



USB Passive Infra-Red (PIR) Motion Detector Module.

Need to detect motion from your computer? The USBPIR Passive Infra-Red sensor module easily interfaces to your computer, laptop, or tablet and can be set up to detect motion of humans or animals. Uses of the USBPIR:

- Room Occupancy Sensing
- Kiosk Action Trigger
- Museum Displays
- Daycare Area Alerts
- Motion-Activated Nightlight
- Alarm Systems
- Animated Halloween/Christmas Displays
- Elderly Well-being Sensors
- Pet/Animal Detection



The CGUSPPIR module assembly is a motion detector that identifies as a USBmicro U421 USB Interface. The module detects changes in the field of infra-red that it "sees", and interprets that as motion. The sensor has a range of about 20 feet.

The PIR module performs a self-calibration on power up that lasts up to 40 seconds. During this period it is important to provide the sensor with as little motion or change in environment as possible.

Features

Typically detects a person up to approximately 20 ft away.

Approximately 90 degree field of view

Dual Element Sensor with Low Noise and High Sensitivity

White Light Immunity to Reduce False Triggers

UV Resistant Dome

Simple Programming Example

Read Port A bit 5 in order to read the PIR status. The pin is normally low, motion makes it momentarily high.

```
#!/usr/bin/env python

from tkinter import *
from ctypes import *
import os
import subprocess
import binascii

root = Tk()
```



```
text1="Finding U4xx:"
text2="Number Found:"
text3="Vendor ID:"
text4="Product ID:"
text5="Firmware:"
text6="USBPIR signal:"

data1=""
data2=""
data3=""
data4=""
data5=""
data6=""

USBbuff = create_string_buffer(200)
USBpy = windll.USBm32

data1 = USBpy.USBm_FindDevices()
if data1 == -1:
    data1 = 'Found'
else:
    data1 = '    '
data2 = USBpy.USBm_NumberOfDevices()
data3 = hex(USBpy.USBm_DeviceVID(0))
data4 = USBpy.USBm_DevicePID(0)
data5 = hex(USBpy.USBm_DeviceDID(0))

ret = USBpy.USBm_Copyright(USBbuff)
print("Copyright: " + str(USBbuff.value, 'utf-8'))

ret = USBpy.USBm_About(USBbuff)
print("About: " + str(USBbuff.value, 'utf-8'))
```



```
ret = USBpy.USBm_Version(USBbuff)
print("Version: " + str(USBbuff.value, 'utf-8'))

USBpy.USBm_InitPorts(0)
USBpy.USBm_ReadA(0,USBbuff)
temp = ord(USBbuff[0]) & 32
print(temp)

line1a = Label(root, text=text1, fg='black', font=('OpenSymbol',16))
line2a = Label(root, text=text2, fg='black', font=('OpenSymbol',16))
line3a = Label(root, text=text3, fg='black', font=('OpenSymbol',16))
line4a = Label(root, text=text4, fg='black', font=('OpenSymbol',16))
line5a = Label(root, text=text5, fg='black', font=('OpenSymbol',16))
line6a = Label(root, text=text6, fg='black', font=('OpenSymbol',16))

line1a.grid(row=1,column=0,sticky='W')
line2a.grid(row=2,column=0,sticky='W')
line3a.grid(row=3,column=0,sticky='W')
line4a.grid(row=4,column=0,sticky='W')
line5a.grid(row=5,column=0,sticky='W')
line6a.grid(row=6,column=0,sticky='W')

line1b = Label(root, text=data1, fg='black', font=('OpenSymbol',16))
line2b = Label(root, text=data2, fg='black', font=('OpenSymbol',16))
line3b = Label(root, text=data3, fg='black', font=('OpenSymbol',16))
line4b = Label(root, text=data4, fg='black', font=('OpenSymbol',16))
line5b = Label(root, text=data5, fg='black', font=('OpenSymbol',16))
line6b = Label(root, text=data6, fg='black', font=('OpenSymbol',16))

line1b.grid(row=1,column=1,sticky='E')
```



```
line2b.grid(row=2,column=1,sticky='E')
line3b.grid(row=3,column=1,sticky='E')
line4b.grid(row=4,column=1,sticky='E')
line5b.grid(row=5,column=1,sticky='E')
line6b.grid(row=6,column=1,sticky='E')

def readPIR():
    USBpy.USBm_ReadA(0,USBbuff)
    data6 = ord(USBbuff[0]) & 32
    if data6 == 0:
        line6b['text'] = ''
        line6b['fg'] = 'black'
    if data6 == 32:
        line6b['text'] = 'Alarm!'
        line6b['fg'] = 'red'
    root.after(500,readPIR)

root.after(500,readPIR)
mainloop()
```

For more product information, documentation, and to purchase see this links:

<http://circuitgizmos.com>