

1. Annexos

Una vegada explicat de que tracta el projecte i explicades les justificacions pertinents, s'ha considerat oportú fer una xicoteta guia d'us, així com explicar el codi i la informació tècnica més rellevant.

1.1. Guia ràpida d'us

El funcionament és molt simple, però no obstant això, va a fer-se una xicoteta explicació dels conceptes bàsics.

Una vegada està engegada la Raspberry Pi i amb el programa corrent (açò ho tindrà que fer la persona que instal·le la Raspberry Pi), l'usuari sols es te que preocupar de que quan li aplegue un correu electrònic, entrar a la pàgina web.

En aquesta, l'usuari trobarà quatre botons amb els quals pot fer totes les opcions que dóna el projecte.

El primer botó et dóna l'opció d'obrir la porta, i al polsar-lo, l'indicador d'informació que hi ha a la pàgina anuncia que la porta està oberta. El segon botó és per a tancar la porta, i al igual que passa amb el primer botó, al polsar el segon també s'indica que la porta està tancada. El tercer botó és per a parar de veure el *streaming*. Quant el polses, el *streaming* es para i apareix un missatge com a que s'ha parat. El quart botó és per a engegar altra vegada el *streaming*, i quant el polses, simplement comença a reproduir-se el *streaming*.

Cada botó te una etiqueta amb la funció que fa, és a dir, al primer botó posa: Obrir porta; al segon botó: Tancar porta; al tercer botó: Parar *streaming*; i al quart botó posa: Començar *streaming*.

Cal comentar que si entres a la pàgina web quant has rebut el correu electrònic, trobaràs el *streaming* en marxa, mentre que si entres a la pàgina web sense haver rebut el correu, no estarà el *streaming* en marxa, però el podràs engegar tu amb el botó corresponent.

1.2. Codi font

En aquest apartat va a mostrar-se tot el codi font utilitzat al projecte, així com la explicació necessària per a que totes les persones ho puguem entendre.

Els scripts utilitzats no són massa extensos, així que es mostrarà primer el codi font i després es farà una xicoteta explicació del codi, i si fa falta, explicar amb més detall alguna funció o alguna part de codi.

Boto.py

```
import smtplib
import RPi.GPIO as GPIO
import time
import os
import mimetypes
import email
import email.mime.application
import sys

msg = email.mime.Multipart.MIMEMultipart()
msg['Subject'] = 'Control de Porta'
msg['From'] = 'lauraquilis93@gmail.com'
msg['To'] = 'lauraquilis93@gmail.com'

body = email.mime.Text.MIMEText("Hola, algú a tocat el timbre i vol entrar!! Entra a veure-ho des de ací: (URL personalitzada per a cada client). ")
msg.attach(body)

GPIO.setmode(GPIO.BCM)

GPIO.setup(23, GPIO.IN, pull_up_down=GPIO.PUD_UP)

while True:

    input_state = GPIO.input(23)
    if input_state == False:
```

```
print('Boto apretat')
s = smtplib.SMTP('smtp.gmail.com:587')
s.starttls()
s.login('lauraquilis93@gmail.com','LaUrita93')
s.sendmail('lauraquilis93@gmail.com',['lauraquilis93@gmail.com'],
msg.as_string())
s.quit()
time.sleep(0.2)
os.system('./stream.sh')
time.sleep(0.2)
GPIO.cleanup()
```

Explicació Boto.py

El primer què és fa en aquest script és importar totes les llibreries necessàries per al correcte funcionament. Seguidament es crea el correu electrònic, definint l'assumpte, el correu electrònic que envia el correu, el correu electrònic que el rep i el cos o missatge que s'envia al correu electrònic. Es defineix el mode dels GPIO a BCM i s'assigna que el pin GPIO 23 siga d'entrada i a baix voltatge. Seguidament, el script espera que el boto siga polsat, i una vegada ocorre això, s'envia el correu i s'executa **stream.sh**, fitxer que s'explica més avant. Per últim, es neteja la configuració del GPIO.

