

Proyecto Final

AlArma de SEGURIDAD CON SENSOR DE PRESENCIA

(PIR)

Rodney González | Control Digital | 21 de marzo 2019

# Índice

* Introducción………………………………………………………………………………2
* Objetivo………………………………………………………………………………………3
* Materiales……………………………………………………………………………………4
* Marco Teórico……………………………………………………………………………5-8
* Programaciones utilizadas para el control de dispositivos ………9-13
* Diagrama de conexiones (Raspberry Pi 3) ………………………………14-16
* Conexión del proyecto……………………………………………………...……17
* Conclusión y ……………………………………………………………………………18

Introducción

Las Alarmas y sistemas de seguridad tiene como objetivo proteger los inmuebles, los bienes y Asus habitantes.

Los sistemas tecnológicos dirigidos a proteger nuestros inmuebles, bienes y habitantes contra amenazas y peligros se denominan sistemas de alarmas y seguridad. Los sistemas de seguridad y alarmas en el contexto del edificio y hogar se pueden clasificar en varias áreas:

Alarmas de Intrusión (Movimiento, presencia).

Video Vigilancia (IP/Analógica)

Control de accesos

Alarmas Técnicas Incendio, Humo, etc.)

Alarmas personales (SOS)

Los sistemas de cámaras de seguridad tienen como finalidad aumentar el nivel de confianza y bienestar del usuario. Un sistema de este tipo por sí sólo no aumenta la seguridad pero tiene un gran efecto disuasorio y permite al usuario en cualquier momento visualizar los principales locales de su edificio sin tener que desplazarse.

Objetivo

El objetivo del proyecto es realizar un sistema de seguridad con sensor de presencia (PIR) transmite en vivo una cámara la cual con motivos de seguridad nos muestra video de lo que pasa en la hogar o comercios. Que para no cuando no haya nadie en el hogar el sensor detecte la presencia y envié por correo electrónicos un aviso Alerta Intruso y a la vez me adjunte la imagen del mismo; y guardar la captura de la imagen en mi ordenador.

Lista de Materiales.

● Raspberry Pi Modelo B+

● Cable de alimentación

● Cable HDMI

● Cámara

● Teclado y Ratón

● Pantalla con entrada HDMI

● Adaptador Wifi

● Jumpers macho-hembra

● Alambre para conexiones

● Laptop

Marco Teórico

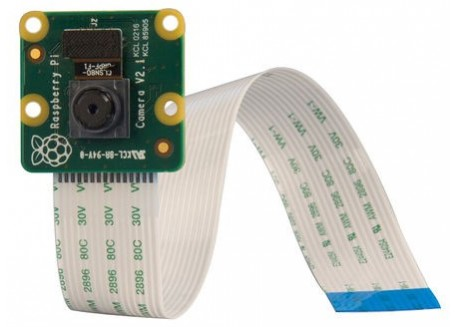
Conexión del módulo de cámara a Pi

Un cable plano de 15 cm fijado a las ranuras del módulo directamente en el puerto de interfaz serie de su cámara Pi (CSI). Una vez conectado, puede acceder a la placa de cámara a través de la capa de abstracción multimedia (MMAL) o el vídeo para API de Linux (V4L). También hay bibliotecas como Picamera Python y muchas otras que puede encontrar en línea.

Aplicaciones

El módulo de cámara Raspberry Pi es ideal para vídeos HD y fotografía. También puede probar la función de cámara lenta y time-lapse. Otras aplicaciones incluyen botánica, observación de vida salvaje y seguridad doméstica.

Características

Imágenes de alta calidad

Alta capacidad de datos

Enfoque fijo de 8 megapíxeles

Compatible con 1080p, 720p60 y VGA90

Sensor de imagen CMOS Sony IMX219PQ

Cable plano de 15 contactos

Cámara de red

Una cámara de red (cámara IP) es una cámara que emite imágenes directamente a la red (Intranet o Internet) sin necesidad de un ordenador, una cámara de red incorpora su propio miniordenador, lo que le permite emitir video por sí misma.

Webserver

Es un equipo capaz de generar información en formato de páginas HTML, con la enorme ventaja de poderlas visualizar con cualquier navegador estándar de mercado (IE 5, Netscape, Mozilla...)

