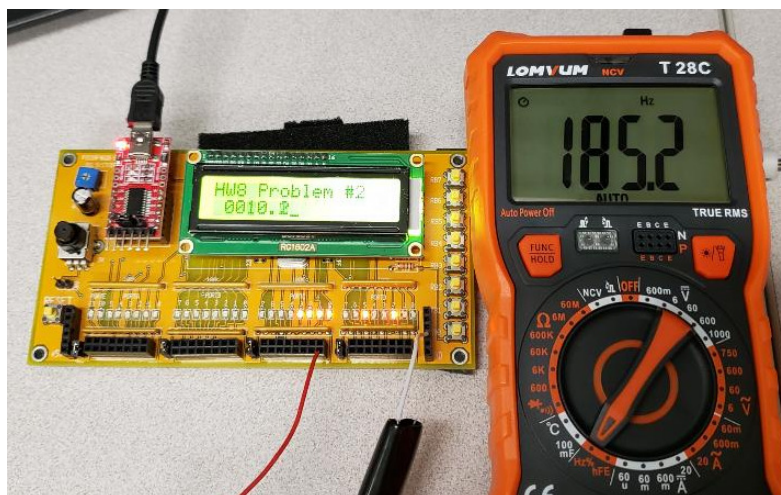
With this trick, you can measure the execution time of a program. Suppose for example you want to know how long the following program takes to execute:

```
while(1) {
  RC0 = !RC0;
  TIME = TIME + 1;
  LCD_Move(0,0);
  LCD_Out(TIME, 7, 1);
}
```

Line #2 toggles the signal on RC0 (port C pin 0). If you measure the frequency at this pin, it will tell you the execution time as

$$\text{seconds} = \left(\frac{1}{2 \cdot \text{Hz}} \right)$$

(the reason for the '2' is you need to toggle RC0 twice to get one cycle).



Using the frequency counter option on a multimeter:
The signal is 185.2Hz, meaning the pin toggles every 2.70ms (1 / 2Hz)

Transistor Gain: Many multimeters will also tell you the gain of a BJT transistor (either NPN or PNP). This is also useful if you don't know what the correct polarity is:

- The connection with the higher gain is usually the correct way to connect the transistor

For example, measuring the gain of several 3906 PNP transistors results in

NPN	Correct (hfe)	Incorrect (hfe)
#1	293	3
#2	291	9



Measuring the gain of a PNP transistor: hfe = 291

Oscilloscopes

Multimeters are really designed to measure DC signals (although you can measure the rms voltage of an AC signal). While this does provide you with some information, it's somewhat limited. Oscilloscopes, in contrast, present the voltages vs. time as a picture. This tells you a *lot* more about what's happening.

Oscilloscopes are just darned useful. They can tell you a lot about a signal.

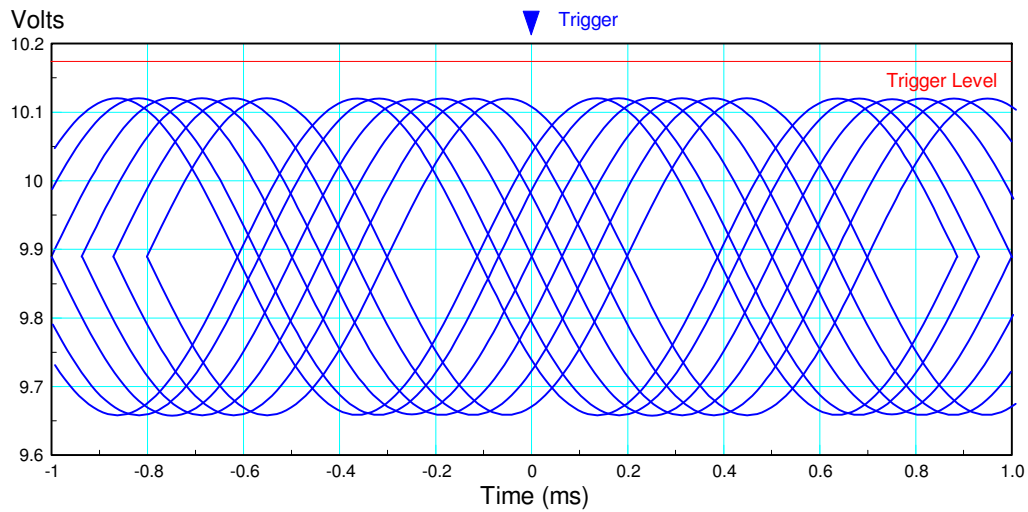
- DC and AC Voltage
- Clipping
- Phase Delay
- Time Delay (digital signals)
- Rise Time (digital signals)
- Spectral Content