Stream.sh

```
cd /home/pi/mjpg-streamer/mjpg-streamer  
killall mjpg_streamer  
./mjpg_streamer -i "/input_uvc.so -d /dev/video0 -y" -o "/output_http.so -w ./www" &
```

Explicació Stream.sh

Els fitxers amb extensió .sh són scripts amb comandaments per a la terminal de qualsevol sistema operatiu que utilitzi o provinga de Linux.

Una vegada explicat que són els fitxers .sh es passa a explicar el script Stream.sh.

El primer que fa aquest script és entrar a la carpeta on es troba el programa MJPG-streamer amb el comandament: `cd /home/pi/mjpg-streamer/mjpg-streamer`. Després mata qualsevol procés que continga al nom `mjpg_streamer`. I, per últim, crea el procés amb el qual el programa MJPG-streamer comença a reproduir el *stream* amb la càmera que s'indica i per la ruta que s'indica.

Parar.sh

```
cd /home/pi/mjpg-streamer/mjpg-streamer
```

```
killall mjpg_streamer
```

Explicació de parar.sh

Aquest fitxer també es un .sh, que es recorda que és un scripts amb comandaments per a la terminal de qualsevol sistema operatiu que utilitze o provinga de Linux.

El que fa aquest script és primer entrar a la carpeta on es troba el programa MJPG-streamer i una vegada allí, mata tot procés que tinga al nom mjpg_streamer.

Led.py

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.OUT)

for i in [1]:
    print "S'encen el LED"
    GPIO.output(17, GPIO.HIGH)
    time.sleep(5)
    GPIO.output(17, GPIO.LOW)
    print "S'apaga el LED"

GPIO.cleanup()
```


ledRoig.py

```
import RPi.GPIO as GPIO
import time

GPIO.setmode(GPIO.BCM)
GPIO.setup(21, GPIO.OUT)

for i in [1]:
    print "S'encen el LED"
    GPIO.output(21, GPIO.HIGH)
    time.sleep(5)
    GPIO.output(21, GPIO.LOW)
    print "S'apaga el LED"

GPIO.cleanup()
```

Explicació de Led.py i ledRoig.py

Aquests dos scripts s'expliquen junts ja que són pràcticament iguals, menys el pin GPIO associat, que en el cas de **Led.py** és el pin GPIO 17, i en el cas de **ledRoig.py** és el pin GPIO 21. Una vegada aclarit açò, comencem amb l'explicació.

El primer que es fa en aquests dos scripts és importar les llibreries necessàries per al correcte funcionament. Després es defineix el mode dels pins GPIO a BCM i s'assigna el pin 17 i 21, respectivament, com a pin d'eixida. Seguidament, amb un bucle for de tan sols 1 element, és fa primer que s'encenga, donant-li així 3V, i fent que s'encenga el LED; després s'està 5 segons encès, i per últim, fa que s'apague el LED, donant-li 0V. Per últim, es neteja la configuració del GPIO.

StreamOn.py

```
import os
import time

for i in [1]:

    os.system('./stream.sh')
    time.sleep(0.2)
```

StreamOff.py

```
import os
import time

for i in [1]:

