



Redes Ambientales y Computación Ubicua

Bloque I. Computación Ubicua

Sesión 2

2018-19

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
 - Conexión remota (más formas)
 - Spyder
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I

I. Introducción

- **Objetivos**

- Conexión en remoto con el equipo
- Spyder
- Captura de audio con Raspberry Pi
- Captura de audio con Python
- Presentación del ejercicio “entrega I”

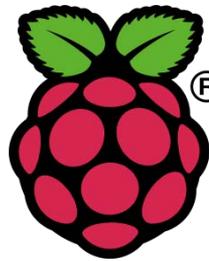
I. Introducción

- Descripción de las sesiones
 - **Sesión 1:**
 1. Introducción a Raspberry Pi – Comp. Ubicua
 2. Tutorial de Python con algunos ejercicios
 - **Sesión 2:**
 1. Ampliación info sobre RaspPi
 2. Audio con Raspberry Pi
 3. Planteamiento y codificación del ejercicio “*Entrega 1*”
 - **Sesión 3:**
 1. Sense Hat
 2. Planteamiento y codificación del ejercicio “*Entrega 2*”

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
 - Conexión remota (más formas)
 - Spyder
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I

2. Raspberry Pi (conexión SSH)



Para establecer conexiones remotas se hará uso de SSH. Para ello, Raspbian debe tener activado en servidor de ssh.

Se puede activar y desactivar en el menú de configuración de raspbian. En las unidades del laboratorio ya está activado pero si no fuera así, puede modificarlo ejecutando en un terminal

```
sudo raspi-config
```

 (Opción 5. *Interfacing Options*)

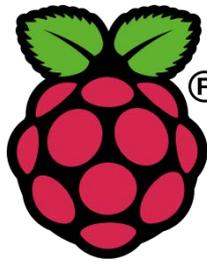
Desde otro equipo es posible establecer una conexión remota con ssh.

- Desde Windows es necesario algún programa específico (por ejemplo Putty)
- Desde Linux, en un terminal puede ejecutar

```
ssh pi@192.168.5.10X
```

 (pedirá sólo el password → raspberry)

2. Raspberry Pi (conexión SSH)



Pruebe a abrir un programa desde la sesión **ssh** que tiene abierta, por ejemplo **leafpad**. El terminal le dirá que no es posible.

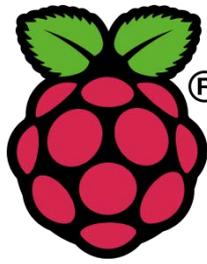
Abra otra conexión ssh (puede cerrar la actual y abrir una nueva o abrir otra en otro terminal), pero añada el argumento **-X** (mayúscula)

A continuación abra de nuevo leafpad en esa nueva conexión.

```
ssh pi@192.168.5.10X -X
```

2. Raspberry Pi

(Sesiones con *screen*)



Instale screen

```
apt-get install screen
```

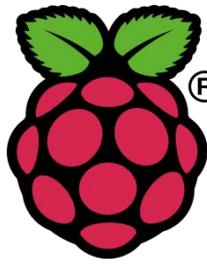
Información en terminal con `man screen`

Operaciones básicas

1. Inicie screen, simplemente tecleando screen
2. Abrir una nueva sesión `screen -S nombreSesion`
 - Cree dos sesiones con nombres diferentes
3. Listar las sesiones abiertas `screen -ls`
4. Listar los procesos asociados a screen
`ps axf |grep -i screen`

2. Raspberry Pi

(Sesiones con *screen*)



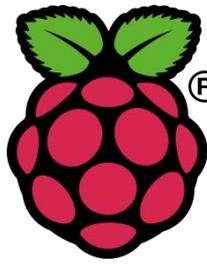
Combinaciones de teclas básicas (Ctrl+a + letra)

1. Ctrl+a ? → Muestra la ayuda con las combinaciones de teclas (^ equivale a Ctrl)
2. Ctrl+a d → Cierra el terminal (si es el último sale de screen)
3. Ctrl+a c → Abre un nuevo terminal
4. Ctrl+a n → Cambia de una terminal a otro
5. Ctrl+a Shift+c → Nos permite renombrar el terminal, por defecto todos se llaman bash
6. Ctrl+a d → Cierra el terminal sin cerrar los procesos en ejecución
7. Ctrl+a S → Divide el terminal

Más información en

https://es.wikibooks.org/wiki/Introducci%C3%B3n_a_Linux/Screen

2. Raspberry Pi (Sesiones con *screen*)



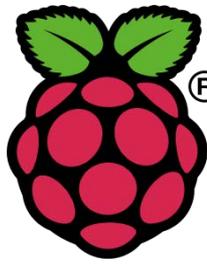
¿Y por qué es más útil que una simple conexión ssh?