Si eres instalador encontrarás páginas que contienen información de programación y puesta en marcha de la instalación, protegidas por clave de instalador.

Como usuario, tendrá acceso a otro tipo de páginas, de control e informativas del estado de su vivienda.

Y en caso de necesidad, el servidor le notificará cualquier eventualidad a sus direcciones de e-mail, o a su teléfono WAP con tecnología GPRS.

Raspberry PI:

Es un computador de placa reducida, computador de placa única o computador de placa simple (SBC) de bajo costo desarrollado en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas.

Aunque no se indica expresamente si es hardware libre o con derechos de marca, en su web oficial explican que disponen de contratos de distribución y venta con dos empresas, pero al mismo tiempo cualquiera puede convertirse en revendedor o redistribuidor de las tarjetas RaspBerry Pi, por lo que da a entender que es un producto con propiedad registrada, manteniendo el control de la plataforma, pero permitiendo su uso libre tanto a nivel educativo como particular.

En cambio, el software sí es open source, siendo su sistema operativo oficial una versión adaptada de Debian, denominada Raspbian, aunque permite usar otros sistemas operativos, incluido una versión de Windows 10. En todas sus versiones incluye un procesador Broadcom, una memoria RAM, una GPU, puertos USB, HDMI, Ethernet (El primer modelo no lo tenía), 40 pines GPIO y un conector para cámara. Ninguna de sus ediciones incluye memoria, siendo esta en su primera versión una tarjeta SD y en ediciones posteriores una tarjeta MicroSD.

Sensor de Presencia (PIR)

Principios de funcionamiento:

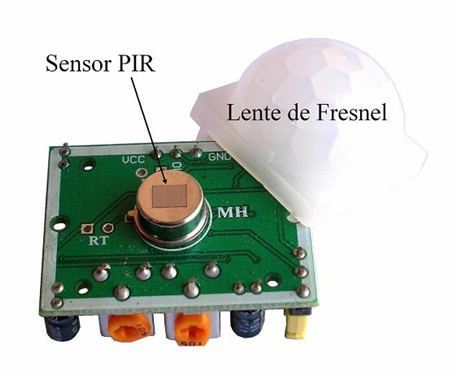
La radiación infrarroja:

Todos los seres vivos e incluso los objetos, emiten radiación electromagnética infrarroja, debido a la temperatura a la que se encuentran. A mayor temperatura, la radiación aumenta. Esta característica ha dado lugar al diseño de sensores de infrarrojo pasivos, en una longitud de onda alrededor de los 9.4 micrones, los cuales permiten la detección de movimiento, típicamente de seres humanos ó animales.

Estos sensores son conocidos como PIR, y toman su nombre de ‘Pyroelectric Infrared’ ó ‘Passive Infrared’.

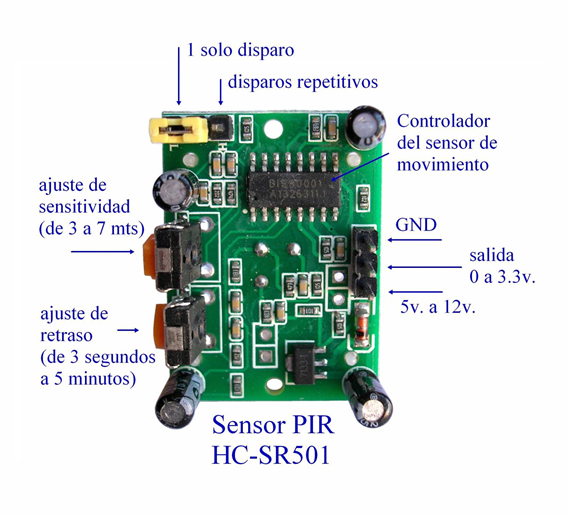
El lente de Fresnel:

El lente de Fresnel es un encapsulado semiesférico hecho de polietileno de alta densidad cuyo objetivo es permitir el paso de la radiación infrarroja en el rango de los 8 y 14 micrones. El lente detecta radiación en un ángulo con apertura de 110° y, adicionalmente, concentra la energía en la superficie de detección del sensor PIR, permitiendo una mayor sensibilidad del dispositivo.