    os.system('./parar.sh')
    time.sleep(0.2)
```

Explicació StreamOn.py i StreamOff.py

Aquests dos scripts s'expliquen junts per què són pràcticament iguals, sols canvia que en el **StreamOn.py** s'executa **stream.sh** i en **StreamOff.py** s'executa **parar.sh**. Una vegada aclarit açò, comencem amb l'explicació.

Els dos scripts comencen important les llibreries necessàries per al correcte funcionament. Després, amb un bucle for de tan sols un element, executa el fitxer .sh corresponent(**stream.sh** a **StreamOn.py** i **parar.sh** a **StreamOff.py**). Cal recordar que aquests fitxers .sh ja s'han explicat amb anterioritat a aquest mateix apartat.

Index.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Control de Porta</title>
<link href='http://fonts.googleapis.com/css?family=Righteous' rel='stylesheet' type='text/css'>
<script type="text/javascript" src="jquery.min.js"></script>
<script>
function cargar(div, desde)
{
    $(div).load(desde);
}
</script>

<style>
iframe {
    visibility:hidden;
}

.estat {
    width: 222px;
    height: 34px;
    border: 1px #0000FF solid;
    background:#F33;
    padding: 10px 10px 10px 10px;
    padding-top: 20px;
    float:left;
}
```

```
.stream {  
  width: 512px;  
  height: 384px;  
  border: 1px #0000FF solid;  
  background:#F33;  
  padding: 10px 10px 10px 10px;  
  padding-top: 20px;  
}
```

```
.boto {  
  width: 100px;  
  height: 25px;  
  border: 1px #0000FF solid;  
  background:#09F ;  
  padding: 10px 10px 10px 10px;  
  float:left;  
}
```

```
.contingut {  
  width:512px;  
}
```

```
body {  
  background-color:#1c2328;  
  color:#FFF;  
}
```

```
.stream {  
  background-color:#0C3;  
  color:#000;  
}
```

```
h1 { color: #fff; font-family: 'Righteous', cursive; font-size: 65px; font-weight: normal; line-height: 60px; margin: 10px 0 20px; text-transform: uppercase; text-shadow: 2px 2px 0 #000, margin: 10px 0 24px; text-align: center; }
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<center>
```

```
<h1>Control de Porta</h1>
```

```
<div id="stream" class="stream">
```

```
;
```

```
</div>
```

```
<br>
```

```
<div class="contingut">
```

```
<a href="obrir2.php" target="finestra" onClick="carregar('#boto', 'obrir.php')">
```

```
<div class="boto" >
```

```
Obrir Porta
```

```
</div>
```

```
</a>
```

```
<a href="tancar2.php" target="finestra" onClick="carregar('#boto', 'tancar.php')">
```

```
<div class="boto" >
```

```
Tancar Porta
```

```
</div>
```

```
</a>
```

```
<a href="#" onClick="carregar('#stream', 'off.php')">
```

```
<div class="boto" >
```

```
Stream Off
```

```
</div>
```

```
</a>
<a href="on2.php" target="finestra" onClick="carregar('#stream', 'on.php')">
  <div class="boto" >
    Stream On
  </div>
</a>

<br><br><br><br><br>
<div class="estat" id="boto">
  Porta Tancada
</div>
</div>
<iframe name="finestra">
</iframe>

</center>

</body>
</html>
```


Explicació de index.php

El index.php el que bàsicament fa és mostrar tot el contingut de la pàgina web, és a dir, mostrar el títol, els botons, el requadre on es retransmet el *streaming* i el requadre on es mostra informació; cadascun dels elements té el seu estil i els seus colors. També el que fa és que quant polses un botó, carrega un altre .php per a que faja la funció i a més a més carrega la informació en el “div” corresponent amb el següent fragment de codi :

```
<script>
function carregar(div, desde)
{
    $(div).load(desde);
}
</script>
```

El “div” amb el nom boto és el que correspon al requadre informatiu, i el “div” amb el nom stream és el que correspon al requadre on es retransmeteix el *stream*.

Obrir.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>
<?php
    $a- exec("sudo python /var/www/led.py");
    echo $a;

?>
</body>
</html>
```

Tancar.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>
<?php
    $a- exec("sudo python /var/www/ledRoig.py");
    echo $a;

?>
</body>
</html>
```

On.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<body>

<?php
    $a- exec("sudo python /var/www/streamOn.py");
    echo $a;

?>
</body>
</html>
```

Explicació de **obrir.php**, **tancar.php** i **on.php**

Aquest codi el que fa és donar-li valor a la variable `a`. Aquest valor és una ordre per a que execute un comandament en Python. Seguidament, s'imprimeix la variable `a` amb un `echo`, i amb això es fa que s'execute.

