



## FSUIPC: Lua Tutorial

### For Microsoft Flight Simulator and GoFlight Equipment



Home

"There are 10 types of people in the world, those who ready binary and those who don't!"

About Lua

Binary

If you read that as ten people, you don't read binary. Binary is a mathematical system based on 2. Why? Because computers are stupid and can't count any higher. Actually the real reason is because they run on electricity. It's either On or Off. That's it. Look at Morse Code, long or short.

Executing Code

Bits, Bytes and all the other stuff

Binary

Did you know that using the binary system, you can count to 31 with ONE hand? Have you ever counted binary before? Open up the calculator on your computer. Click View and click Scientific. Enter your age... Click the Bin button. Mine comes out as "101010" The trick is to read from right to left.

Hexadecimal

Normally, we use one finger to represent one number... "1". Pick a number from one to five and then hold up that many fingers. How many fingers do you have up? You just added one, that many times.

Bits & Bytes

Now what if... your index finger represents the number "1" and your middle finger represents the number "2", doubling every finger. Got it? Stay with me. Remember, each finger "represents" a different number.

Do you need to see it?

Memory Offsets

Let's say 0 is a down finger and 1 is an up finger...

We start with 0000 = 0

Variables

The Code

The Editor

Get Started

FlightSim

Cleaning Up

Assignments



Then 0001 = 1

## RESOURCES

[FSUIPC Support Forum](#)

[GoFlight Inc.](#)  
[GoFlight Support Forum](#)

[Lua Website](#)  
[Lua Reference Manual](#)  
[Lua for Beginners](#)  
[LuaEdit](#)

[Notepad++](#)  
[Online Documentation](#)  
[Support Forum](#)

[Tutorial Files](#)



