

# Breadboards

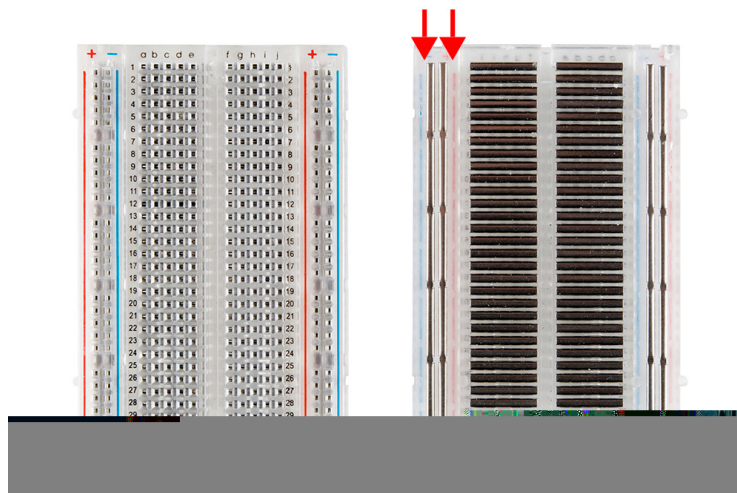
Once you have your circuit

- Designed on paper, and
- Tested in simulation (CircuitLab)

you're ready to test your design in hardware. Breadboards are an easy way to build your circuit and test your design. They're also easy to modify and change: components can be easily added and removed from a breadboard.

Most of the breadboard used in ECE are 830 tie breadboards. These have

- Four edge connectors that are shorted along the length of the breadboard (two left side, two right side). These are usually used for power and ground.
- Two sets of connectors in the middle (one left and one right of the center bar). These are shorted horizontally.
- Across the middle is an insulator - this separates the middle connectors by 300 mils: the width of a typical IC,

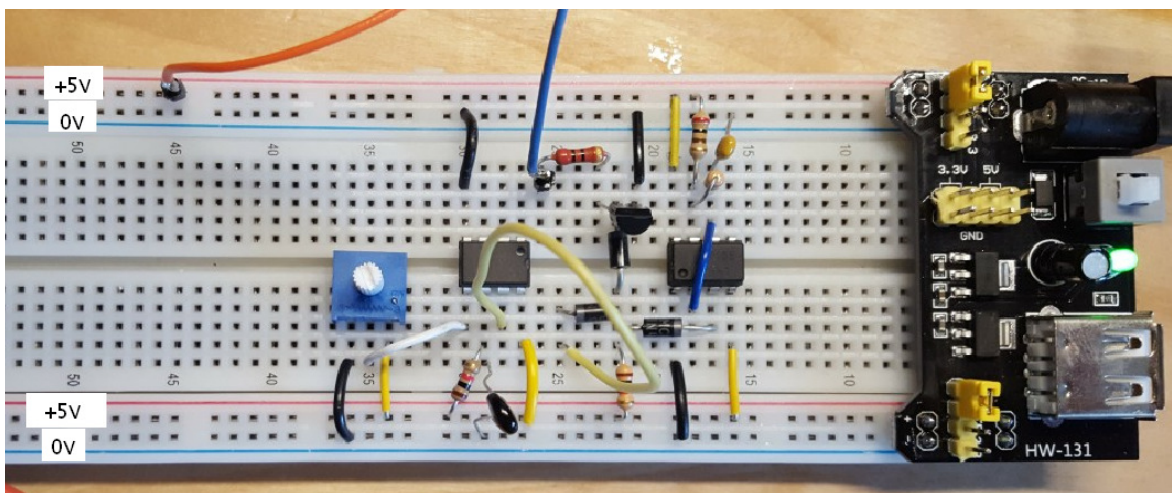


Breadboard Top (left) and back side (right)

For example, the following breadboard circuit uses

- The red trace along the top and bottom as +5V
- The blue trace along the top and bottom as 0V
- Two IC's go across the middle divider.