```
$a- exec("sudo python /var/www/led.py");  
echo $a;
```

Cada `.py` executat en cada `.php` ja ha sigut explicat amb anterioritat a aquest mateix apartat.

La resta del codi de **obrir.php**, **tancar.php** i **on.php** és el encapçalat i la finalització del cos i HTML.

Off.php

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
</head>

<style>
.stream {
    background-color:#F33;
    color:#000;
}

</style>
<body>
<h1>Stream Parat</h1>
<?php
    $a- exec("sudo python /var/www/streamOff.py");
    echo $a;

?>
</body>
</html>
```

Explicació **off.php**

Aquest codi el que fa és el mateix que els arxius **obrir.php**, **tancar.php** i **on.php**, però a més també canvia el color del requadre on està emetent-se el *stream* i apareix “Stream Parat” dins del requadre.

La resta del codi de **off.php** és el encapçalat i la finalització del cos i HTML.

1.3. Informació tècnica

En aquest apartat s'han introduït algunes fulles dels *datasheets* dels components utilitzats. Cada fulla conté al peu de la pàgina una lleugera explicació que serveix per a situar-se en el component.

Sols s'han posat algunes fulles ja que no es veia necessari incloure-les totes. El document del SoC Broadcom BCM2835, per exemple, ja ocupa 205 pàgines, més que l'actual memòria.

Les fulles de característiques de cada un dels components han sigut inserits en format PDF a la memòria USB, on també es pot trobar l'actual memòria, per si es vol consultar. De totes formes una possible font per a adquirir-los des de Internet es escrivint la seua referència i la paraula *datasheet* en la famosa pàgina web : www.google.com.

De la Raspberry Pi no s'ha trobat cap *datasheet* en format PDF; no obstant, a la pàgina web:

<https://www.raspberrypi.org/documentation/hardware/raspberrypi/>

es pot trobar tota la informació del hardware que porta la Raspberry Pi.



Instruction Manual

Congratulations on your purchase of the PLAYSTATION®Eye camera. Before using the camera, carefully read this manual and retain it for future reference. This camera is designed for use with the PLAYSTATION®3 computer entertainment system.

WARNING

To avoid potential electric shock or starting a fire, do not expose the camera to rain, water or moisture.

Precautions

Safety

This product has been designed with the highest concern for safety. However, any electrical device, if used improperly, has the potential for causing fire, electric shock or personal injury. To ensure accident-free operation, be sure to follow these guidelines.

- Observe all warnings, precautions and instructions.
- Do not use the camera if it functions in an abnormal manner.
- If the camera does not function properly, call the appropriate PlayStation® Customer Service number which can be found:
 - within every PlayStation®, PlayStation®2 and PLAYSTATION®3 format software manual; and
 - on our website playstation.com

Using and handling the PLAYSTATION®Eye camera

- Do not expose the camera to high temperatures, high humidity or direct sunlight (operate in an environment where temperatures range between 5°C and 35°C).
- Do not allow liquid or small particles to get into the camera.
- Do not put heavy objects on the camera.
- Never disassemble or modify the camera.
- Do not twist the cable or pull it forcibly.
- Do not throw or drop the camera, or physically damage it any way.
- Do not touch the metal parts or insert foreign objects into the PLAYSTATION®Eye camera's USB connector.
- Do not place or use the camera on an unstable surface.
- Make sure you have enough room to play.

Cleaning the PLAYSTATION®Eye camera

- Dust may build up on the camera after an extended period of time.
- Before cleaning the camera, disconnect it from the PLAYSTATION®3 system for safety.
- Wipe the camera's surface with a soft, dry cloth.
- Use air pressure to clean off any dust that has collected on the lens.

Notes

- This product contains small parts, which, if removed, may present a choking hazard to children.
- The camera should be cleaned by an adult, or cleaned under close adult supervision.
- Do not use a damp cloth to clean the camera. If water gets inside, it may cause the camera to malfunction.
- Do not use benzene, paint thinner or other chemicals, as these may damage the camera.
- When using a commercially available cleaning cloth, follow the instructions supplied with the cloth.

Connecting the PLAYSTATION®Eye camera

Securely insert the camera's USB connector into one of the USB connectors on the front of the PLAYSTATION®3 system. The blue LED power indicator on the front of the camera should light up to indicate that it is now ready to use.