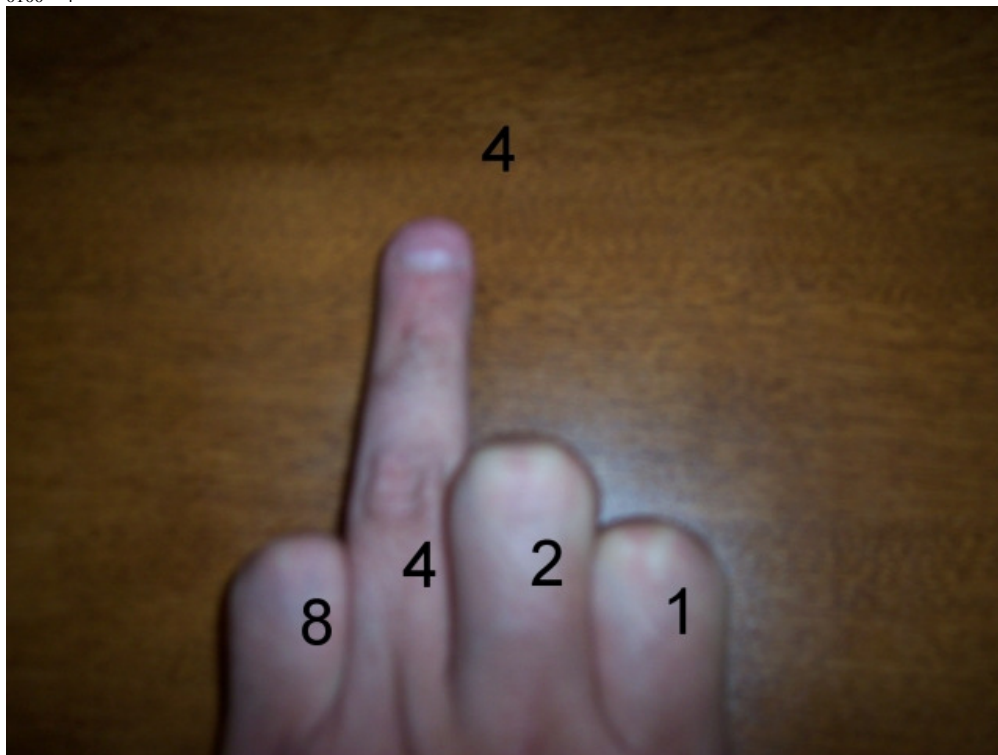
0010=2



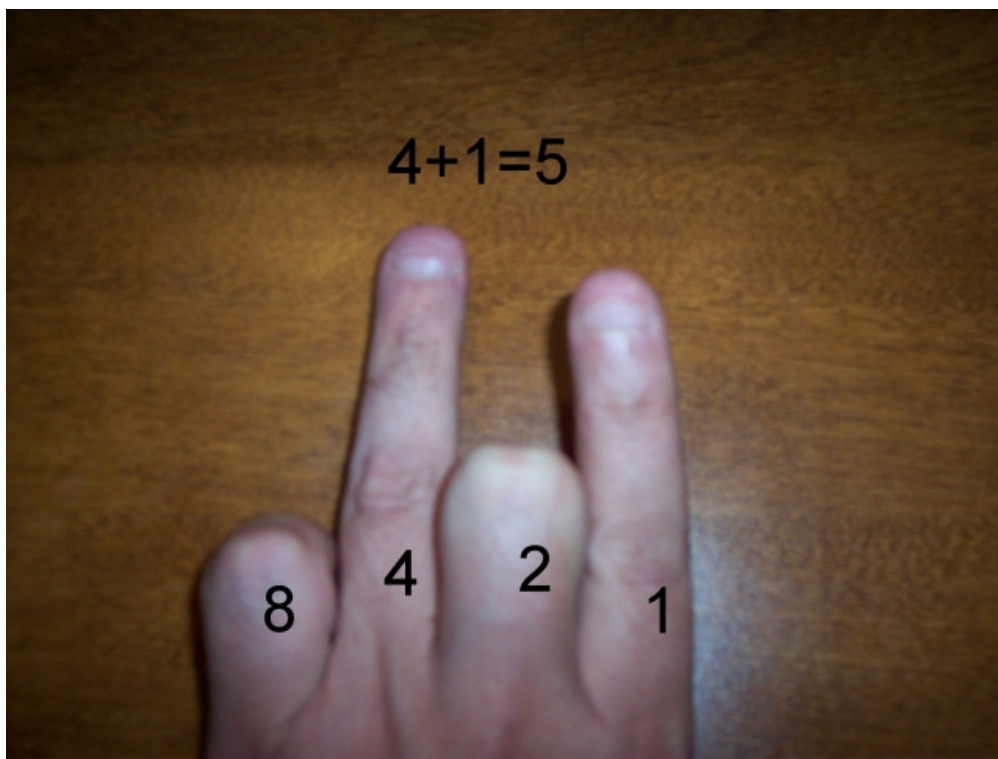
0011=3



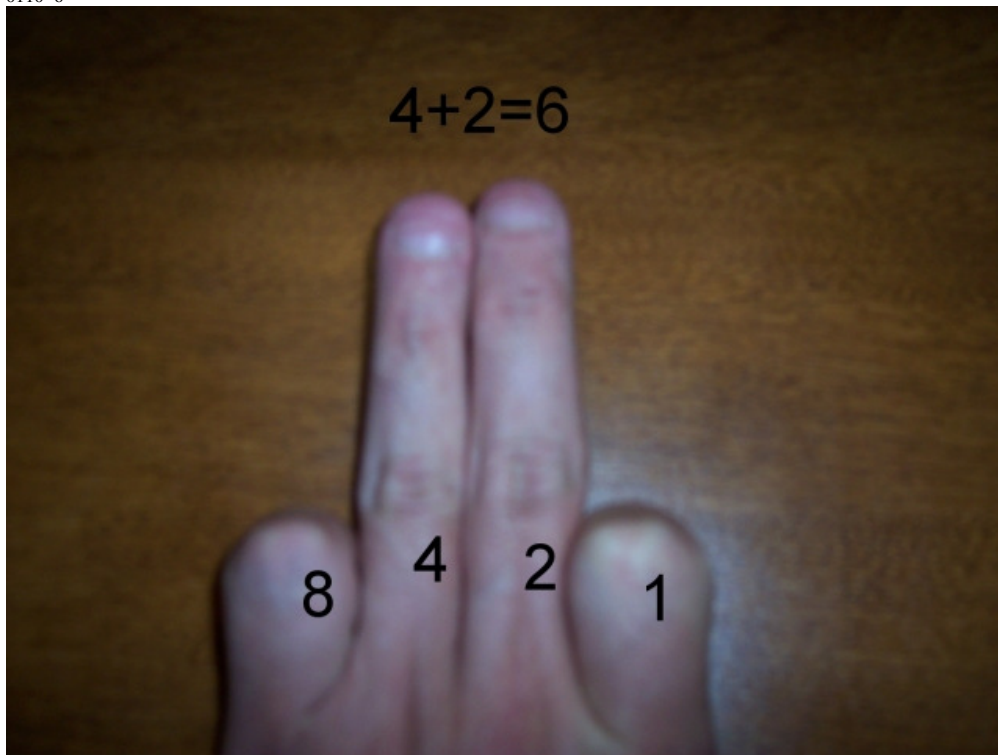