The four pins above and below each IC then allow you to connect to that pin



Typical Use of a Breadboard.  
Top and bottom rails are used for power and ground  
ICs go across the center divide  
Each column of pins above and below the IC connect to that IC's pin in that column

## Purpose of Breadboarding:

The purpose of a breadboard is verify your design works in practice. Typically, the design process includes:

- Step 1) Paper Design
  - In theory it works...
- Step 2) Simulation (CircuitLab)
  - Check your design (with models)
- Step 3) Breadboard
  - Check your actual design
  - What's the bare minimum I need?
- Step 4) PCB
  - More permanent & abuse tolerant
  - Less noise sensitive
  - Smaller, easier to package

With a breadboard, you can quickly build your circuit, and if you need to tweak it, components can be swapped out or moved fairly easily. Once you've finalized your schematic with a breadboard circuit, you're ready to lay out a printed circuit board (PCB).

---

# Tricks of Breadboarding

## 1) Keep Your Circuit Neat

- Use short wires
- Use short component leads
- Organize your breadboard into sections

Keeping your wires short

- Reduces the noise picked up by your wires
- Reduces the chance of a wire falling out
- Helps you see the wiring in your board
- Helps when you need to modify your breadboard circuit.



Do use short wires. Keep your breadboard neat and organized.  
The circuit on the left will be very noisy and very hard to debug

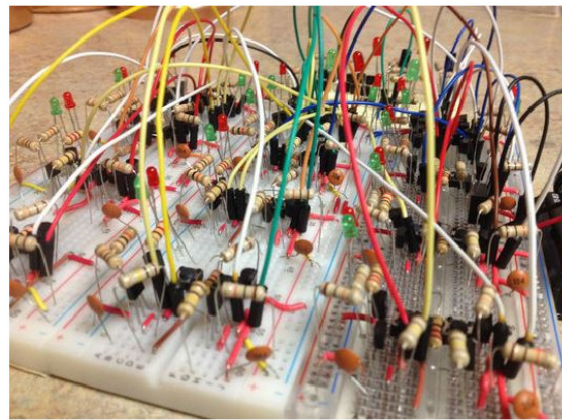
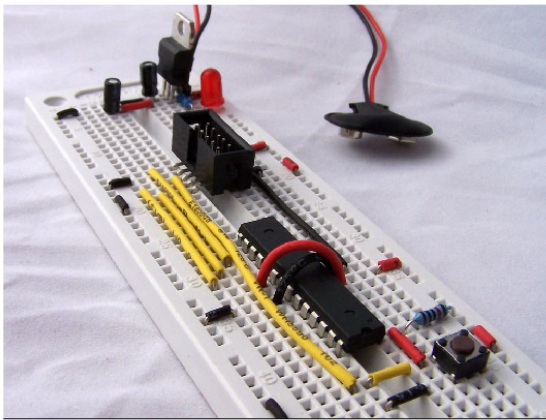
---

## 2. Color Code your Wires

- Use red wires for +5V
- Use black wires for ground
- Use different colors for different types of signals.

By color coding your wires,

- You can quickly spot if a chip is missing power and/or ground.
- You can quickly see if a signal wire is missing between two ICs



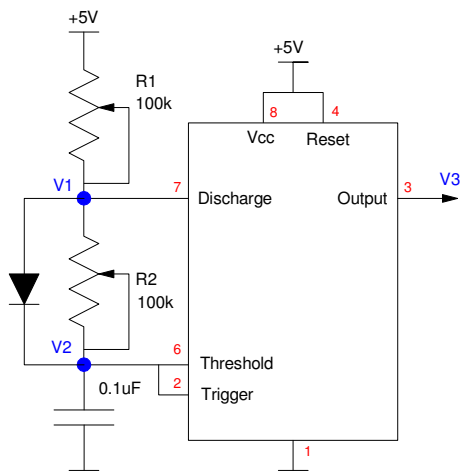
Do color code your wires. It makes it much easier to spot missing wires or wires going to the wrong spot.