Disconnecting the PLAYSTATION®Eye camera

To disconnect the camera, gently pull the camera's USB connector out of the PLAYSTATION®3 system. Do not pull on the cable itself as this may damage it. Removal of the camera during play will result in the termination of the live feed.

Using the PLAYSTATION®Eye camera

It is recommended to stand between 1.5 metres and 2.0 metres away from the camera.



Ensure the lens angle selector is rotated to the appropriate setting. Turn the lens angle selector to the right so that the lens angle marker is in line with the blue lens angle indicator for "wide angle view". Alternatively, turn the lens angle selector to the left so that the lens angle marker is in line with the red lens angle indicator for "standard view".

If your camera is positioned on the floor, simply tilt the camera for extra elevation or swivel the camera base around to reposition it. Players of different heights should tilt the angle of the camera for extra convenience. DO NOT move the whole camera, just tilt it gently.

Make sure that the background behind you is as motionless as possible, as background motion may hinder gameplay. Ensure there is enough room to play – watch out for shelves, doors, walls, pets and above all, other people.

For further information about the PLAYSTATION®Eye camera's voice/video chat feature, please refer to the PLAYSTATION®3 System Software User's Guide at playstation.com

Troubleshooting

The PLAYSTATION®Eye camera does not recognise your movements.

- You may be experiencing problems with the light in your room.
- Check that there are no flickering lights and that you are evenly lit from the front.
- Avoid using low-energy bulbs and fluorescent tubes if possible and switch on any extra available lights.

The on-screen buttons activate without your interaction.

- There may be background motion behind you.
- Make sure the background behind you is as motionless as possible.
- If there is a window in your background, consider drawing the curtains and using artificial lights instead.

Your on-screen image appears too bright or too dark.

- The lighting in the room is not suitable.
- If your on-screen image looks too dark, switch on all available lights in the room and point them at you.
- If your on-screen image looks too bright, draw the curtains and use artificial light instead.

You cannot see yourself on-screen.

- The camera is not plugged in.
- The camera is not positioned correctly.

Interface	Connector	Power consumption	Dimension (approx.)
USB 2.0	USB	DC5V, Max. 500mA	84 x 67 x 57mm

Weight (approx.)	Cable length (approx.)	Operation temperature	Video capture
173g	2 metres	5°C to 35°C	640 x 480 pixel

Video format	Lenshead
Uncompressed or JPEG	2.1 F-stop, <1% distortion, fixed focus (25cm to ∞ at 75° FOV)

Field of view	Frame rate
56° to 75° FOV zoom lens	640 x 480 at 60 frames/second 320 x 240 at 120 frames/second

Design and specifications are subject to change without notice.

©2007 Sony Computer Entertainment Europe. "PS", "PlayStation" and "PLAYSTATION" are registered trademarks of Sony Computer Entertainment Inc. All rights reserved.