Triggering: Triggering tells the oscilloscope when to start recording a signal. Most (all?) oscilloscopes have three types of triggering: normal, auto, and single

Normal: With normal triggering, you specify

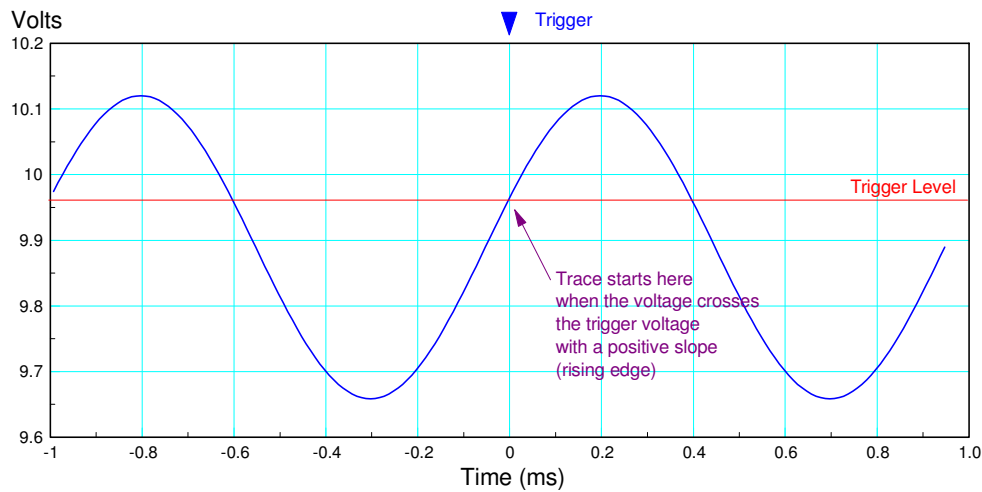
- The voltage and
- The rising / falling edge

for starting the trace. There is usually a knob that allows you to adjust the trigger voltage. If the trigger voltage is too large or small, then the trace never happens or it defaults to some frequency. This usually results in a messy display like the following:



Oscilloscope Trace when the trigger level is too high or too low

If you adjust the trigger level so that the waveform hits it, the trace will lock onto a nice clean signal that's updated over and over again

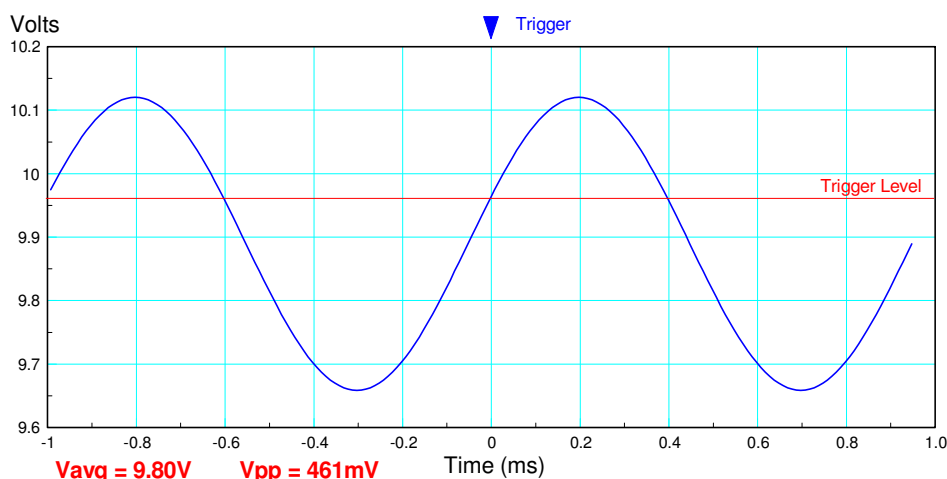


Oscilloscope Trace when the trigger level is between V(min) and V(max)