Better chance of seeing which wire is in the wrong spot

### 3. Use Potentiometers (2 max)

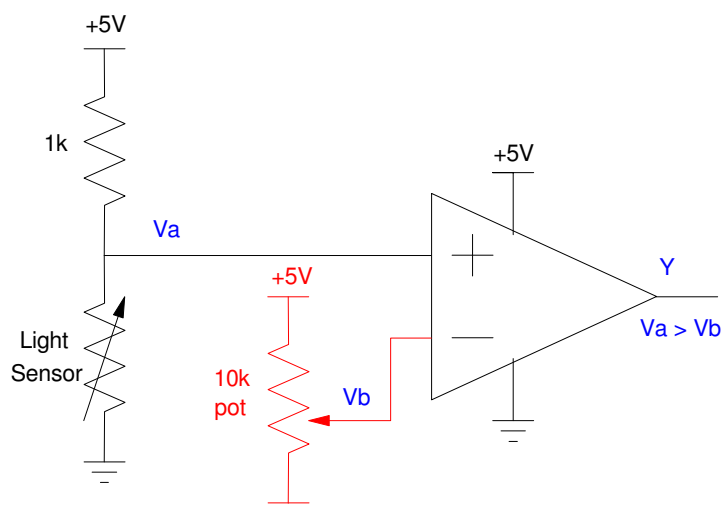
Often times, you're going to need to adjust a resistor in order to make a circuit work. For example, when using a 555 timer, the resistors R1 and R2 set the on and off times. If you replace these with potentiometers, you can adjust them on your breadboard to tune your circuit and get the on/off times within your tolerances

For example, if R1 and R2 are calculated to be 72k and 52k Ohms respectively, you could use two 100k potentiometers connected as follows:



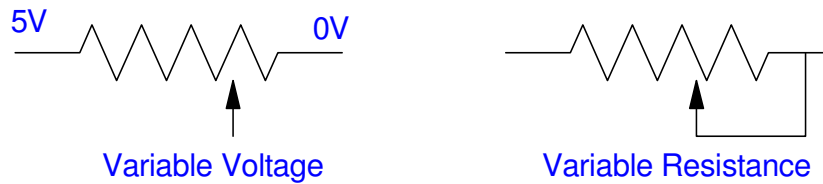
By replacing R1 and R2 with potentiometers, you can adjust each over the range of 0k to 100k

Potentiometers can also be used to provide an adjustable voltage. For example, the following circuit uses a comparator for a night light. The potentiometer allows you to adjust the voltage at Vb from 0V to 5V - allowing you to adjust the light level where the night-light turns on and off.



Pots allow you to adjust a voltage

The basic circuit for a pot when used as a variable voltage or resistance is:



Two common configurations for potentiometers

Potentiometers are a little expensive, however:

- Resistor = \$0.02
- Pot: \$1.55

So, for ECE 401, please limit your circuit to using two potentiometers or less.

## 4. Breadboards & Test Points

Some things to think about when using a breadboard are:

- How do you test your circuit?
- What signal do you look at?
- What should the signals look like?
- What procedures do you use?

Note what signals you look at and record what you read. This affects your upcoming PCB layout:

- These same signals should be measured in simulation and on your PCB
- Test points should be added to your PCB so that you have access to these signals.

## 5. Keep Your Breadboard

When done testing your breadboard, keep it together, intact (i.e. don't cannibalize it for parts). If your PCB doesn't work properly, your (working) breadboard circuit will be helpful in debugging what part of your PCB works (and has similar signals), and which part does not work.

This means you'll need two of every part in ECE 401

- One for your breadboard circuit, and
- One for your PCB.

That's OK.

## Test Points

Once you build your breadboard, you need to test it to make sure it works. When you do so, pay attention to what signals you're looking at. When you build your PCB, you need to make sure you have access to these signals (i.e. add test points to your PCB).

Also also, in OneNote, record

- The conditions of the test,
- What you are measuring (show on the schematic), and
- What the output was (usually a multimeter or oscilloscope reading)

This will be needed later on when you're testing your PCB.