Descripción del HC-SR501

El módulo PIR modelo HC-SR501 es de bajo costo, pequeño, e incorpora la tecnología más reciente en sensores de movimiento. El sensor utiliza 2 potenciómetros y un jumper que permiten modificar sus parámetros y adaptarlo a las necesidades de la aplicación: sensitividad de detección, tiempo de activación, y respuesta ante detecciones repetitivas.

Sus especificaciones técnicas son:

-Usa el PIR LHI778 y el controlador BISS0001

-Voltaje de alimentación: de 5 a 12 VDC

-Consumo promedio: <1 miliampere

-Rango de distancia de 3 a 7 metros ajustable.

-Angulo de detección: cono de 110°

-Ajustes: 2 potenciómetros para ajuste de rango de detección y tiempo de alarma activa.

-Jumper para configurar la salida de alarma en modo mono-disparo ó disparo repetitivo (‘rettrigerable’)

-Salida de alarma de movimiento con ajuste de tiempo entre 3 segundos a 5 minutos.

-Salida de alarma activa Vo con nivel alto de 3.3 volts y 5 ma source, lista para conexión de un led, ó un transistor y relevador.

-Tiempo de inicialización: después de alimentar el módulo HC-SR05, debe transcurrir 1 minuto antes de que inicie su operación normal. Durante ese tiempo, es posible que el módulo active 2 ó 3 veces su salida.

-Tiempo de salida inactiva: cada vez que la salida pase de activa a inactiva, permanecerá en ese estado los siguientes 3 segundos. Cualquier evento que ocurra durante ese lapso es ignorado.

-Temperatura de operación: -15° a +70° C.

-Dimensiones: 3.2 x 2.4 x 1.8 cms.

Programaciones Utilizadas

Instalación de Raspbian

Si está utilizando la distro Raspbian, es probable que tenga la cámara instalada de forma predeterminada. Puedes averiguarlo simplemente iniciando Python e intentando importar picamera:

python -c "import picamera"

python3 -c "import picamera"

Si no recibe ningún error, ¡ya tiene instalado Picamera! Simplemente continúe con Primeros pasos . Si no tiene instalado Picamera verá algo como lo siguiente:

python -c "import picamera"

Traceback (most recent call last):

File "<string>", line 1, in <module>

ImportError: No module named picamera

python3 -c "import picamera"

Traceback (most recent call last):

File "<string>", line 1, in <module>

ImportError: No module named 'picamera'

Para instalar Picamera en Raspbian, es mejor usar el administrador de paquetes del sistema: apt. Esto asegurará que la cámara fotográfica sea fácil de mantener actualizada y fácil de eliminar si así lo desea. También hará que la cámara esté disponible para todos los usuarios del sistema. Para instalar picamera usando apt simplemente ejecuta:

sudo apt-get update

sudo apt-get install python-picamera python3-picamera

Para actualizar su instalación cuando se realicen nuevas versiones, simplemente puede utilizar el procedimiento normal de actualización de apt:

sudo apt-get update

$sudo apt-get upgrade

Si alguna vez necesitas quitar tu instalación:

sudo apt-get remove python-picamera python3-picamera

1.2. Instalación de distribución alternativa

En otras distribuciones que no sean Raspbian, es probablemente más sencillo instalar todo el sistema con la herramienta pip de Python:

sudo pip install picamera

Si desea usar las clases en el módulo picamera.array , especifique la opción “array” que se colocará en forma de dependencia como:

sudo pip install "picamera[array]"

Advertencia

Tenga en cuenta que las versiones anteriores de pip intentarán crear números desde la fuente. Esto llevará mucho tiempo en un Pi (varias horas en modelos más lentos). Las versiones modernas de pip descargarán e instalarán una "rueda" numpy preconstruida, que es mucho más rápida.