Auto: With auto-trigger, the oscilloscope tries to figure out the trigger voltage on its own. Usually it works, sometimes it doesn't.

Single: With normal trigger, the trace is updated every time the signal crosses the trigger level. Sometimes you don't want this, such as when you're looking at the transient voltages when turning on a device. In this case, setting the trigger level to *single* results in the oscilloscope taking a single reading of the trace then displaying it. If you want to try it again, there's usually a button you can press to reset the display and have it wait for the next trigger event.

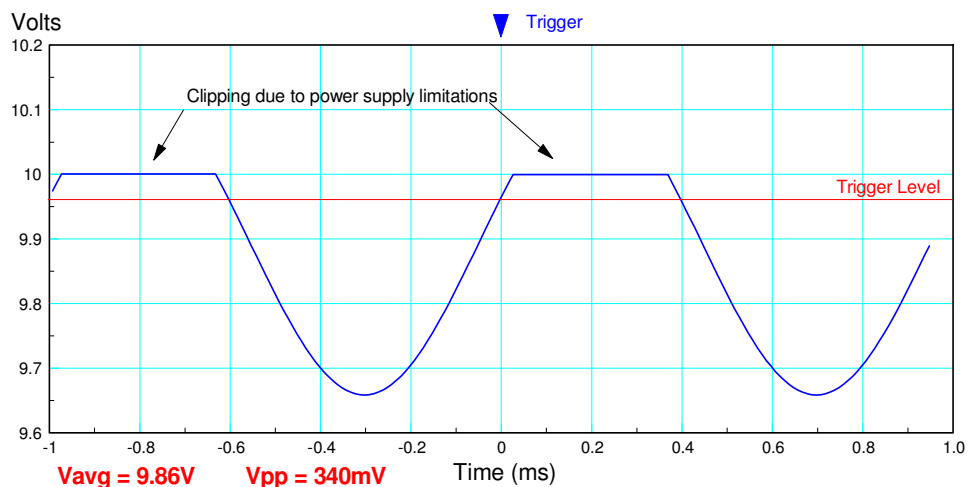
DC and AC Voltage: Oscilloscopes can be used as an expensive multimeter, telling you the DC and AC voltage. With the AC voltage, you'll need to specify the units: V_{rms} , V_p , or V_{pp} . Personally, I prefer V_{pp} for AC readings since it avoids confusion as to what you are reading.



Using the *measure* option, most oscilloscopes will tell you the voltages for a waveform

Clipping: One of the big advantages of using an oscilloscope over a multimeter is you can see what the waveform looks like. Ideally, if the input to a linear circuit is a sine wave, the output should also be a sine wave. If you're using op-amps, however, and the voltages exceed the power supply voltages, the output clips at the power supply limits.

For example if a 0V - 10V power supply is used for the op-amps, the waveform might look like the following. The resulting DC and AC readings are usually invalid in the case since the waveform has been altered by the clipping.