Each project must include at least six test points. These include

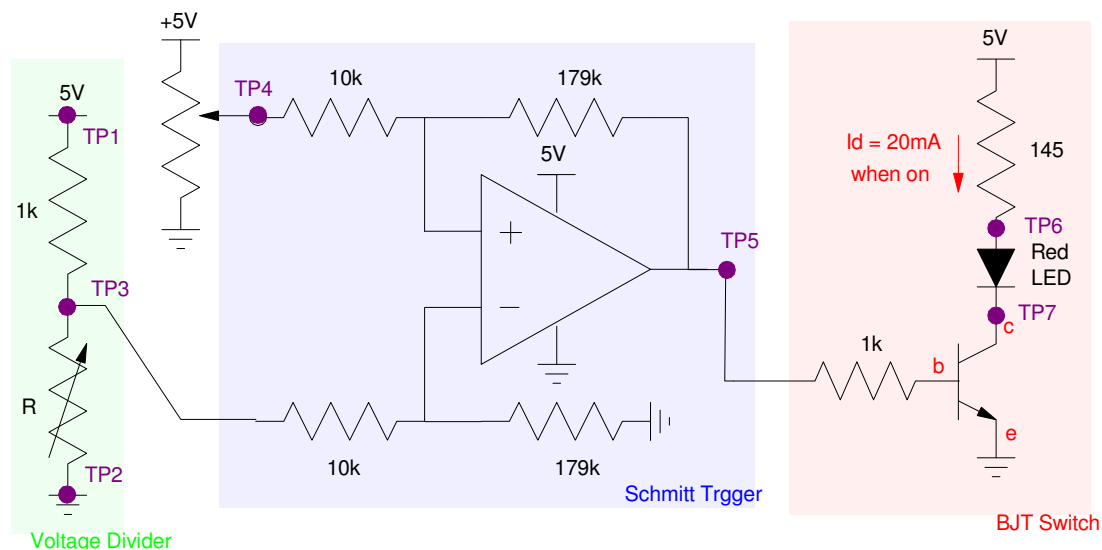
- 9V power
- 5V power
- Ground (really important - everything is measured relative to ground)
- The collector of each transistor (needed to check if the BJT is saturated)

Other signals are OK to include as well: this is just the minimum.

Just pay attention to what you are measuring when you verify your breadboard is working.

- These measurements should be recorded in your OneNote document,
- Your PCB should include test points at each of these signals as well.

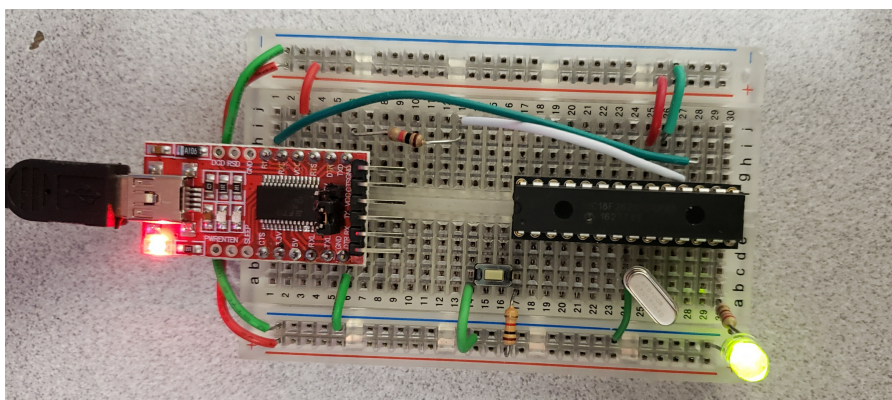
For example, with the Schmitt Trigger from before, eight test points would be useful when testing and debugging the circuit:



One possible set of test points for a Schmitt Trigger

- TP2: Ground. You need ground since all voltages are relative to ground
- TP1: +5V. If you don't have 5V, your circuit isn't going to work.
- TP3: The voltage should vary as temperature changes
- TP4: The voltage where the Schmitt trigger turns on.
- TP5: Check that the Schmitt trigger is turning on (5V) and off (0V)
- TP6: Lets you determine the current through the LED:  $I_d = (5 - V_6)/145$
- TP7: Check that the transistor is saturating when the LED is on ( $V_7 = 0.2V$  if saturated)

## Breadboards with a PIC18F2620



In ECE 376, we use a PIC18F4620 (40-pin PIC) along with a boot loader for downloading code. The smaller cousin of this chip is the PIC18F2620 (28-pin version).

