

Implement Enhanced TV dongle on Raspberry pi

Embedded Operating System Implementation Final proposal

Group 7

Member: 王澤元、吳現任、李昇穆、李官翰、許又展

● 研究動機

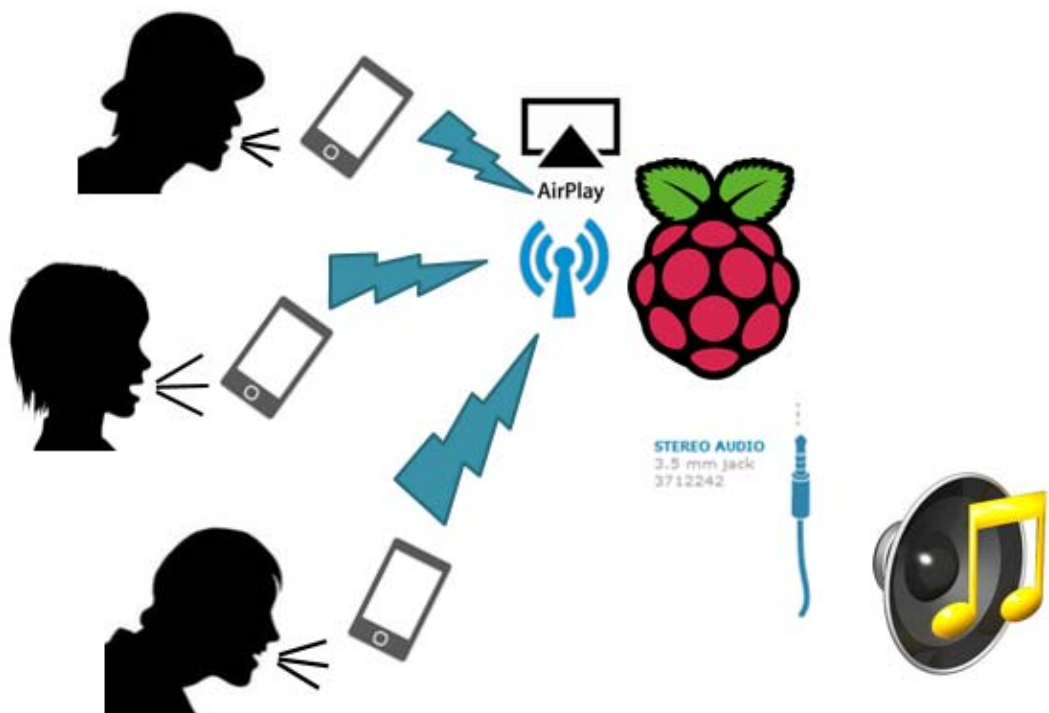
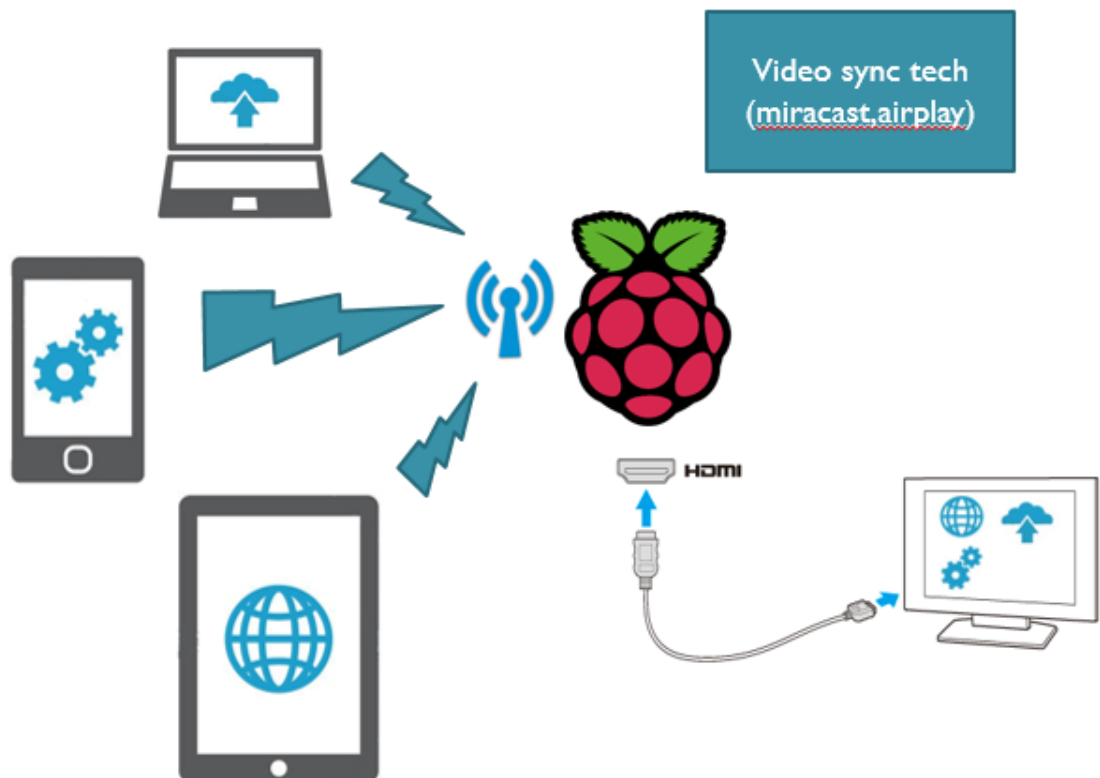
現今電視棒存有一些缺點，例如播放畫面的延遲太過嚴重，可能會需要特定的裝置或設備才能使用，以及儲存空間不符合使用者需求等等，我們這次期末 project 的主要目的是想改進目前市面上的電視棒，除了讓原有功能更加優化之外，再添加一些新的功能，使電視棒能夠普及化、更讓大眾接受。