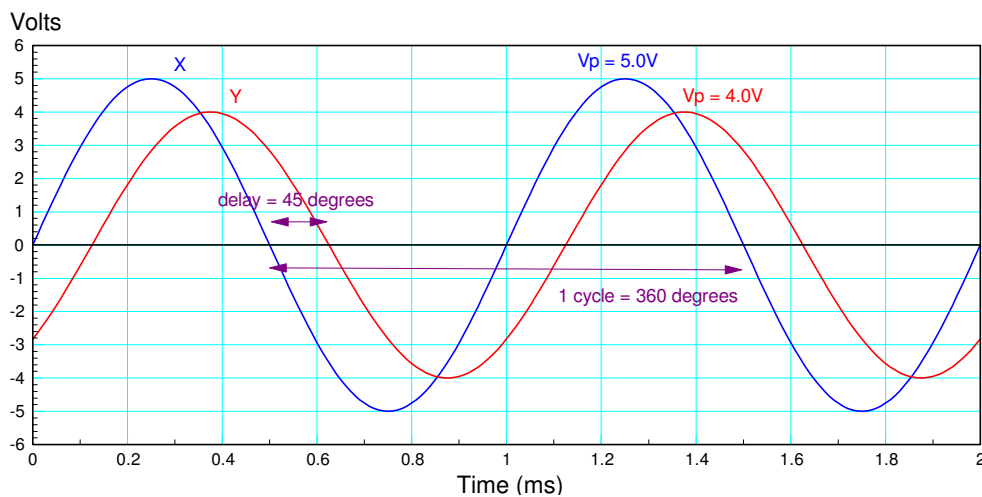


If a signal exceeds the power supply limitations, the output signal is usually distorted (clipped)

Phase Delay: If you have a filter, the gain of the filter at any given frequency has two components: the amplitude and phase (i.e. it's a complex number). Both of these show up on an oscilloscope.

- The gain is the ratio of output / input. Units don't matter as long as you're consistent.
- The phase is how much the output is delayed from the input (for a negative phase shift).

For example, if a filter has a gain of $0.8\angle -45^\circ$, the oscilloscope trace will look something like this:



$$gain = \frac{Y}{X} = 0.8\angle -45^\circ$$

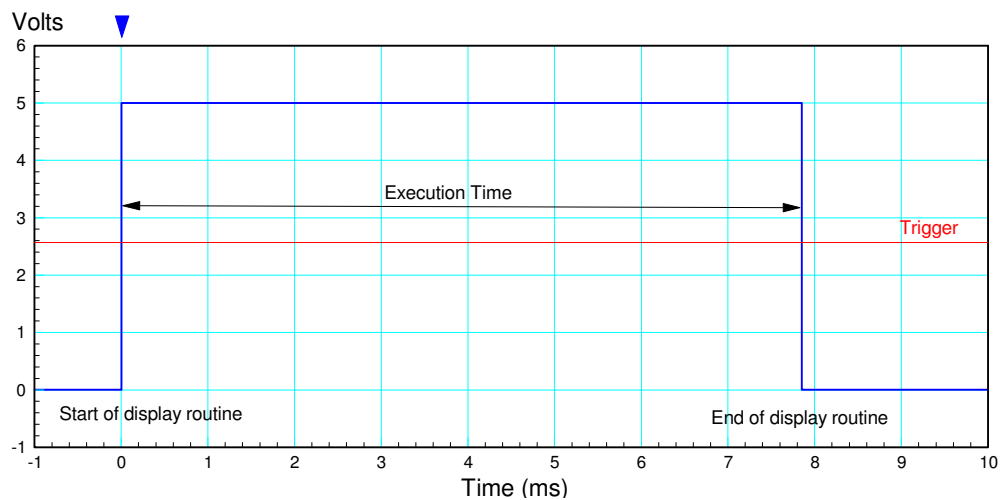
Pulse Width: With software, you can measure the executing time for a section of code. If you set a pin at the start of that section then clear it when you finish, the pulse width is equal to the executing time of that code.

Example: Determine the executing time for a LCD display routine on a PIC controller.