Para actualizar su instalación cuando se hacen nuevas versiones:

sudo pip install -U picamera

Para quitar instalación:

sudo pip uninstall picamera

Advertencia

Anteriormente, estos documentos han sugerido el uso de la utilidad rpi-update para actualizar el firmware del Pi; esto ahora está desanimado. Si anteriormente ha utilizado la utilidad rpi-update para actualizar su firmware, puede volver a usar apt para administrarlo con los siguientes comandos:

sudo apt-get update

sudo apt-get install --reinstall libraspberrypi0 libraspberrypi-{bin,dev,doc} z

> raspberrypi-bootloader

sudo rm /boot/. firmware\_revisio

Capturando secuencias de timelapse

La forma más sencilla de capturar secuencias largas de lapso de tiempo es con el método capture\_continuous (). Con este método, la cámara captura imágenes continuamente hasta que le dices que se detenga. Las imágenes reciben automáticamente nombres únicos y puede controlar fácilmente el retraso entre capturas. El siguiente ejemplo muestra cómo capturar imágenes con un retraso de 5 minutos entre cada toma:

from time import sleep

from picamera import PiCamera

camera = PiCamera ()

camera . start\_preview ()

sleep ( 2 )

for filename in camera . capture\_continuous ( 'img {counter:03d} .jpg' ):

print ( 'Captured %s ' % filename )

sleep ( 300 ) # wait 5 minutes

Sin embargo, es posible que desee capturar imágenes en un momento determinado, por ejemplo, al comienzo de cada hora. Esto simplemente requiere un refinamiento del retraso en el bucle (el módulo datetime es un poco más fácil de usar para calcular fechas y horas; este ejemplo también muestra la plantilla de timestamp en los nombres de archivos capturados):

from time import sleep

from picamera import PiCamera

from datetime import datetime , timedelta

def wait ():

# Calculate the delay to the start of the next hour

next\_hour = ( datetime . now () + timedelta ( hour = 1 )) . replace (

minute = 0 , second = 0 , microsecond = 0 )

delay = ( next\_hour - datetime . now ()) . seconds

sleep ( delay )

camera = PiCamera ()

camera . start\_preview ()

wait ()

for filename in camera . capture\_continuous ( 'img{timestamp:%Y-%m- %d -%H-%M}.jpg' ):

print ( 'Captured %s ' % filename )

wait ()

Uso de smtplib en Python

Minireceta : Ejemplo de conexión a servidor SMTP: muestra cómo enviar correos que incluyan, además, direcciones CC y BCC.

from smtplib import SMTP

from email.mime.image import MIMEImage

from email.mime.text import MIMEText

from email.mime.multipart import MIMEMultipart

to = 'direccion@destino.com'

cc = 'direccion\_cc@destino.com'

bcc = 'direccion\_bcc@destino.com'

msg = MIMEMultipart()

msg['To'] = to

msg['Cc'] = cc

msg['From'] = 'direccion@origen.com'

msg['Subject'] = 'Asunto de prueba'

text = MIMEText('Texto del mensaje')

text.add\_header("Content-Disposition", "inline")

msg.attach(text)

s = SMTP('smtp.servidor.com')

s.sendmail(msg['From'], [to, cc, bcc], msg.as\_string())

s.quit()

Trabajo Confeccionado

from picamera import PiCamera

from time import sleep

import smtplib

gmail\_user = 'rodneyenvio@gmail.com'

gmail\_password = 'rodn1991'

import time

from datetime import datetime

from email.mime.image import MIMEImage

from email.mime.text import MIMEText

from email.mime.multipart import MIMEMultipart

import RPi.GPIO as GPIO

import time

to = 'gonzalezrodney994@gmail.com'

me = 'rodneyenvio@gmail.com'

Subject='Alerta Intruso'

GPIO.setmode(GPIO.BCM)

P=PiCamera()

P.resolution= (1024,768)

P.start\_preview()

GPIO.setup(4, GPIO.IN)

while True:

if GPIO.input(4):

print("capturando...")

#camera warm-up time

time.sleep(2)

P.capture('Intruso.jpg')

time.sleep(3)

subject='Alerta Intruso'

msg = MIMEMultipart()

msg['Subject'] = 'Alerta Intruso'

msg['From'] = 'gonzalezrodney994@gmail.com'

msg['To'] = 'gonzalezrodney994@gmail.com'

msg['Cc'] = 'rodneyenvio@gmail.com'

text = MIMEText('De Confirmarce el intruso en casa llamar inmediatamente al 104. Urgente!!!!!!!!')

text.add\_header("De Confirmarce el intruso en casa llamar inmediatamente al 104. Urgente!!!!!!!!", "inline")

msg.attach(text)

fp = open('Intruso.jpg','rb')

img = MIMEImage(fp.read())

fp.close()

msg.attach(img)

