



CCADET  
CENTRO DE CIENCIAS APLICADAS  
Y DESARROLLO TECNOLÓGICO

**SOMI** CONGRESO  
DE  
INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

## ALARMA VIA E-MAIL CON RASPBERRY PI Y PIFACE

**M.C. Gloria Mónica Martínez Aguilar, M.C. Vanessa Maribel Morales Ibarra, Ing. Eduardo Salazar Valle**

gmartinez@utt.edu.mx; vmorales@utt.edu.mx, esalazar@utt.edu.mx

Universidad Tecnológica de Torreón  
Carretera Torreón-Matamoros Km. 10 S/N Ejido el águila, Torreón, Coahuila, México

### RESUMEN

La tarjeta Raspberry Pi es un minicomputador él cuenta con los elementos básicos de hardware, debido a su estructura modular y open source, con esto permite su uso en diferentes sistemas operativos ampliando su capacidad de manejo para el programador. PiFace Digital es una placa de expansión I/O que se conecta al GPIO de la Raspberry Pi con esta se puede detectar el estado de un interruptor, en este trabajo se usó un sensor magnético para una puerta, en donde una vez detectado envía un correo mediante Mutt, este correo realizara la función de aviso de alarma. El sistema de alarma se puede aplicar en diferentes áreas como los son el residencial, empresarial o de cualquier bien a proteger. El uso de la tarjeta PiFace permite el manejo de sensores y actuadores de manera simultánea, dándole una arquitectura abierta y adaptabilidad al sistema según se requiera.

**PALABRAS CLAVE:** Alarma, Raspberry Pi, Piface, Mutt, Python.

### 1 INTRODUCCIÓN

Una alarma es un dispositivo electrónico o elemento de seguridad pasiva el cual funciona como aviso o advertencia sobre la proximidad de un peligro. Existen múltiples sistemas de alarmas los cuales podrán ser implementados en casas, automóviles, negocios o cualquier bien que desee ser resguardado.

#### 1.1 Raspberry Pi

Raspberry Pi (Figura 1) Raspberry Pi es un ordenador de placa reducida (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. El diseño incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 700 MHz (el firmware incluye unos modos "Turbo" para que el usuario pueda hacerle overclock de hasta 1 GHz sin perder la garantía), un procesador gráfico (GPU) VideoCore IV, y 512 MB de memoria RAM (Figura 1).

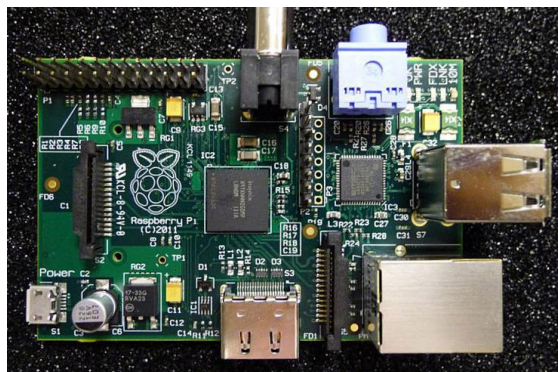


Figura 1. Raspberry Pi.

La fundación da soporte para las descargas de las distribuciones para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y sPidora (derivado de Fedora); y promueve principalmente el aprendizaje del lenguaje de programación Python, y otros lenguajes como Tiny BASIC, 12 C y Perl [1].

## 1.2 Piface Digital

Este tipo de tarjeta expansiva ayuda a controlar los periféricos que se deseen a través de Raspberry Pi, se añade directamente al puerto GPIO en el Raspberry Pi y se obtiene un número de entradas y salidas donde se pueden detectar estados de conmutación y controlar actuadores según se requiera [2]. La tarjeta se puede programar de la forma que se necesite. Todos los conectores de entrada y salida son terminales de tornillo, por lo que es muy fácil de cablear el Pi a todo tipo de sistemas externos. Las características de la Piface Digital son:

- ✚ Se conecta directamente a la toma de Raspberry Pi GPIO
- ✚ Se fija directamente sobre el Raspberry Pi y dentro del firmware del Raspberry Pi
- ✚ 2 relés de conmutación
- ✚ 4 Interruptores táctiles
- ✚ 8 entradas digitales
- ✚ 8 salidas open-Collector
- ✚ 8 indicadores LED
- ✚ Fácil de programar en Python, Scratch y C
- ✚ Emulador gráfico y simulador
- ✚ Los relés se pueden utilizar para conmutar tensiones de hasta 250 V (máx.) o las corrientes hasta 10A (Max)

## 1.3 Mutt

Mutt es un cliente de correo renombrado por sus poderosas características que funciona en consola y ofrece un amplio abanico de posibilidades. Mutt, a pesar de solo contar con un poco más de una década desde su lanzamiento, sigue siendo el cliente de correo favorito de un gran número de usuarios expertos. La potencia de Mutt está en su amplia configurabilidad.