0100 = 4



0101=5



0110=6



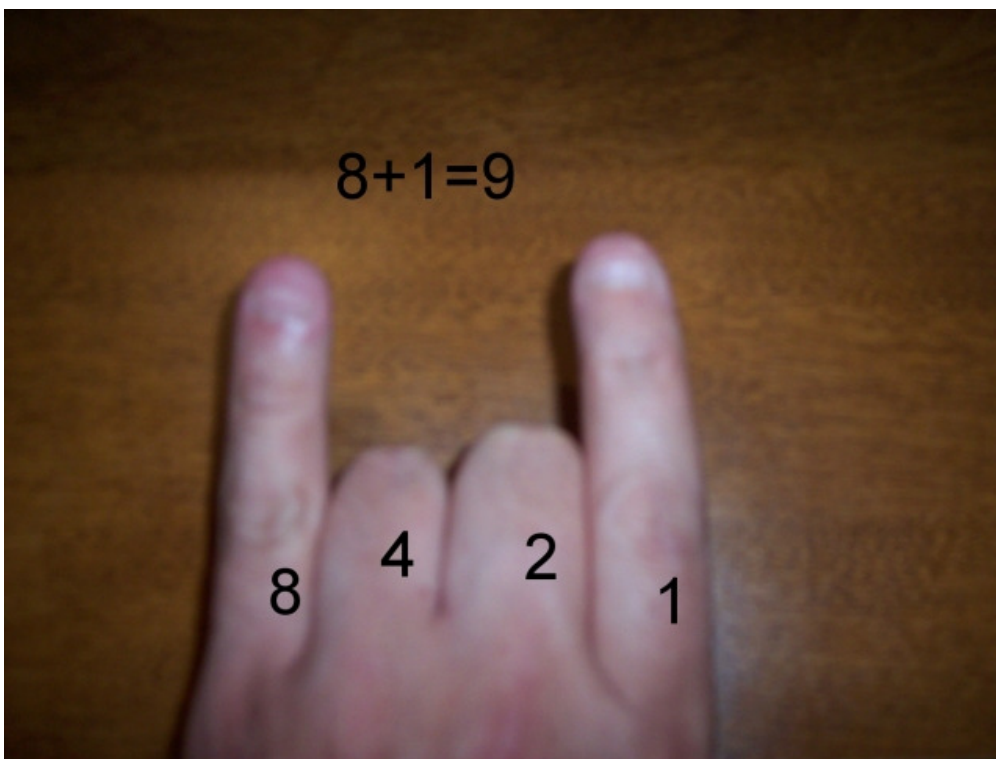
0111=7



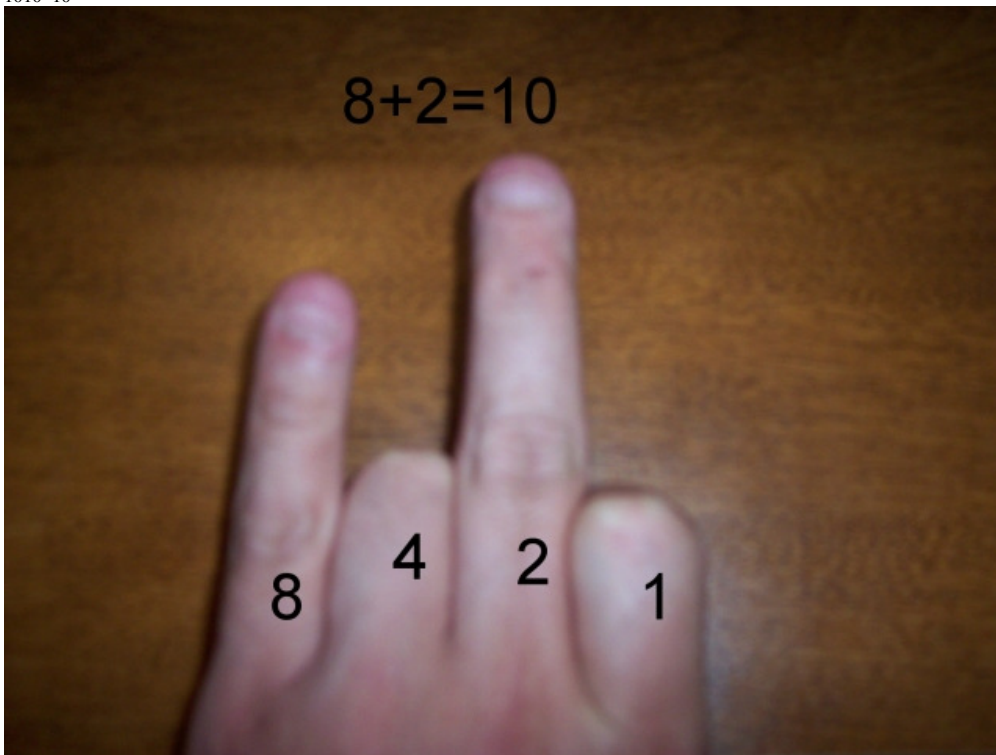