If you use a different PIC chip, you'll need

- A C-compiler for that chip,
- A way to program that chip, and
- To dig through the data sheets to determine how to turn on/off interrupts, A/D conversions, etc.

If you use either the 18F4620 or 18F2620, everything you learned in ECE 376 applies. Likewise, it is *strongly* recommended that you use one of these chips.

This document gives the minimum of what you need to get a PIC18F2620 up and running on a breadboard. This can also act as the core of a PCB layout. Note that the I/O pins on a PIC are limited:

- Logic 0 = 0V
- Logic 1 = 5V
- Max Current = 25mA

If you need more than 25mA, you need to use a transistor as a buffer (same as with a 555 timer).

To give something concrete to go by, give the hardware that allows an LED to blink on and off.

## Hardware Design:

Select a PIC18F2620 since

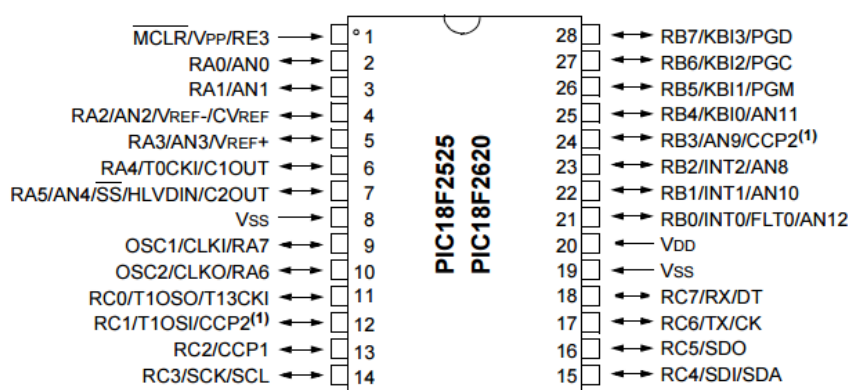
- We have a boot-loader for this chip (same as used in ECE 376)
- We have experience using this chip (same as ECE 376)
- We have a C compiler for this chip (same as ECE 376)

The only difference between the 40-pin version used in ECE 376 and the 28-pin version used in ECE 401 is

- You have 22 I/O pins with the 28-pin version (vs. 33 I/O pins), and
- PORTD and PORTE are not connected to any I/O pins with the 28-pin version

Otherwise, they're the same.