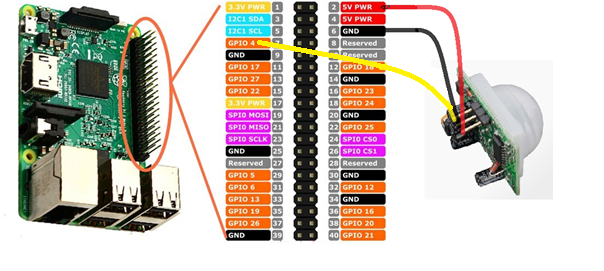
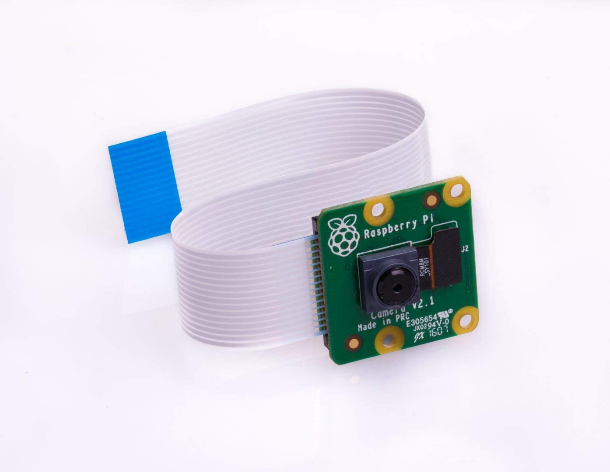
server = smtplib.SMTP\_SSL('smtp.gmail.com',465)

server.login(gmail\_user, gmail\_password)

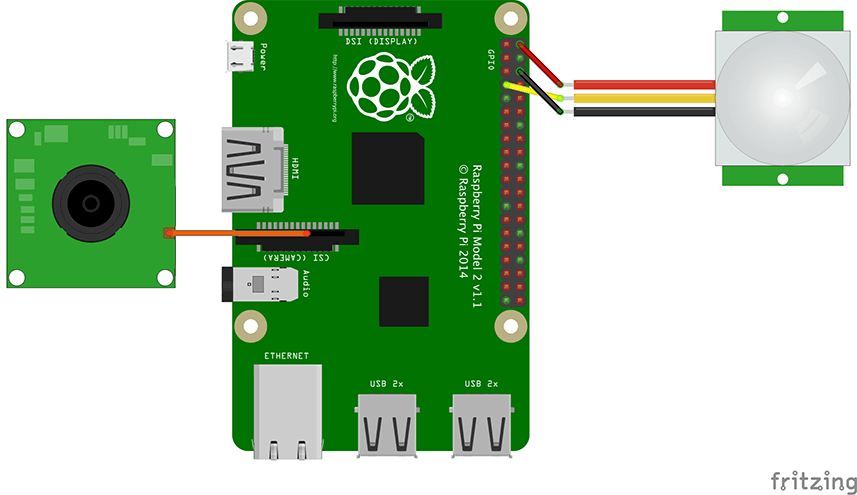
server.send\_message(msg)

server.quit()

Armado del Dispositivo



|  |  |
| --- | --- |
| Dispositivo | Puerto GPIO |
| Alimentación | 2 |
| Ground | 6 |
| Sensor | 7 |



Conclusiones.

Como conclusión del proyecto aplique un sistema de seguridad de alarma con sensor de presencia, lo que es fundamental de conocer en la realización de este proyecto es cómo es que nosotros podemos utilizar internet no solo en el uso de nuestros smartphones en redes sociales si no que pudimos implementar nuestro propio sistema inteligente de control por medio del uso del microcontrolador rasberry pi.

Me sirve como seguridad en la casa que, de hecho, además de tener un sistema de alarma que cada vez que el sensor detecta la presencia de alguien manda un mensaje a nuestro correo informando que alguien está cerca, y a la vez adjunta la imagen de la presencia en ese momento.

En conclusión, final el rasberry pi3 se puede usar entre muchas otras cosas más que talvez aún no sabemos, pero con este proyecto fue una clave para mi aprendizaje a lo largo de mi carrera.