Figura 1. Fulla de característiques de la càmera Eye Toy

BROADCOM **BCM2835 ARM Peripherals**

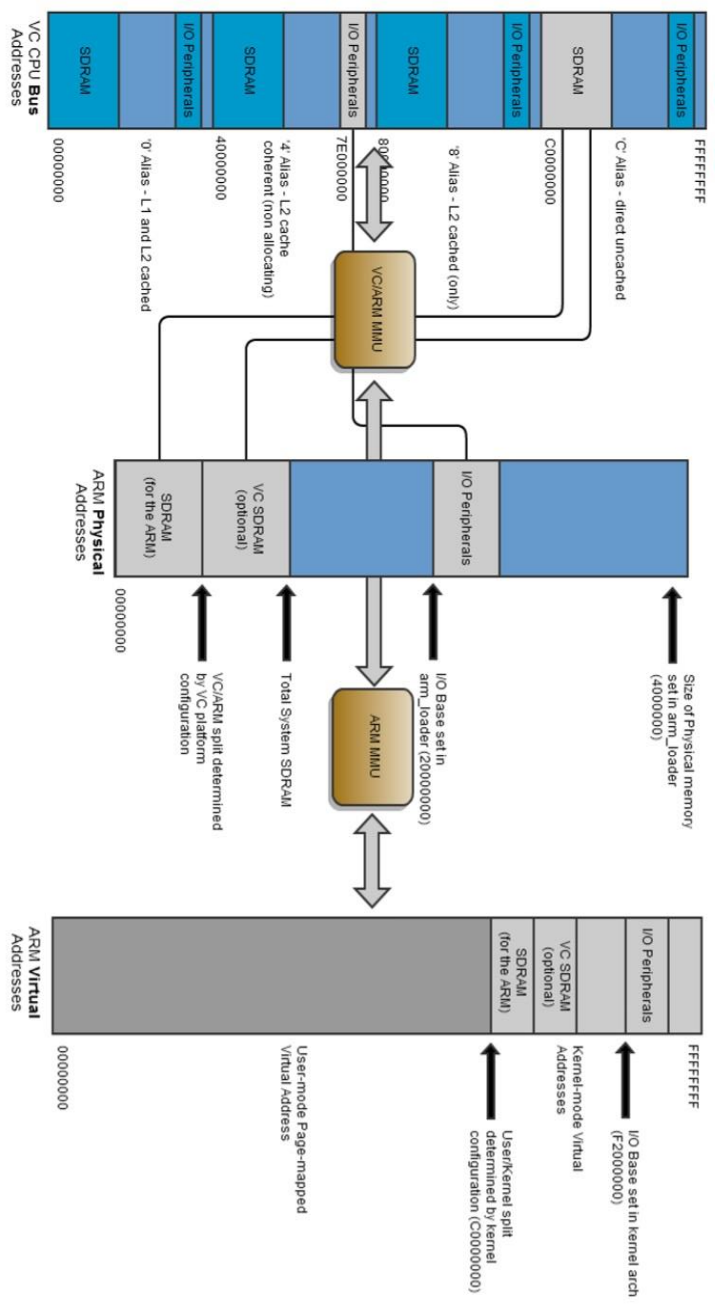


Figura 2. Mapa de direcciones del Broadcom BCM2835

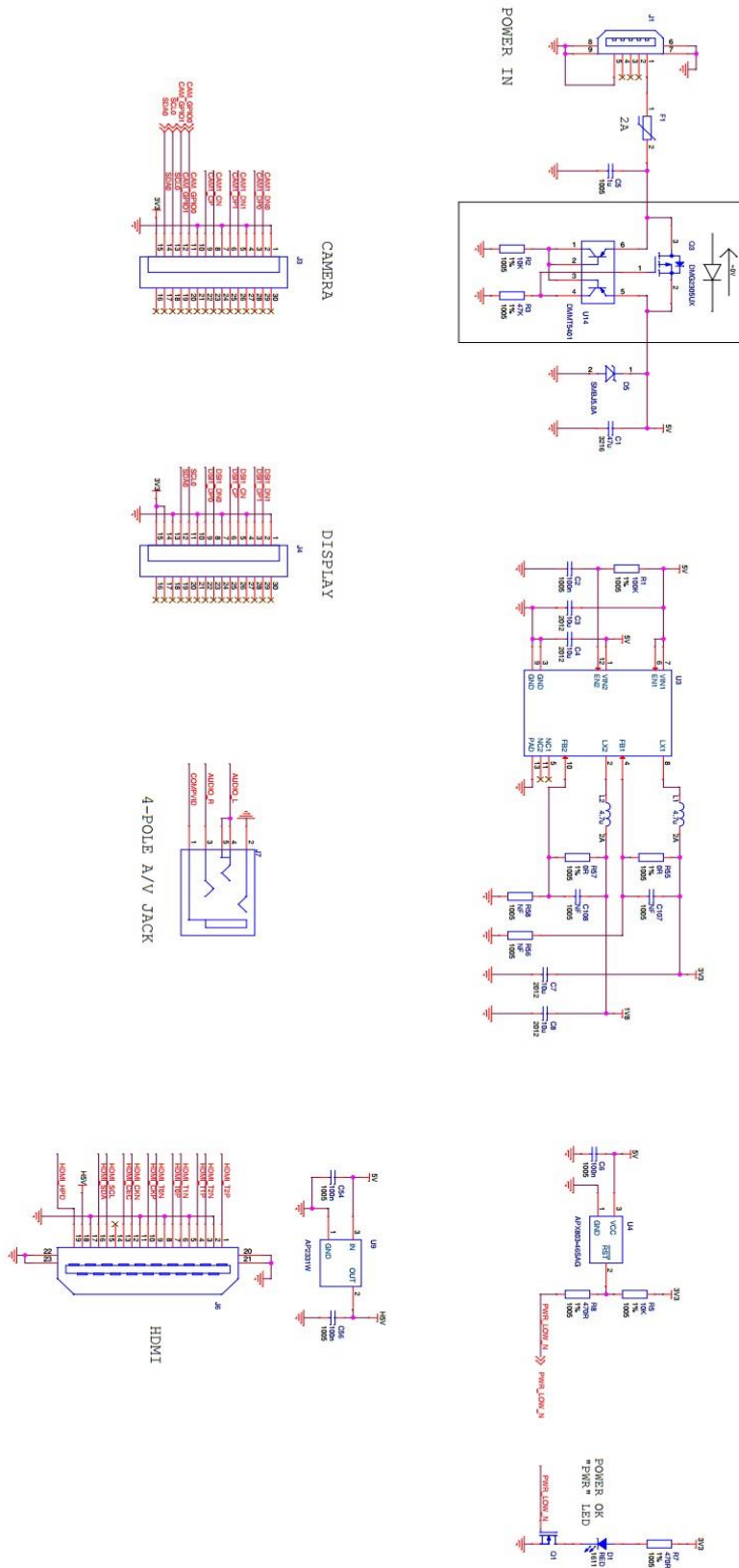


Figura 3. Esquema de alguns components de la Raspberry Pi

