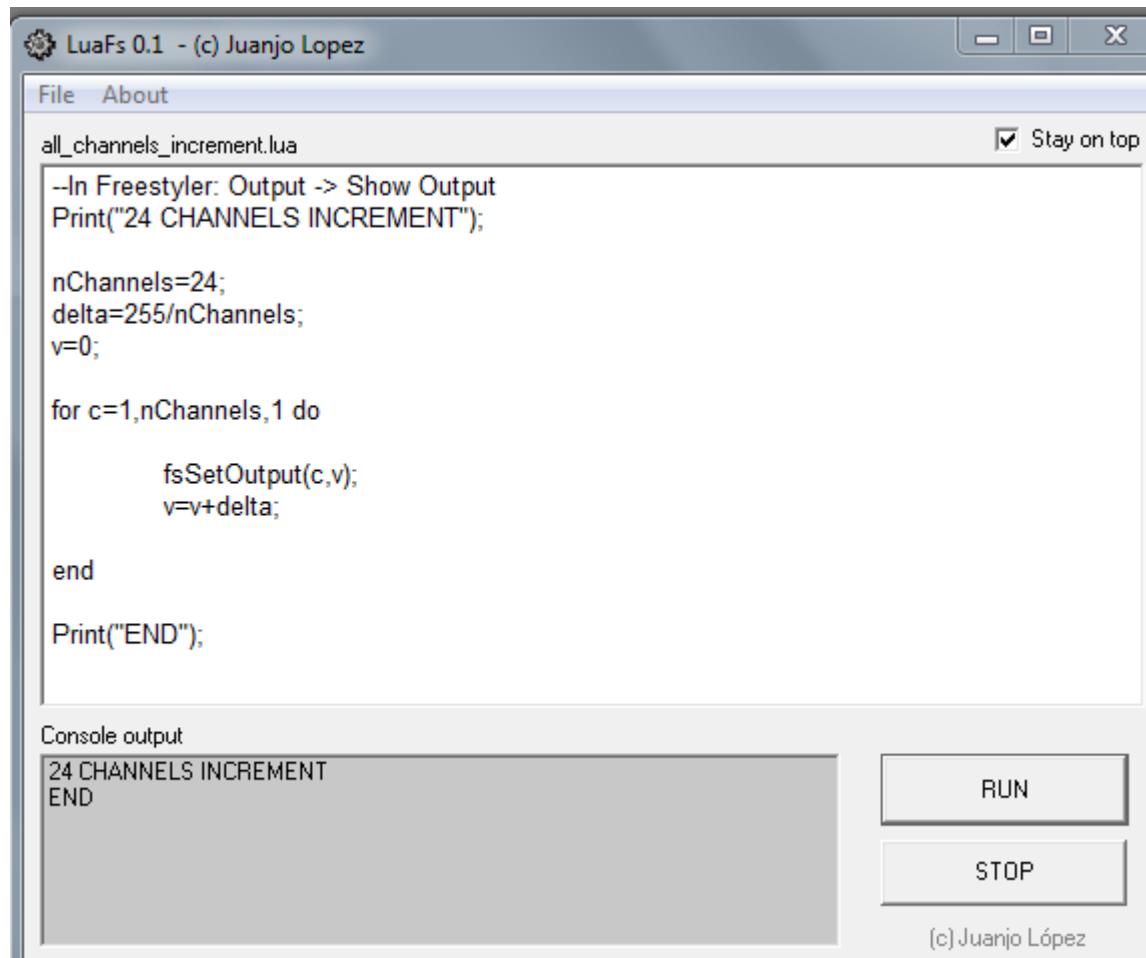


LuaFs – Script Examples

by Juanjo Lopez

