



FSUIPC: Lua Tutorial For Microsoft



Flight Simulator and GoFlight Equipment

Home

About Lua

Executing Code

Binary

Hexadecimal

Bits & Bytes

Memory Offsets

Variables

The Code

The Editor

Get Started

FlightSim

Cleaning Up

Assignments

FlightSim

Now that we have basic commands to control the GF modules, Let's see if we can get our script to monitor a FlightSim memory offset. In the last example we simply had a light turn on, now we are going to make FS turn that light ON or OFF.

This time, Let's use a GFT8 for our example. Open up your copy of the "FSUIPC4 Offset Status" and locate the entry for "Pitot Heat Switch". You should find it with a memory offset of "029C"

We are also going to introduce using "functions" and "event monitoring". Open your editor and type in the following lines

```
function PitotHeat(offset, value)
end
```

We've just created a function block called "PitotHeat". Functions will not run by themselves. They must be called! So how do we "call" a function? Add the following line...

```
event.offset("029C", "UB", "PitotHeat")
```

This "event.offset" line, MUST be after the function it's calling. Now let's break down this line... "event" is a library containing numerous commands. "offset" is a command to look at a FS memory address. "029C" is the memory address we want to look at. "UB" is the memory address type, an Unsigned Byte, "PitotHeat" is the name of our function.

But there's something missing. There's no code in the function, so it has nothing to do. Let's add some room between the "function" line and the "end" line. Your file should look like this

```
function PitotHeat(offset, value)

end
event.offset("029C", "UB", "PitotHeat")
```

The way functions work is they get and give values. In our example, the function "gets" the offset from the event, "029B", it looks at that offset and gives it's value in the variable "value".

To make our new function "work", we have to "make a determination". Say to yourself, "If my car is running, I can drive it." Is that a TRUE or FALSE statement? I hope it's False because you're here reading this tutorial and I hope your car isn't running right now. So it's the question of "IF" that makes a determination and allows us to take action.

if...then...else...end

Let's put this into practical use, type this in between the "function" line and the "end" line.

```
if value == 1 then

else

end
```

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](#)

FSUIPC will read the first line `if value == 1` and ask "is it true?" If it IS TRUE, it will then run the lines of code in between the "if...then" line and the "else" line. If it is NOT true, it will run the code between the "else" line and the "end" line.

So now let's tell it what to do "if it's true"... Enter the following line between the "if...then" line and the "else" line...

```
gfd.SetLight(model, unit, id)
```

We should have something like this...

```
function PitotHeat(offset, value)

if value == 1 then
gfd.SetLight(model, unit, id)

else

end

end
event.offset("029C", "UB", "PitotHeat")
```

Before we start getting messy, let's add a little "visual" structure to our code. The "function" is a chunk of code. The "if...then" statement is a chunk of code. The command "gfd.SetLight" is a chunk of code. To help us "see" the structure, we indent one chunk of code within another. A typical indent is four spaces. The first chunk is the "function" so we won't indent that. The second chunk is the "if...then" so we will indent that 4 spaces. The command "gfd.SetLight" will be indented twice (8 spaces) because it's inside the "if...then" chunk, which is inside the "function" chunk. As follows...

```
function PitotHeat(offset, value)

    if value == 1 then
        gfd.SetLight(model, unit, id)

    else

    end

end
event.offset("029C", "UB", "PitotHeat")
```

Now it's much easier to "see" the "command", within the "if...then", within the "function".

But we are missing something very important. The variables model, unit and id have no meaning. Let's add these lines between the "function" line and the "if...then" line.

```
model = GFT8
unit = 0
id = 1
```

So now our it should look like this...

```
function PitotHeat(offset, value)
    model = GFT8
    unit = 0
    id = 1
```

```
        if value == 1 then
            gfd.SetLight(model, unit, id)

        else

        end

    end

end
event.offset("029C", "UB", "PitotHeat")
```

Now we need to tell it what to do if the "if...then" statement is NOT true...
Add the following line in between "else" and "end"

```
gfd.ClearLight(model, unit, id)
```

Now go through it all and remove any blank lines. As shown...

```
function PitotHeat(offset, value)
    model = GFT8
    unit = 0
    id = 1
    if value == 1 then
        gfd.SetLight(model, unit, id)
    else
        gfd.ClearLight(model, unit, id)
    end
end
event.offset("029C", "UB", "PitotHeat")
```

Go ahead and save the file as "PitotHeat.Lua". We have just create an interactive Lua script. It monitors FS's memory and if it finds a change, it will run the "PitotHeat" function. It will then ask if the value of the offset is one, If so, it turns on the light, if not, it turns off the light.