Code:

```
while(1) {  
    TIME = TIME + 1;  
    RC0 = 1;  
    LCD_Out (TIME, 7, 3);  
    RC0 = 0;  
    Wait_ms(10);  
}
```

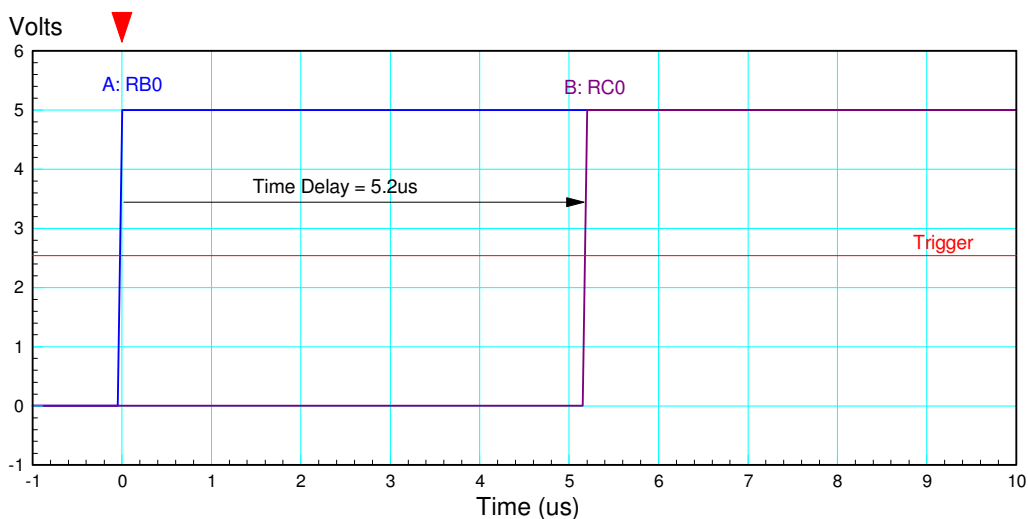
On an oscilloscope, if you look at pin RC0, the signal will look similar to this:



The execution time of a program can be measured on an oscilloscope

Time Delay (digital signals): With digital signals, you can measure the delay between two signals. For example, suppose I want to know how much time elapses between when I press a button and PIC microcontroller responding.

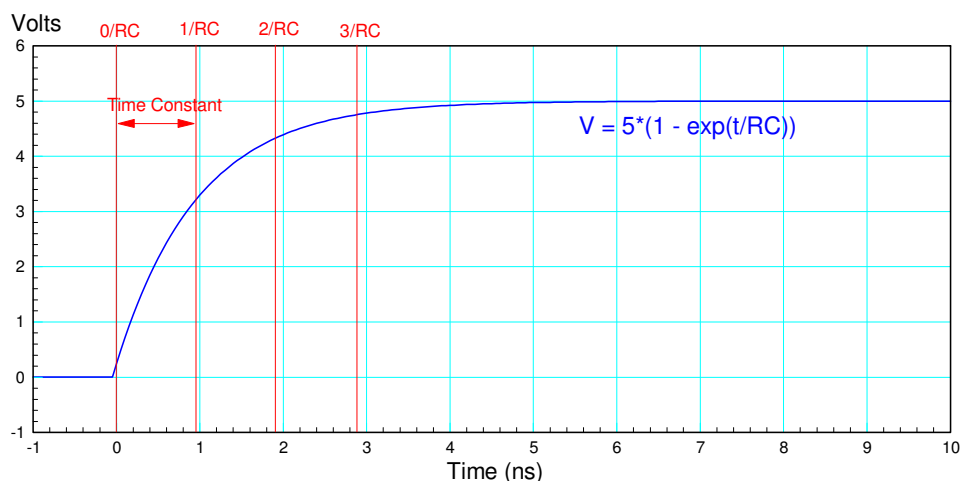
- When pin RB0 goes high, an interrupt is triggered
- When the interrupt happens, pin RC0 is set



The time delay can be measured using an oscilloscope Here, it takes 5.2us for a PIC to respond to an input

Rise Time (digital signals): All traces have some capacitance and resistance. This causes the voltages to rise and fall with an RC time constant. The capacitance may be very small (meaning the RC time constant may be very short). It's there, none the less.

If you zoom in on the time scale, you can usually see this as something like the following:



If you zoom in on the time axis, you can usually see the voltages rising and falling exponentially