● 功能介紹

我們想讓電視棒具有的其中一個功能是能讓電視棒在會議或演講中使用。目前大家在聽演講的模式多半是看著台上的講者和台上大銀幕放的投影片，或者是手邊得拿著一張張紙本資料，當主講人想和台下觀眾有所互動時，台下觀眾想發表意見可能得等一段時間後，麥克風才能傳到手上，多少會有些不方便，這時若有一項工具能使得上台台下互動更方便的話，相信能增進大家的參與感和討論度，在此同時，觀眾們也可透過此工具去觀看主講人分享的投影片或資料。

不過這次的 project 因為時間因素，對電視棒內部的程式碼還不甚了解，無法作些適當的改寫，所以只先用手機去達到我們想做的功能－麥克風和檔案共享。

● 系統架構



● 實作方法

功能一 手機麥克風播放器

為了將 Raspberry Pi 變成播放手機聲音的媒介，我們使用了 shairport 套件模擬 AirPort Express，將音響接上 Raspberry Pi 變成可以接收 AirPlay 聲音串流的 server 藉此達到在演講中無須遞送麥克風，只需使用手機透過 AirPlay 將音訊傳送到 Raspberry Pi，即可當作麥克風使用。

首先，要先連上 shairport 在 GitHub 的網頁。找到下載連結後複製下來，再用 SSH 連進 Raspberry Pi，貼上連結位址：

```
wget https://github.com/abrasive/shairport/archive/master.zip
```

下載這個壓縮檔，解壓縮後再安裝其他套件：libao、PulseAudio、avahi。

輸入以下指令：

```
sudo apt-get -y install libssl-dev libavahi-client-dev libasound2-dev
```

```
sudo apt-get -y install libao-dev libpulse-dev
```

下載完所需套件後就編譯（make），完成後再安裝一個套件，接著就能直接執行它：

```
sudo apt-get -y install avahi-daemon
```

```
./shairport -a 'RPi2 AirSpeaker'
```

「RPi2 AirSpeaker」是我們自訂的名稱，當啟動 shairport 這個服務 Raspberry Pi 開始模擬 AirPort Express，會在手機端顯示出來

以上是 Raspberry Pi 端的環境設置，而手機端須先安裝模擬 Air Play 的 app，確定和 Raspberry Pi 在同一網域下後開啟，即可搜尋到已變成喇叭的 Raspberry Pi 裝置，手機透過該 app 也具備麥克風功能。

功能二 檔案共享

我們還做了檔案共享的功能，若是小容量需求可在作業系統 SD 卡上分享一個目錄又或者使用 USB 即可；若有大容量的需求（如：影音檔）就可以外接一個行動硬碟，但得注意 Raspberry Pi 供給的電源供應器是否夠力，不夠力時要外接一個可供電的 USB hub，並將行動硬碟接到 USB hub 上。