Porque permite dejar sesiones en ejecución, salir del terminal, y luego recuperarlas de nuevo.

Vea este tutorial para ver cómo, sección “Detaching” y “Reattach”

<https://www.rackaid.com/blog/linux-screen-tutorial-and-how-to/#detach>

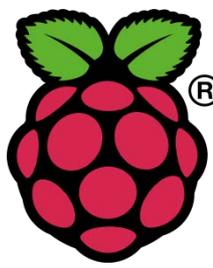
2. Raspberry Pi (Tunel ssh con VNC)



¿Cómo puedo conectarme desde casa al equipo en el laboratorio?

- Red privada / IP pública
- NAT /NAPT - Redirección de IPs y puertos
- Servicios NO-IP (Dynamic IP Address)
<http://noip.com>
- Puerto VNC – habitualmente 5900
(se puede comprobar con nmap, por ejemplo)

2. Raspberry Pi Spyder



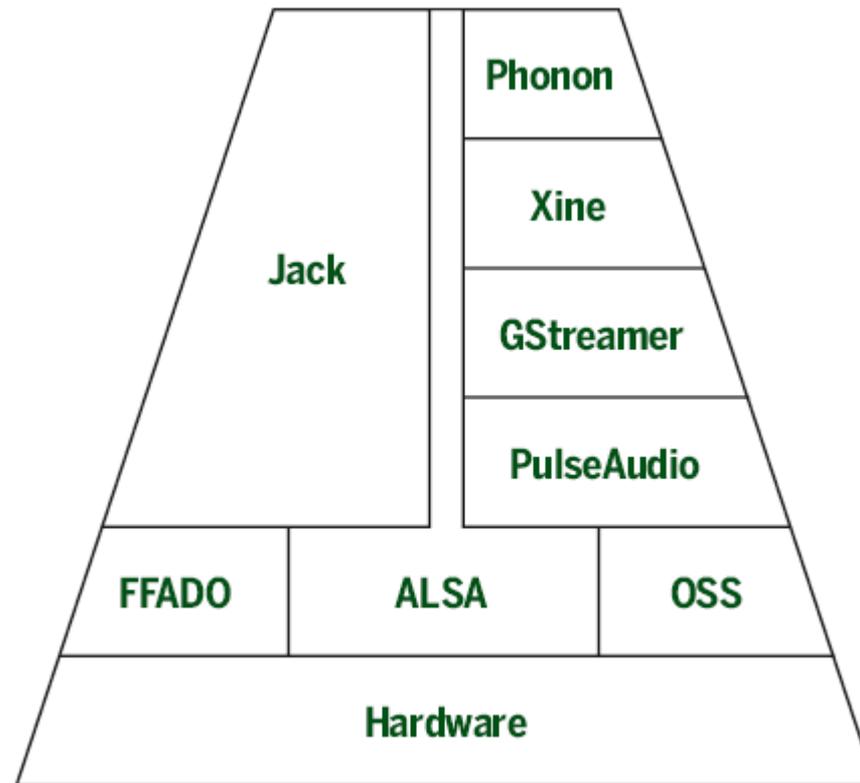
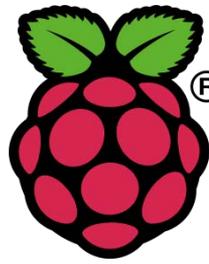
Entorno de programación inspirado en Matlab.