Mutt se enfoca en ser un Mail User Agent, y fue originalmente escrito solo para ver correo. Debido a esto, la recuperación de correo, características de envío y filtrado implementadas posteriormente

son básicas comparadas con otras aplicaciones. La mayoría de las instalaciones de Mutt dependen de programas externos para estas tareas.

Sin embargo, el paquete Mutt de Arch ha sido compilado con soporte para IMAP, POP3, y SMTP [3].

## 2 ARMADO Y CONFIGURACION DE LA ALARMA

El armado y configuración de la alarma se realizó en varias secciones las cuales se presentan a continuación.

### 2.1 Configuración Raspberry Pi

La primera parte a realizar es la configuración del Raspberry Pi y el conexionado del Piface, los pasos a seguir para ello son los siguientes:

1.- En primer lugar hay que descargarse la imagen del Sistema operativo que se va a utilizar. En este caso la distribución Raspbian, basada en Debian. En el siguiente enlace se pueden encontrar los S.O. que pueden cargarse a la Raspberry Pi: <http://www.raspberrypi.org/downloads>

2.- Con la ayuda del programa Win32 Disk Imager (Figura 2) se carga el S.O. a una memoria SD de mínimo 4 GB, la descarga de S.O. debe de descomprimirse para poder grabarse en la SD. Para proceder a la escritura de la imagen de Raspbian Wheezy a nuestra SD (en este caso la unidad I:\) basta con pulsar sobre el botón "Write" y esperar a que nos diga que ha finalizado el proceso.

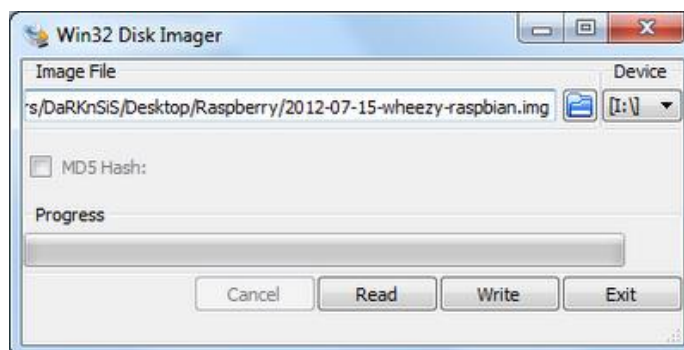


Figura 2. Carga de S.O. a SD mediante Win32 Disk Imager.

3.- Una vez cargado el sistema a la SD esta se introduce a la Raspberry Pi y se conectan los periféricos de la siguiente manera: Port HUB (USB), monitor (HDMI) y al port HUB mouse, teclado, Wifi (Figura 3).





Figura 3. Conexión de periféricos a Raspberry PI.

4.- Se alimenta la Raspberry Pi con lo cual se pone en marcha el S.O. y aparece un menú en el cual se puede configurar las propiedades del S.O. (Figura 4) en este caso solo se cambió el Password, el cual por default es Username: pi, Password: Raspberry.

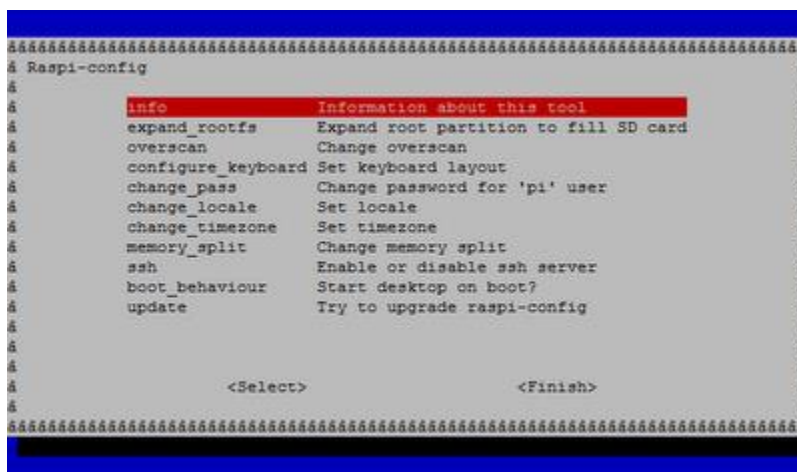


Figura 4. Menú de configuración de Raspbian.