實作步驟：

1. 使用 ssh client 連線到 Raspberry Pi 後，更新最新的套件。

```
sudo apt-get update && sudo apt-get -y upgrade
```

2. 將 USB 插入 Raspberry Pi，執行 `sudo fdisk -l` 確認裝置名稱。
3. 接著用 `df -h` 的指令查看 USB 掛載(mount)的情況，系統會自動將 USB 隨身碟掛載到 `/media/BAC1-268E` 目錄，其中 BAC1-268E 這個目錄是

系統抓取磁碟裝置的 UUID 所產生的，通常為了辨識及設定會更改設定讓 USB 掛載到指定的目錄下。

4. 用下列指令在 /media 目錄下建立一個名為 share 的目錄：

```
sudo mkdir /media/share
```

5. 修改 /etc/fstab 檔案內容，這個檔案內容一般記錄著系統掛載裝置的資訊，在此檔案的最後一行新增：

```
/dev/sda1 /media/share vfat defaults,dmask=0000,fmask=0000 0 0
```

加好後存檔並離開編輯器，再用 **sudo shutdown -r now** 指令使 Raspberry Pi 重新開機

6. 當開機完成後從新連線確認 USB 隨身碟是否掛載到 /media/share
7. 安裝 samba, samba-common-bin, smbclient 三個套件。

```
sudo apt-get install samba samba-common-bin smbclient
```

8. 編輯 /etc/samba/smb.conf 設定我們用來分享檔案的 USB 隨身碟，smb.conf 記錄著 samba 服務的相關參數，

```
sudo vi /etc/samba/smb.conf
```

確認 workgroup = WORKGROUP

找到此段程式碼：

```
#===== Share Definitions =====
```

確認參數每一行前面都加入了分號，例如：

```
[homes]
```

```
comment = Home Directories
```

```
browseable = no
```

改成

```
;[homes]
```

```
;comment = Home Directories
```

```
;browseable = no
```

再移到最後一行加入下列設定：

```
[share]
```

```
comment = USB Share
```

```
path = /media/share
```

```
writeable = Yes
```

```
create mask = 0777
```

```
directory mask = 0777
```

```
browseable = Yes
```

```
public = yes
```

```
guest ok = no
```

```
valid users = pi
```

9. 使用 `sudo service samba restart` 指令重新啟動 samba 服務。
10. 使用 `sudo testparm -s` 指令確認剛剛修改 smb.conf 的資料有載入到 samba 服務。
11. 接著建立 windows 連線的帳號，由於 Raspbian 已經內建了一個 pi 的使用者，所以我們只要用 `sudo smbpasswd -a pi` 這個指令就可以讓 windows 用 pi 這個用戶登入，在執行 `sudo smbpasswd -a pi` 後會要求你輸入密碼，這是往後從 windows 所登入的密碼。
12. 接著使用 windows 進行連線，首先打開檔案總管在路徑下輸入 `\\Raspberry Pi 的網路 IP 位置\share`，輸入用戶及密碼，接著，我們就可以在檔案總管上看到前面我們所設定的 USB 中的檔案內容了。
13. 相關 samba 的日誌紀錄 (log) 系統會存放在 `/var/log/samba` 目錄下，這裡會記錄一些資訊如果遇到一些障礙或是查詢各台機器的登入資訊。

● 實作遇到的問題

雖然能透過上述實作方式讓手機達到麥克風和檔案共享功能，但實際 demo 時發現聲音的傳遞延遲較高，我們認為可能是手機 app 端所造成的 delay，若時間充足的話將手機 app 重寫應該可以解決傳輸延遲的問題。

● 未來展望 & Business model

若是之後時間和研究資源較為充足的話，我們希望能直接透過修改電視棒內部的程式碼，去改進延遲問題，以及將麥克風功能加進去，另外我們使用的 app 希望也能做些修改而讓它不只適用於手機。

我們希望將此 device 商品化，消費者族群鎖定公司和 presenter / speaker，此商品能解決現實中在會議或演講中的不便利，讓會議中的參與者能共享自己的檔案也可將意見回饋上傳增加開會效率，當在大型演講場合台下需要問問題時也可省去遞麥克風的時間，直接透過手機連接到我們的產品就可及時發聲，與觀（聽）眾互動達到增加 QoS 的效果，藉此提升購買意願。