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I

3. Adquisición de audio. RP

Audio en Raspberry pi + Linux



Fuente <http://tuxradar.com/content/how-it-works-linux-audio-explained>
Información sencilla en <https://www.learnigitalaudio.com/how-linux-audio-works-vs-windows-audio-2017>

3. Adquisición de audio. RP

Audio en Raspberry pi + Linux

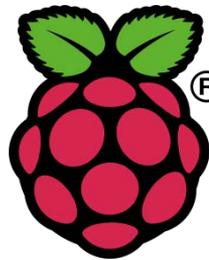


Opciones para capturar Audio en Raspberry Pi:

1. Micrófono analógico con un conversor A/D
2. Micrófono USB (incluso **webcam**)
3. Tarjeta de sonido externa por USB
 - Wolfson Audio Card (37€)
 - Creative Sound Blaster Play (26 €)
 - Otras (ver Amazon, p.e.) (dealxtreme 3€)

3. Adquisición de audio. RP

Audio en Raspberry pi + Linux

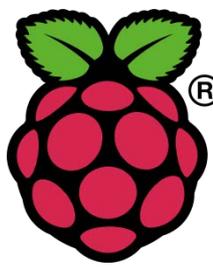


Captura con Micrófono USB o tarjeta de sonido USB

1. Consultar dispositivos soportados
 - http://wiki.linuxaudio.org/wiki/raspberrypi#supported_usb_devices
2. Consultar el consumo de corriente.
 - Rasp. Pi no da mucha potencia por el USB
3. Cargar el driver de sonido
 - `sudo modprobe snd_bcm2835`
 - `lsmod` (debe aparecer el módulo `snd_bcm2835`),
 - para hacer un filtrado de la información, se puede añadir al final del comando `| grep snd`
4. Listar los dispositivos (diferentes opciones)
 - `lsusb`
 - `arecord -l / alsamixer (F6) / usb-devices`

3. Adquisición de audio. RP

Audio en Raspberry pi + Linux



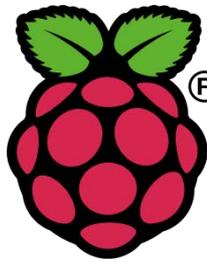
Captura con Micrófono USB o tarjeta de sonido USB

5. Ajustes de Alsamixer `alsamixer`
 - F6 Seleccionar dispositivos de grabación
 - F4, muestra los dispositivos de captura. F5 todos los dispositivos
 - Cursores para desplazar entre elementos y subir/bajar volumen
 - Tecla M para “Mute”
6. Guardar configuración `sudo alsactl store`

(No suele ser necesario)
5. Comprobando si los altavoces funcionan
`speaker-test -t wav -c 5` (ctrl + c para matar el proceso)
Si no hay sonido, comprobar con raspi-config la salida de audio
Menú 7. Advanced Options → A4. Audio
→ I. Force 3.5mm(‘Headphone’) Jack

3. Adquisición de audio. RP

Audio en Raspberry pi + Linux



Captura con Micrófono USB o tarjeta de sonido USB

8. Grabar sonido `arecord -D plughw:1,0 -f cd captura.wav`
 - Opciones interesantes `-d 10 -F 10 -vv`
 - Otras, ver `man` (opciones para formatos)
9. Para reproducir (por salida HDMI o por jack)
 - `aplay [-D plughw:1] captura.wav`
10. Por último, para reproducir mientras graba
 - `arecord -D plughw:1,0 -f cd | aplay`

Más información y archivos relacionados

`/proc/bus/devices`

`/proc/asound/cards`

`/card1` (archivos asociados)

3. Adquisición de audio. RP

Python – Llamada al SO



La forma más sencilla de tratar con el audio con Python es usando directamente una llamada al sistema.

Por ejemplo, si se desea reproducir un archivo de audio en mp3, se puede hacer con el siguiente código.

```
import os
#llamada al OS para que ejecute un comando
os.system ('mpg123 -q mi_archivo.mp3 &')
```

Si da algún problema es posible que requiera instalar mpg123, hágalo con `apt-get install mpg123`

Pruebe con **arecord** y **aplay** para capturar y reproducir en formato wav. Cuidado con el símbolo “&” al final del comando.

Nota: si quiere parar un proceso y no sabe cómo, una forma rápida de hacerlo es con `killall`. Por ejemplo, si tras ejecutar su código se queda un audio en reproducción y quiere pararlo puede ejecutar

```
killall mpg123
```

3. Adquisición de audio. RP

Python – Mediante PyAudio



Otra alternativa es emplear el paquete PyAudio, que opera sobre PortAudio proporcionando una interfaz más sencilla para grabar y reproducir audio.