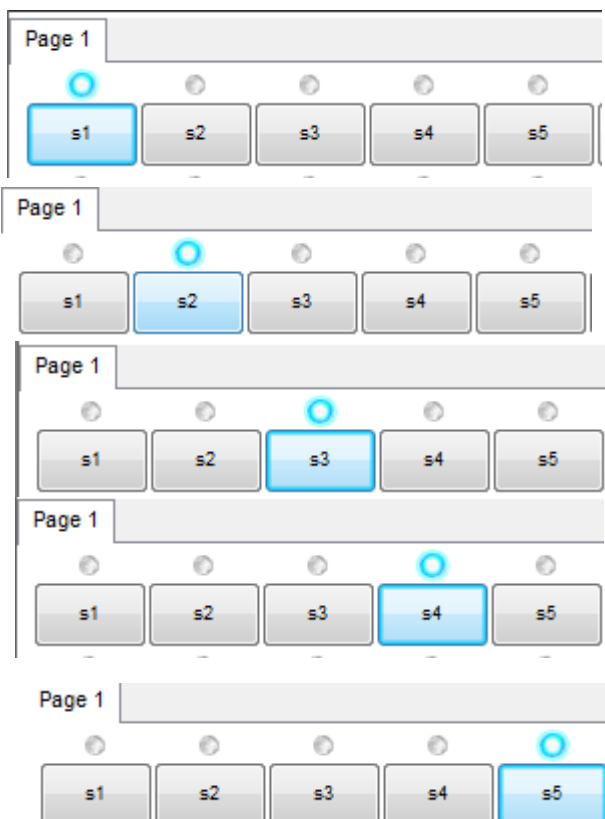
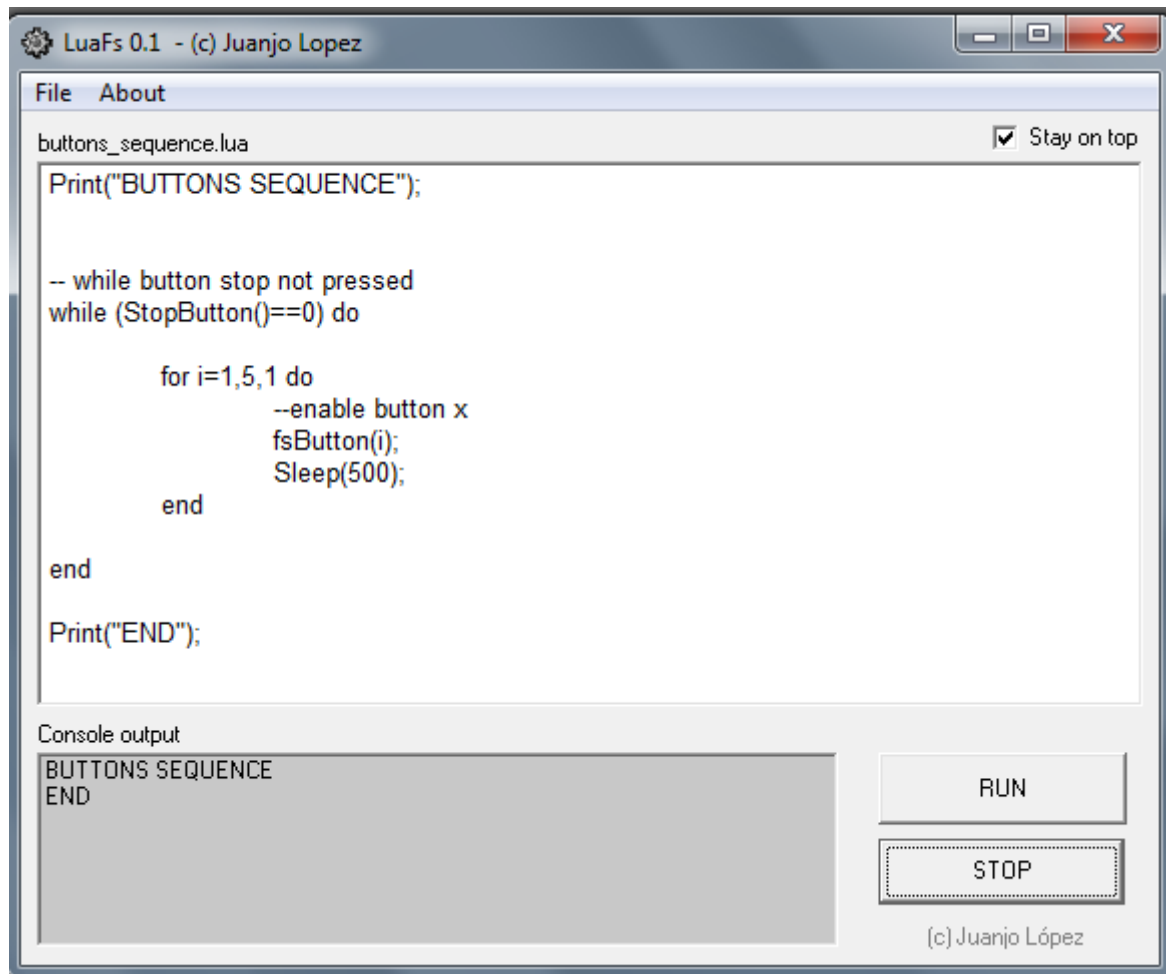
24 Channels Increment