5.- Se accede al sistema con el username y Password, el S.O. empieza en modo consola y para pasar a modo grafico (Figura 5) se tecldea: **startx**

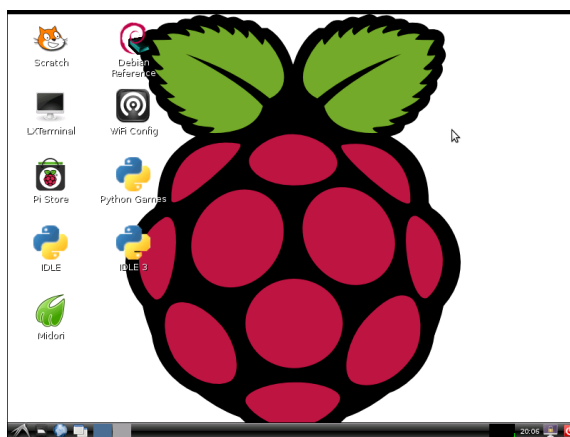


Figura 5. Modo grafico Raspbian.

6.- Se conecta a una red inalámbrica mediante Wifi Config, donde se presiona scan y se agrega la red a la cual se conectara y listo se desplegara un aviso de conexión completa (Figura 6).

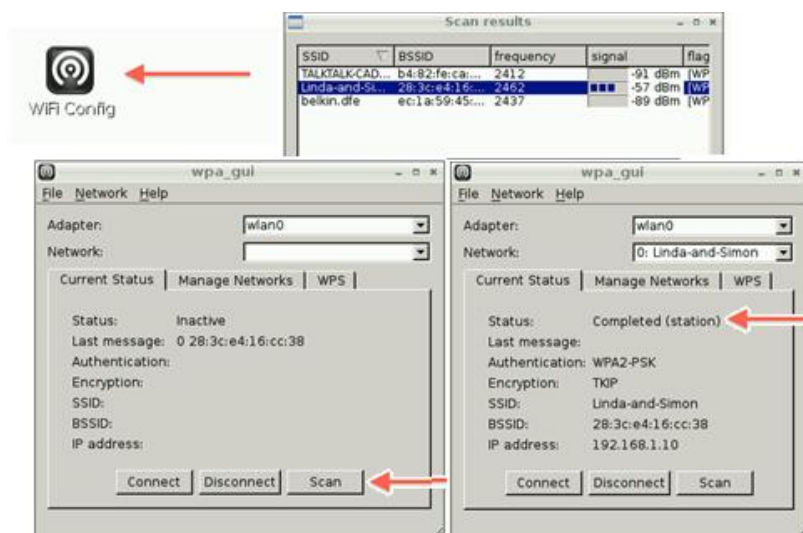


Figura 6. Configuración de Wlan0.

7.- Ahora se procede a la configuración e instalación de los paquetes de la Piface Digital, para eso se abre una consola y se escriben los siguientes comandos:

***sudo apt-get update***

***sudo apt-get upgrade***

Estos comandos actualizaran e instalaran los paquetes para raspbian

***sudo apt-get install raspi-config***

### ***sudo raspi-config***

Estos comandos instalarán el nuevo kit de configuración de Raspbian y lo abrirán y dentro de él se elige lo siguiente:

#### ***Option 8 Advanced Options*** ***A5 SPI***

***Yes***

#### ***Finish***

Con esto se habilita el puerto SPI para la conexión de la tarjeta Piface Digital.