### 28-Pin SPDIP, SOIC



Pinouts for a PIC18F2620

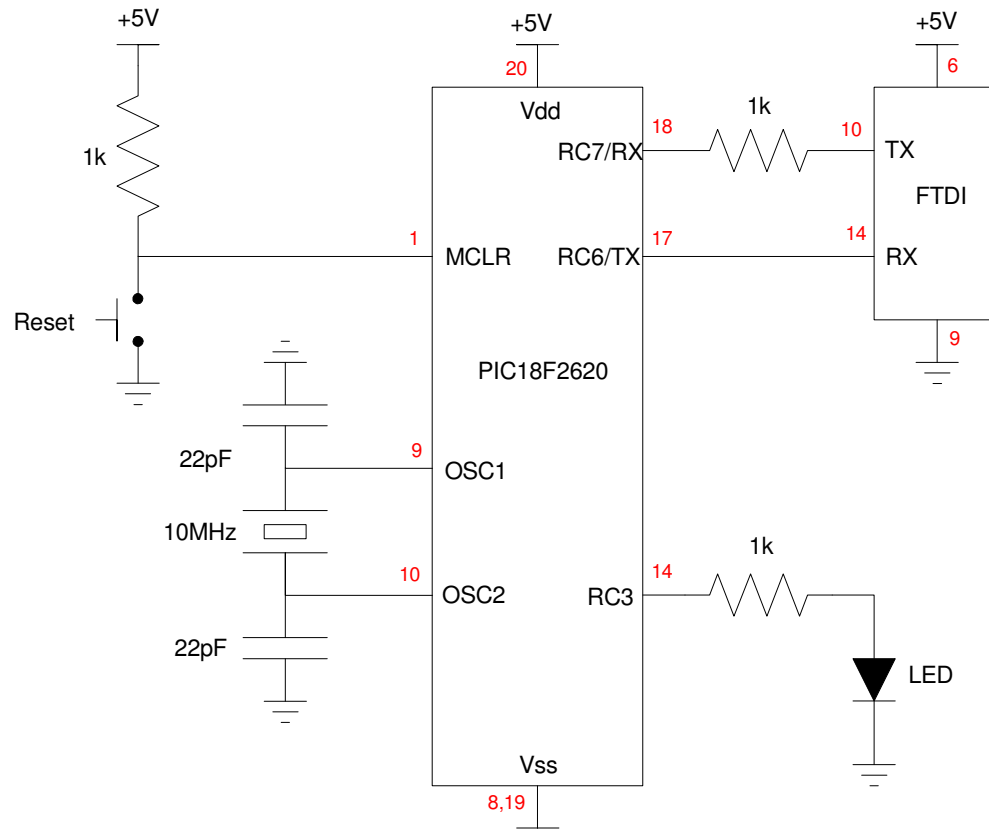
When you design a system around a PIC processor, you need to identify the function of each I/O pin. With this processor, there are three I/O ports: A, B, and C. If a single LED is connected to PortC pin 3, the pin assignments could be something like this:

PORTA							
x	x	5	4	3	2	1	0
n/a	n/a	-	-	-	-	-	-

PORTB							
7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-

PORTC							
7	6	5	4	3	2	1	0
TX	RX	-	-	LED	-	-	-
Out	In			Output			

A schematic for this minimal setup:



Schematics for getting a PIC to run and drive an LED on PORTC pin 3.

The 22pF capacitors are not necessary: there is enough stray capacitance for the clock to function if these are removed

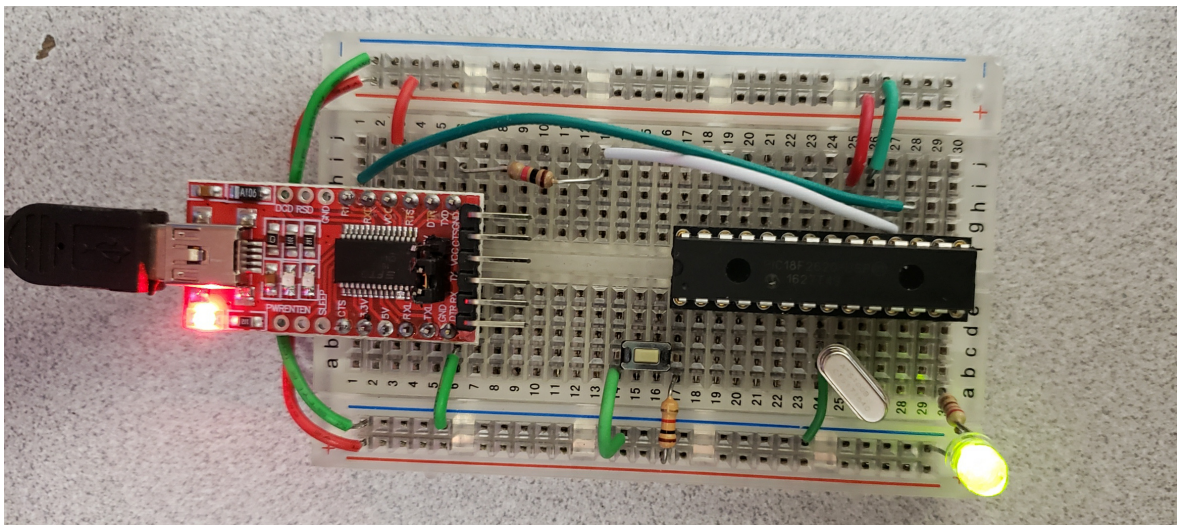


Photo of breadboard for the above schematic

## Major Components:

**Digi-Key** All Products United States | 1-800-344-4539 Change Country English USD

[PRODUCTS](#) [MANUFACTURERS](#) [RESOURCES](#) [LIVE CHAT](#) 0 item(s) [Login or REGISTER](#)