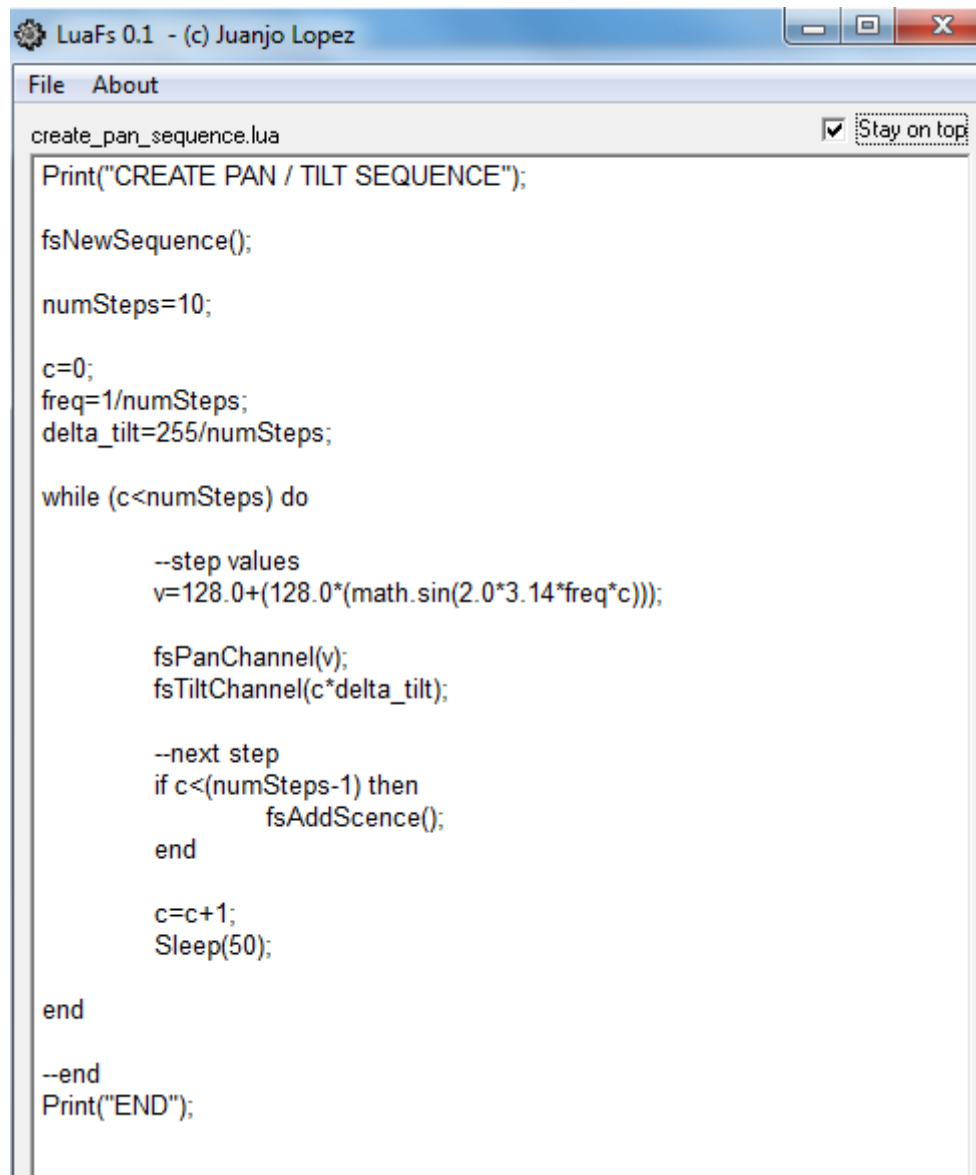
Dimmer output:

[illegible]

Buttons sequence



Create an automatic PAN / TILT sequence (sine)



The screenshot shows a window titled "LuaFs 0.1 - (c) Juanjo Lopez". The window has a menu bar with "File" and "About". Below the menu bar is a tab labeled "create_pan_sequence.lua" with a "Stay on top" checkbox. The main area contains the following Lua code:

```
Print("CREATE PAN / TILT SEQUENCE");

fsNewSequence();

numSteps=10;

c=0;
freq=1/numSteps;
delta_tilt=255/numSteps;

while (c<numSteps) do

    --step values
    v=128.0+(128.0*(math.sin(2.0*3.14*freq*c)));

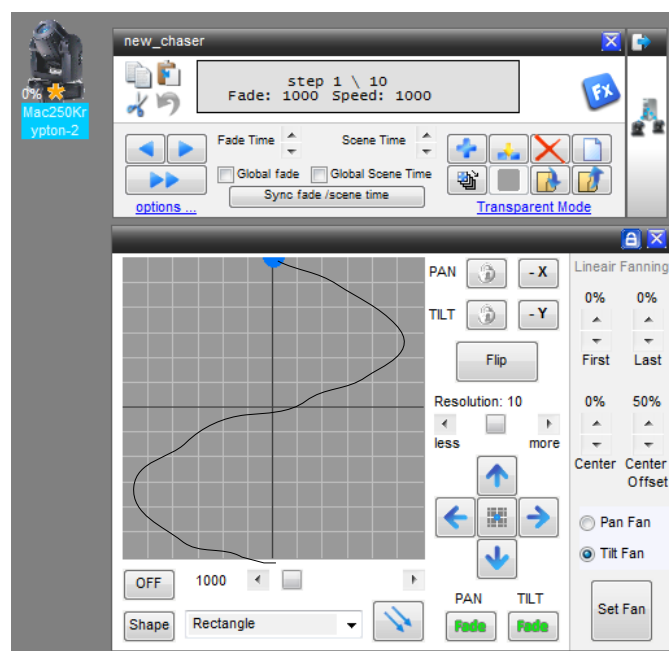
    fsPanChannel(v);
    fsTiltChannel(c*delta_tilt);

    --next step
    if c<(numSteps-1) then
        fsAddScene();
    end


    c=c+1;
    Sleep(50);

end

--end
Print("END");
```