## **2.2 Conexión de Piface a Raspberry Pi**

Con la configuración de Raspberry Pi terminada solo falta instalar el emulador de Piface y el controlador con los siguientes comandos:

***sudo apt-get install python3-pifacedigital-emulator***  
***sudo halt***

Se apaga el sistema para la instalación de la tarjeta (Figura 7)

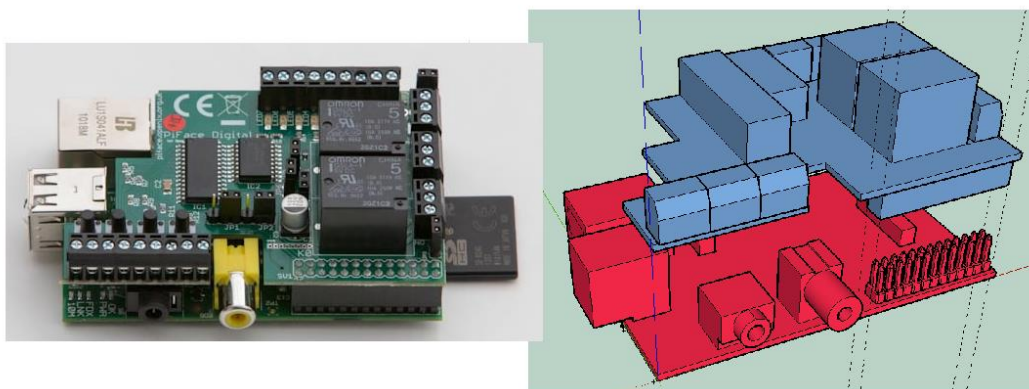


Figura 7. Conexión de Raspberry Pi con Piface Digital.

Se reinicia el sistema y en la consola se escriben los siguientes comandos para probar el funcionamiento del módulo

***python3 /usr/share/doc/python3-pifacedigitalio/examples/blink.py***

Este ejemplo hace parpadear una salida digital de Piface y para detenerlo solo se tecléa **Ctrl c**, otra manera de checar el módulo es mediante el emulador el cual se puede llamar desde la consola con el comando **piface-emulator** seleccionando el icono en el escritorio, lo cual nos abrirá el emulador desde el cual podremos activar salidas y checar el estado de las entradas (Figura 8).



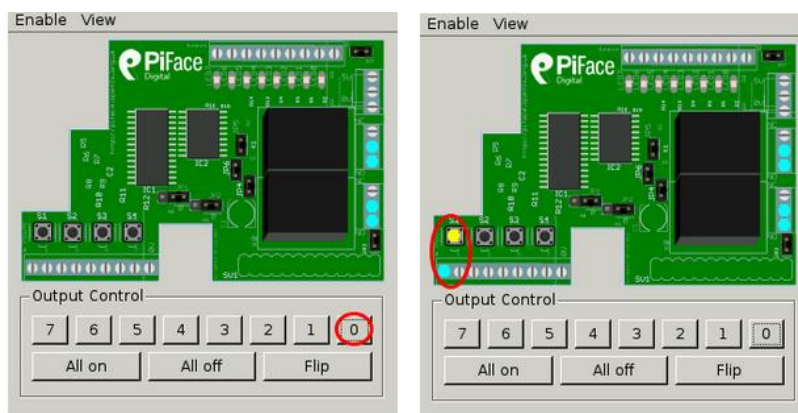


Figura 8. Piface Emulator.

## 2.3 Conexionado de Piface a sensores de alarma

Una vez instalado correctamente el modulo se procede a conectar a la entrada 0 del Piface digital el sensor que activara la alarma (cabe destacar que se pueden conectar varios y de diferentes índoles) como se muestra en la figura 9.

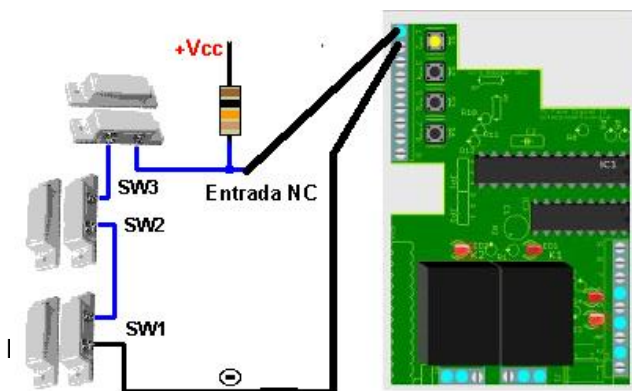


Figura 9. Conexión de Varios.