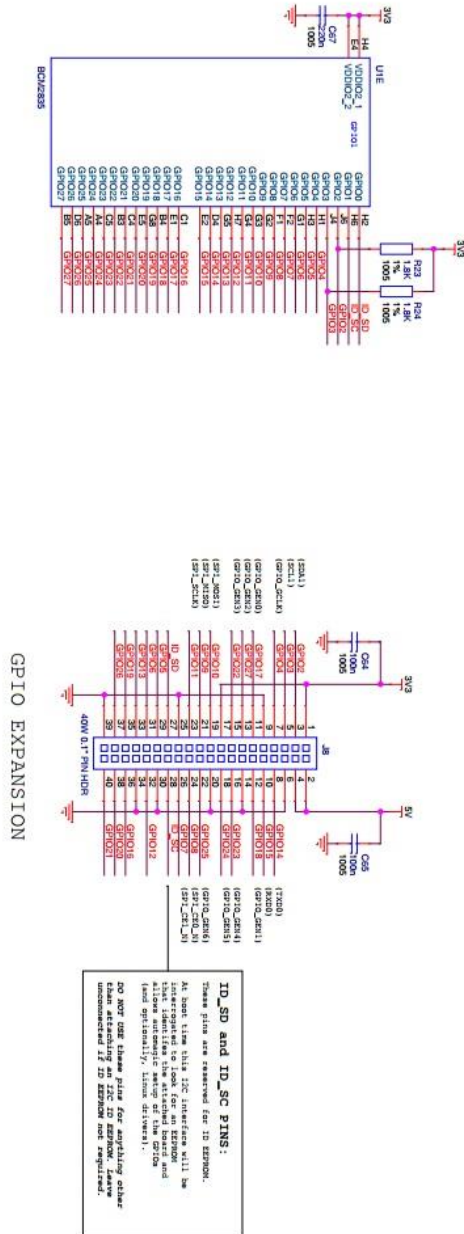


Figura 4. Esquema de alguns components de la Raspberry Pi

1.4. Contingut distribuït

S'adjunta una memòria portàtil USB que conté la present memòria, junt amb totes les fulles de característiques dels components utilitzats per si es desitja fer alguna consulta.