[Product Index](#) > [Integrated Circuits \(ICs\)](#) > [Embedded - Microcontrollers](#) > Microchip Technology PIC18F2620-I/SP

[Add To Favorites](#) [Share](#) [Link](#)

This part can be programmed by Digi-Key; for details please contact our custom department at 1-800-344-4539 x5725 or [custom.orders@digikey.com](mailto:custom.orders@digikey.com)



### Product Overview

Digi-Key Part Number	PIC18F2620-I/SP-ND
Quantity Available	1,169 Can ship immediately
Manufacturer	<a href="#">Microchip Technology</a>
Manufacturer Part Number	PIC18F2620-I/SP
Description	IC MCU 8BIT 64KB FLASH 28SDIP
Lead Free Status / RoHS Status	Lead free / RoHS Compliant
Moisture Sensitivity Level (MSL)	1 (Unlimited)
Manufacturer Standard Lead Time	4 Weeks
Detailed Description	PIC PIC® 18F Microcontroller IC 8-Bit 40MHz 64KB (32K x 16) FLASH 28-SPDIP

### Price & Procurement

Quantity

PIC18F2620-I/SP-ND

Customer Reference

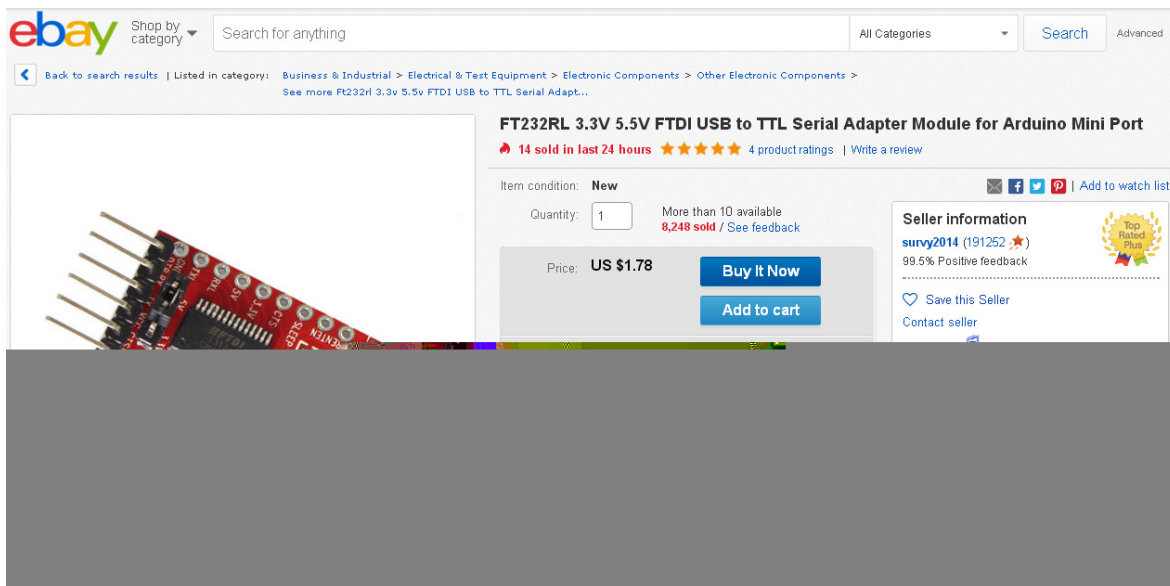
Add to Cart

All prices are in USD.

Price Break	Unit Price	Extended Price
1	5.29000	\$5.29
25	5.16600	\$129.15
100	5.04000	\$504.00

Submit a [request for quotation](#) on quantities greater than those displayed.

Processor: PIC18F2620 (www.Digikey.com)



FTDI: (www.ebay.com: Search for "Arduino FTDI")

-