## 2.4 Configuración de Mutt

Con la conexión realizada se procede a la instalación configuración de Mutt con los siguientes pasos:

1.- Se instala MUTT son los comandos:

```
apt-get update
apt-get install mutt
```

2.- Ahora con el paquete instalado se configurar para enviar desde el root. Para ello se crea el siguiente fichero:



```
sudo nano /root/.muttrc
```

3.- Lo editamos con la siguiente información:

```
set from = "USER@gmail.com"  
set realname = "Frikis House"  
set imap_user = "USER@gmail.com"  
set imap_pass = "CLAVE"  
set folder = "imaps://imap.gmail.com:993"  
set spoolfile = "+INBOX"  
set postponed = "+[Gmail]/Drafts"  
set header_cache = ~/.mutt/cache/headers  
set message_cachedir = ~/.mutt/cache/bodies  
set certificate_file = ~/.mutt/certificates  
set smtp_url = "smtp://USER@smtp.gmail.com:587/"  
set smtp_pass = "CLAVE"
```

4.- Se guarda con **^O** y se crea el directorio:

```
mkdir -p /root/.mutt/cache
```

## 2.5 Programación Python de alarma

Lo siguiente es realizar un programa en Python para mandar la alarma, los comandos para crear, abrir y correr un archivo Python son:

```
touch yourProgramName.py  
leafpad yourProgramName.py  
python3 yourProgramName.py
```

Para acceder a manipular el módulo Piface por medio de Python los se importa la el modulo mediante el comando:

```
import pifacedigitalio as p
```

Antes de usarlo, el modulo debe de inicializarse mediante el comando

```
p.init()
```

Hay tres funciones principales para el control del módulo los cuales son

```
digital_read(pin_number)
```

Lo que retorna 1 o 0 dependiendo del estado de la entrada numerada por el pin\_number

```
digital_write(pin_number, state)
```

Pone la salida numerada por el pin\_number a 0 o 1. El estado 1 prende LED de la salida y habilita el colector para dejar pasar la corriente.

```
digital_write_pullup(pin_number, state)
```





Pone el Pullup de 10k en la entrada numerada por el pin\_number para un estado de 0 o 1. El estado 1 es pullup habilitado.

Con estas funciones se programa las entradas y salidas según lo que se desee controlar. El programa en Python queda de la siguiente manera:

```
from time import sleep  
import os  
import pifacedigitalio as p  
p.init ()  
DELAY = 1.0 # seconds  
while True:  
    x = p.digital_read(0)  
    if x == 1:  
        os.system("echo \'Alarma\'|mutt -s \'Alarma activada\' User@gmail.com")  
    sleep (DELAY)
```

Para finalizar solo se corre el programa en modo consola como superusuario y se le desconectan los periféricos al Raspberry Pi que no va a utilizar y se posiciona en el lugar donde estará monitoreando.

### 3 RESULTADOS

El sistema funciona correctamente, al activarse el sensor se recibe el correo de aviso en aproximadamente dos segundos, cabe destacar que se le pueden adicionar más sensores según se requiera además de actuadores como sirenas, cámaras web, etc. para realizar un sistema tan complejo como se requiera.

### 4 CONCLUSIONES

El sistema mostrado en este artículo es de fácil implementación y de un costo asequible, además que está realizado con tecnología open hardware y open source lo cual le da la ventaja de una arquitectura abierta y adaptabilidad según el área de aplicación. Cabe destacar que la modularidad de la Raspberry Pi permite sensores más complejos como un GPS con lo cual se puede desarrollar una alarma más completa.

### 5 AGRADECIMIENTOS

Se agradece a la Universidad Tecnológica de Torreón y al Cuerpo Académico UTTOR-05 por su apoyo para la realización de este proyecto.



**CCADET**  
CENTRO DE CIENCIAS APLICADAS  
Y DESARROLLO TECNOLÓGICO

# SOMI

CONGRESO  
DE  
INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

## REFERENCIAS

- [1] Mike McGrath “Raspberry Pi in Easy Steps”, Paperback, 2013.
- [2] G. Martínez, V, Morales, E. Salazar, D. Rosales , “Control domótico con Raspberry Pi y Piface Digital”, Academia Journals Mazatlán, 2014.
- [3] Víctor Gregorio. Linux Journal. “Power up your e-mail with mutt ”. Volume 2008 pp 173. Septiembre de 2008.