<https://pypi.org/project/PyAudio/>

Si no está instalado, se puede instalar con

```
sudo pip3 install pyaudio
```

Requiere tener instalado previamente, entre otros, estos módulos o paquetes:

```
sudo apt-get install libportaudio0 libportaudio2  
libportaudiocpp0 portaudio19-dev
```

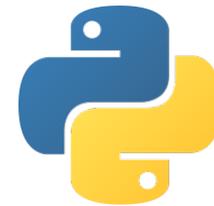
Puede encontrar ejemplos de uso de PyAudio en:

<http://people.csail.mit.edu/hubert/pyaudio/>

<https://www.programcreek.com/python/example/52624/pyaudio.PyAudio>

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I



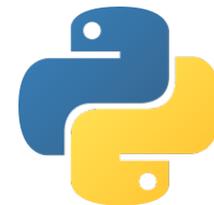
4. Pitch

“El Pitch representa la percepción de la frecuencia de un sonido. Es uno de los principales atributos de los tonos musicales , junto con la duración , intensidad , timbre y localización de la fuente de sonido . El pitch se compara con "superior" e "inferior" en el sentido de que permite la construcción de melodías.”

Fuente: <https://interfazmusica.blogspot.com.es/2011/02/que-es-pitch.html>

En otros términos, se puede definir como la frecuencia fundamental del sonido percibido.

Aunque no es sencillo, esta característica se puede emplear para detectar si una voz corresponde a un hombre o una mujer. En términos generales, se suele asociar un valor de frecuencia de pitch pitch “alto” a una voz de mujer, mientras que uno más bajo a hombres.



4. Pitch

En una voz humana este valor puede variar desde los 40Hz para las voces más graves (de hombre) hasta los 600 Hz de las voces más agudas (niños y mujeres).

Para obtener este valor se pueden emplear diferentes algoritmos basados por ejemplo en FFT, Zero Crossing, Autocorrelación, Redes Neuronales, KNN, etc.

En esta sesión se verá:

- Obtención sencilla con FFT
- Información básica para obtener con autocorrelación
- Empleando una librería específica (Aubio)

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I



5. AUBIO

AUBIO (<https://aubio.org>)

En esta sesión se va a emplear la librería aubio para la extracción del pitch de un sonido fuente.

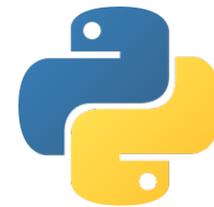
Para instalar aubio,

```
sudo pip3 install aubio
```

En la misma web de aubio encontrará la información sobre su uso así como ayudas y ejemplos.

CONTENIDOS

1. Introducción a la sesión 2
2. Raspberry Pi (ampliación)
3. Adquisición de audio
4. Pitch
5. Aubio
6. Entregable I



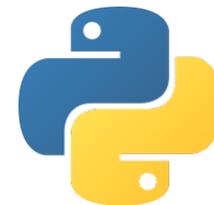
6. Entregable I

Entregable I. Detección del género con el *pitch* y actuación sobre un relé.

Para este entregable debe realizar un código que, capture la voz del interlocutor, detecte si es hombre o mujer, y en función de esto active o desactive uno de los relés.

Indicaciones.

1. Deberá establecer un umbral para la separación hombre/mujer. Para ello deberá hacer algunas pruebas. Dado que no es una tarea trivial, basta que funcione con los sujetos de pruebas empleados.
2. Para la captura del audio tiene dos opciones. O bien captura durante unos segundos y procesa, o bien crea un código que detecte cuando está hablando y cuando hay silencio.
3. El código debe ejecutarse en bucle infinito, mostrando por pantalla los resultados obtenidos.



6. Entregable I

Entregable I. Detección del género con el *pitch* y actuación sobre un relé.

Evaluación.

1. Método de obtención del archivo de audio:
 - Si se emplea una llamada al SO – 2 puntos.
 - Si se emplea PyAudio – 4 puntos.
2. Forma en la que se pasa la captura al código que procesa los datos:
 - Los datos se obtienen a partir de un archivo – 2 puntos.
 - Los datos obtenidos con PyAudio se guardan en un buffer y se van procesando conforme se van adquiriendo - 4 puntos.
3. Funcionamiento del código.
 - El código captura el audio, actúa sobre el relé, se ejecuta en bucle infinito y muestra de resultados por pantalla – 1 punto.
 - El código acierta al menos el 60% de la veces – 1 punto.