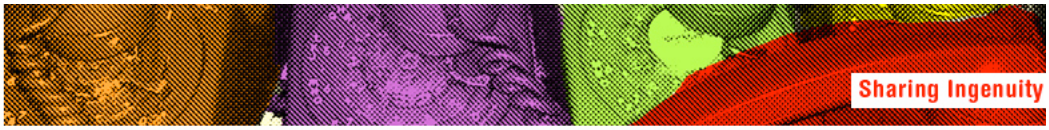

Beginning Embedded Electronics - 11

Skill level: Beginner

by Nate | June 19, 2008 | *2 comments*

Common Mistakes, Tips and Tricks

1. All grounds need to be connected together.
2. TX/RX loop back trick: When in doubt of a serial conversion circuit, short the TX and RX pins together to get an echo.
3. Normal length wires for breadboard connections: Don't use a 9" wire where a 2" wire will do.
4. Minimize short potential in your breadboard wiring: Don't expose an inch of wire from the insulation if all you need is 1/4".
5. You will learn best when you have a *simple* project to work on. Don't create the 'house-pet robot' just yet.
6. Google is, of course, your friend. When you don't know, go do some research.
7. for(x = 0 ; x < 400 ; x++) : If x is defined as an 8-bit integer, the for loop will loop forever!
8. Soldering basics: Wet your @#$% sponge.
9. Take your time with ground plane solder joints. Do not be fooled by a cold joint.
10. Never trick yourself into thinking you're that good. Print out a 1:1 and compare the footprints!
11. Check that TX and RX are wired correctly to all peripherals. TX/RX swap is the one of the greatest causes of PCB failures.
12. When laying out a PCB with SMD micros, don't forget to include the programming port!
13. Don't run silkscreen across pads.
14. Connector PCB footprint mis-numbering: always check the pin number on your connector - they can have very obfuscated schemes.
15. In Eagle, use vector fonts only!
16. Review your gerber files before submitting them.

Lecture 1 - Background and Power Supply

Lecture 2 - How to Get Code Onto a Microcontroller

Lecture 3 - What is an oscillator?

Lecture 4 - UART and Serial Communication

Lecture 5 - AVR GCC Compiling

Lecture 6 - Soldering Basics

Lecture 7 - SMD Soldering

Lecture 8 - Eagle: Schematics

Lecture 9 - Eagle: PCB Layout

Lecture 10 - Eagle: Creating a new part

Common Mistakes, Tips and Tricks

Back to Tutorials

Comments

2 comments

Log in to post comments.

by Tracekill +1 | June 25, 2009 at 0:47 AM 0

Hey guys. These are a great selection of tips, I found a few of them useful even at my early stage. As a computer science student however, I have to know: Why does the for-loop in 7 increment indefinitely? I'm not very intimate with the characteristics of an 8-bit integer. With the data sets we work with, I find myself more commonly using 16- and 32-bit integers. Is it simply because of the limited range of the 8-bit integer? If so would not for(x = 0; x < 256; x++) also loop indefinitely? Great advice and great tutorials. I've spent nearly $100 at this site already, and it was worth every penny.

by **Keegan** **+1.1** | **July 4, 2009 at 3:44 AM** **0**

If an 8-bit integer is 255, and you increment it, it will rollover and become 0. 8-bit integers can only hold a value from 0 to 255. The condition part of the loop would have to be "x < 255" or lower.