All channels (512) ON/OFF



The screenshot shows a window titled "LuaFs 0.1 - (c) Juanjo Lopez". The window has a menu bar with "File" and "About". Below the menu bar is a toolbar with three icons: a minus sign, a square, and a red X. The main area of the window displays a Lua script in a text editor. The script is named "all_channels_on_off.lua" and has a "Stay on top" checkbox checked. The script content is as follows:

```
--start
Print("START");

while (StopButton()==0) do

    --all output channels to 0
    for c=1,512,1 do
        fsSetOutput(c,0);
    end

    Sleep(500);

    --all output channels to 255
    for c=1,512,1 do
        fsSetOutput(c,255);
    end

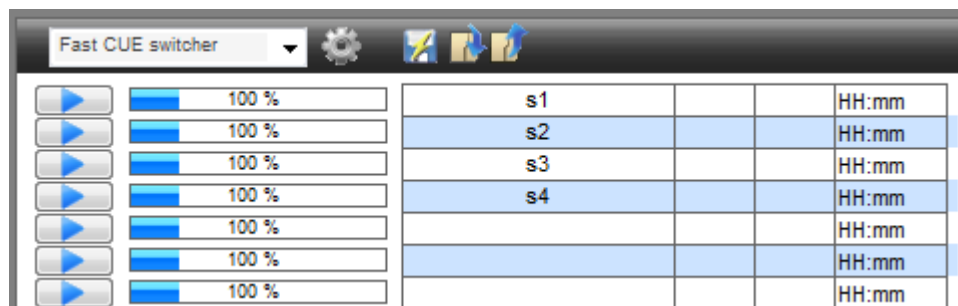
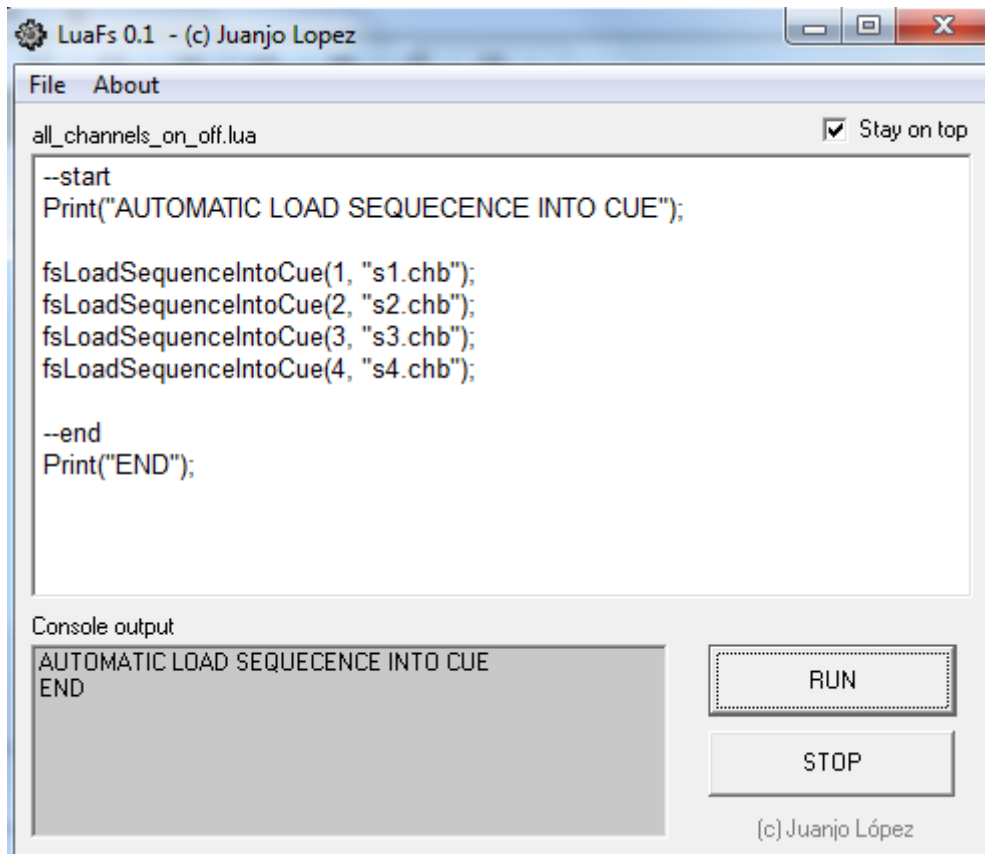
    Sleep(500);

end

--end
Print("END");
```

[illegible][illegible]

Automatic load sequence into cue



Hour programming

hour_programming.lua

```
--start
Print("BLACKOUT HOUR PROGRAMMED...");

--programming hour and minute
h=12;
m=35;

while (StopButton()==0) do

    --get current date time
    d=os.date("*t");

    --check hour and minute
    if (d.hour == h) and (d.min==m) then

        Print("Blackout event!");
        fsToggleBlackout();

        --wait 1 minute to next check
        SleepS(1);

    end

    --wait 1 second
    Sleep(1000);

end

--end
Print("END");
```

Console output

```
BLACKOUT HOUR PROGRAMMED...
END
```

At 12.35



BLACKOUT ON !!!