

Project 2 Home Automation

Background:



First built in the 1960s, smart buildings use a computer to control the temperature and lighting in individual rooms to save energy and to cater to the convenience of the people who use them. Smart houses now allow owners to automate lighting, AC, heating, appliances and entertainment using mobile apps from their phones.

Computers control output ports by checking the content of certain locations in memory. If your computer is old enough it might have a parallel port. The location of the parallel port is **888**. If it is a newer model, you need a GPIO (General Purpose Input-Output). These usually plug into a USB (Universal Serial Bus) port and need to have a driver (a small program that connects the device to the operating system). Once you have the driver installed, find out what resources are used using the device manager. You can run the device manager from the control panel.

Ports will either be 5 volts DC or 0 volts DC depending upon the number you put in the memory location of the device. Use the table to right for controlling connected electronics. If you add the numbers, the computer will turn on more than one device. So, 3 would turn on both port 1 and 2, and 28 (4+8+16) would turn on ports 4, 5, and 6.

1	Port 1 on
2	Port 2 on
4	Port 3 on
8	Port 4 on
16	Port 5 on
32	Port 6 on
64	Port 7 on
128	Port 8 on

Assignment:

Step 1 – Get the DLL (Dynamic Link Library) and save it to where you keep your programs.

Using an API named **inout.dll**, write a Python program to control four lights wired to a TRIAC array.

To get inout.dll go to:

<http://www.highrez.co.uk/downloads/inpout32/>

Scroll down to binaries, and click the first link titled:

[Binaries only - x86 & x64 DLLs and libs.](#) (Mirror)

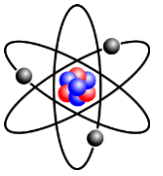
Step 2 – Copy the function calls into your program.

Type or copy the following import and functions to the top of your program:

```
import ctypes

def In(addr) :
    return ctypes.windll.inpout32.Inp32(addr)

def Out(addr, byte) :
    ctypes.windll.inpout32.Out32(addr, byte)
```



Step 3 – Calling the function.

These functions are called the same way all functions are called in Python. Simply type the name and put the right numbers or variables between the parentheses. Here are some examples:

Out (888,1) <- This turns on port 1

Out (888,28) <- This turns on port 4,8, and 16

Out (888,n) <- This turns on port n

Grading and Objectives:

Points will be awarded according to the table below:

	Project	Description	-- Points--		Sgn
			First	Follower	
1	Console control	Turn on lights using numbers typed in from the console.	2	1	
2	Light sequencer	A light sequencer will turn each light on and off in order. When the last light is turned on and off, it starts over with the first light. It runs the lights in continual loop until stopped.	6	3	
3	Button control	Labeled graphic buttons control each light. When clicked, the light will turn on if it was off and off if it was on.	9	5	
4	Timed light	Turn one light on and off at specific times. The times should be programmable	9	3	
5	Radio button control	A modification to button control that turns off all lights except the one you click to save energy.	6	2	
6	Dancing Lights	Play a tune and make the lights come on depending upon the note played. Demonstrate your program with "Mary had a little lamb".	12	4	
7	Your device	Bring in a device from home and run it from your computer. (Only one device per person.)	2	-	
8	Appliance Organ	Program the keys of the keyboard to run small devices (<50 Watts) plugged in to run, or play their note. The devices should only play while the key is pressed and should stop when the key is released.	18	8	
9	Your own	Student defined. (Get your idea approved.)	0-20	1/3	

You need 12 points for an **A**, 9 points for a **B**, 6 points for a **C**. You can go above an **A**, as you do with tests. Student groups split credit. Learn from anyone willing to teach.

Turn this sheet in with signatures for credit.