1000=8



1001=9



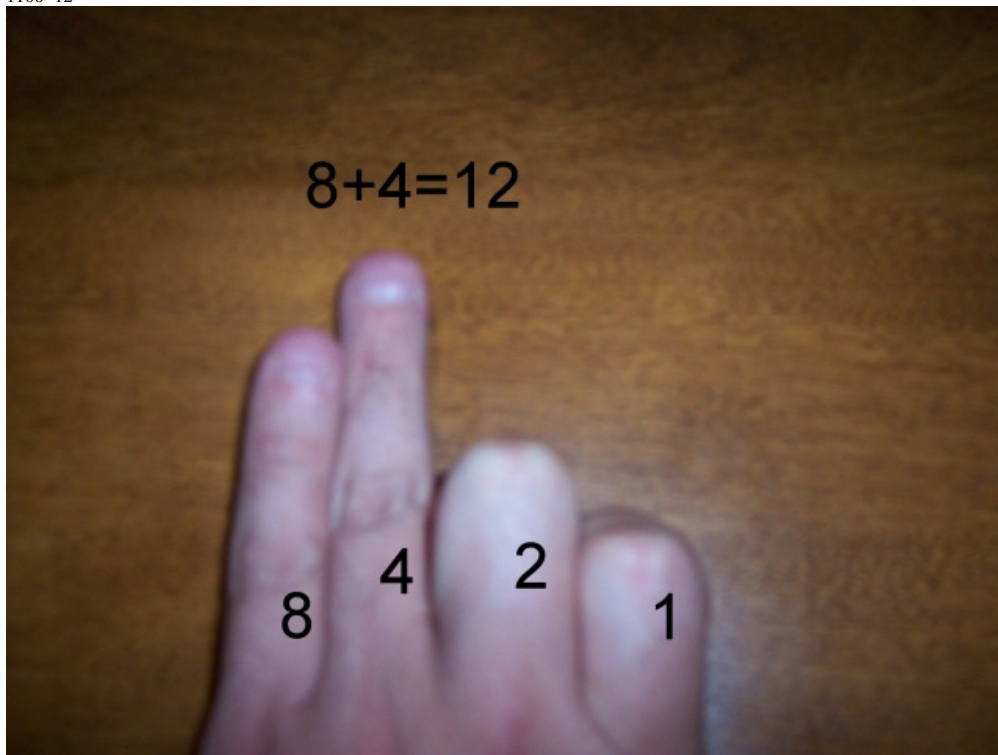
1010=10



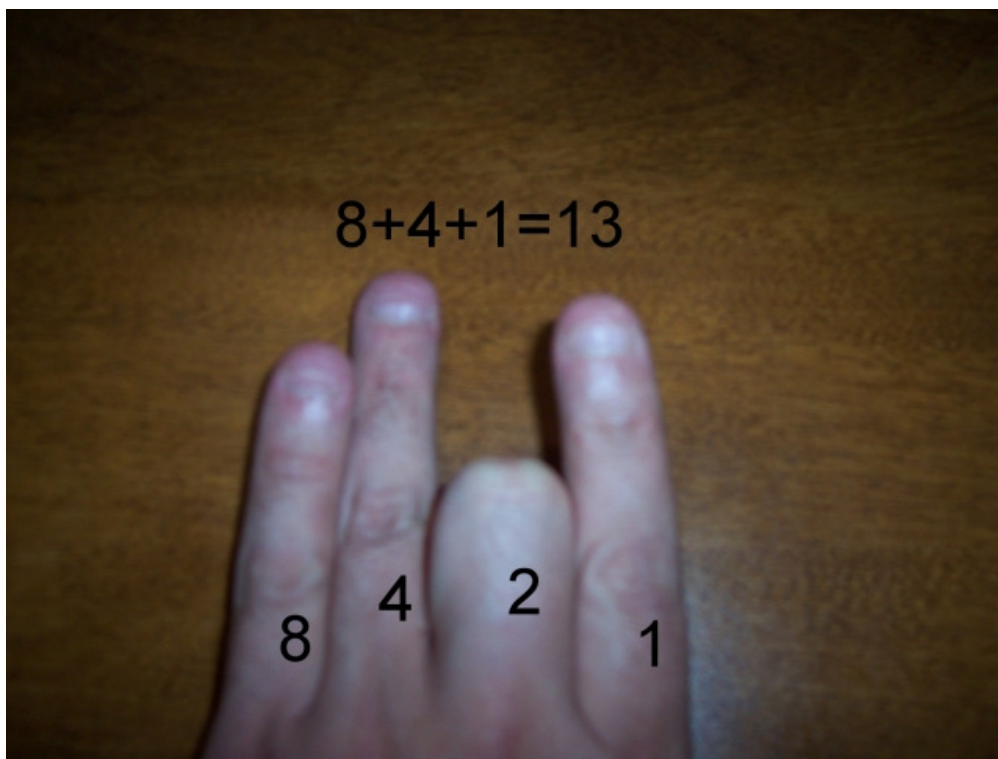
1011=11



1100=12



1101=13



1110=14



1111=15



I hope I didn't loose anyone... we need to move on to Hexadecimal

[Next Chapter](#)