

Ensor-384 + Raspberry Pi

Preparation sequence of a Raspberry Pi 5B card to work with the "Ensor-384" audio recorder card

1.- Elements needed to start

- Raspberry Pi 5 B
- Original Raspberry Pi power supply
- Good original 8GB micro SD card + micro SD USB recorder
- 64GB or larger USB Flash memory
- A computer with internet connection with Windows or Linux OS
- The ensor384.zip file downloaded from the internet

2.- Install the Raspberry Pi OS Lite (64) operating system without desktop for RPI 5B

Use Raspberry Pi Imager by configuring the user and password,
enabling ssh server

user: ensor

password: ***** any

Recommended to use the Putty program to send commands via ssh to the Raspberry Pi

3.- Connect the Ensor-384 card to the Raspberry pi and this, in turn, connect it to the internet via ethernet cable. Turn on the system

3.- update operating system

sudo apt update

sudo apt upgrade

4.- run sudo raspi-config

enable autologin with console 1 - S5 - B2

enable ssh 3 - I1

enable i2c 3 - I5

expand filesystem 6 - A1

sudo apt install python3-smbus

sudo apt install i2c-tools

sudo reboot

5.- Connect a USB flash drive, with a capacity greater than or equal to 64GB, to the raspi-3b+ card and check its label. The USB flash drive will be formatted later to exFAT

sudo fdisk -l

-> /dev/sda1

The USB flash drive will be used to store audio recordings

1 hour of 2-channel stereo audio recording at 192000 m/s and 32-bit takes up 5.4 GB

1 hour of 2-channel stereo audio recording at 384000 m/s and 32-bit takes up 10.8 GB

6.- Mount the USB flash drive at system startup

The USB flash drive will be mounted at the following path

/media/ensor/Ensor384

create the ensor and ensor/Ensor384 directories in /media

sudo mkdir -p /media/ensor/Ensor384

add the following line to the /etc/fstab file

```
sudo nano /etc/fstab
```

```
/dev/sda1 /media/ensor/Ensor384 auto auto,user,rw,umask=000,nofail,x-  
system.device-timeout=10 0 0
```

check that the modifications in /etc/fstab are correct

```
sudo mount -a
```

```
sudo systemctl daemon-reload
```

```
sudo reboot
```

to check that the USB flash drive is mounted correctly

7.- The USB flash drive will be formatted in exFAT

This operation can be easily performed in Windows,
or in Linux, as indicated below

```
sudo mkfs.exfat -n LABEL /dev/sdXn
```

in our case

```
sudo fdisk -l to see /dev/sdXn
```

```
sudo mkfs.exfat -n Ensor384 /dev/sda1
```

From a computer with Linux or Windows:

The compressed file ensor384.zip will be downloaded from the Internet, which includes the directory Sensor and the audio configuration file ensor.conf

Unzip ensor384.zip and copy the ensor.conf file and the Ensor directory to the USB flash drive

Plug the USB flash drive into the Raspberry Pi and power on the system

On the USB flash drive, there will be the recording configuration file:

ensor.conf

and the Ensor directory, which will be copied to:

/home/ensor/

```
cp -dr /media/ensor/Ensor384/Ensor /home/ensor/
```

once the Ensor directory is copied, it can be deleted from the USB flash drive if desired

```
rm -dr /media/ensor/Ensor384/Ensor
```

To record files larger than 4GB.

Installing the exfat-fuse program on the Raspberry Pi

```
sudo apt install exfat-fuse
```

8.- Copy the compiled dts file from the "Ensor-384" audio recording card to
/boot/firmware/overlays

The tlv320adcx140-overlay.dts file is located in the /home/ensor/Ensor/dts/tlv320adcx140 directory

```
/home/ensor/Ensor/dts/tlv320adcx140/tlv320adcx140-overlay.dts
```

We will have to compile it with the following command:

```
sudo dtc -@ -H epapr -O dtb -o tlv320adcx140-overlay.dtbo -Wno-unit_address_vs_reg  
tlv320adcx140-overlay.dts
```

and copy the compiled file tlv320adcx140-overlay.dtbo to /boot/firmware/overlays/

```
sudo cp /home/ensor/Ensr/dts/tlv320adcx140/tlv320adcx140-overlay.dtbo  
/boot/firmware/overlays/
```

9.- Modify the file /boot/firmware/config.txt

```
sudo nano /boot/firmware/config.txt
```

```
dtparam=i2c=on
```

```
#dtparam=i2s=on
```

```
dtoverlay=tlv320adcx140-overlay
```

```
#disable the default audio cards
```

```
#dtparam=audio=on
```

```
#The following line cannot be disabled because the screen does not work when restarting
```

```
dtoverlay=vc4-kms-v3d
```

```
#disable wifi and buetooth to avoid noise and consumption
```

```
dtoverlay=disable-wifi
```

```
dtoverlay=disable-bt
```

10.- Restart the system and check that the ADCX140 recording card has been installed

```
arecord -l
```

check that card is 0 and device is 0

card 0: ADCX140 [ADCX140], device 0:

which are the parameters required for the recording program

```
/home/ensor/Ensor/eea.py
```

11.- Check that the card records audio at a sampling rate of 192000 m/s and 32bits

The recording program is located in `/home/ensor/Ensor/eea.py`

and the recording configuration file, `ensor.conf`, is located on the USB flash drive

If the USB flash drive does not have the `ensor.conf` configuration file saved, the `eea.py` program creates a default one with the data shown below and the program will be executed with that data.

By default, the system does not record until "record=yes" is specified

Recording parameters can be modified with any text editor

```
nano /media/ensor/Ensor384/ensor.conf
```

[DEFAULT]

```
record = no           #enable recording or not <yes, no>
```

#by default, do not record

date = 01/12/2024 #date of the day of recording

```
file name = test 192kHz 32bits      #file name
```

[sound.format]

sampling = 192000

#sampling frequency in Hz

bits = 32

#number of bits per sample, fixed value at 32bits

[record.time]

total_time = 10

#total recording time in #seconds

file_time = 10

#since the files are very large, it is interesting to split
#them into several

delay = 0

#wait time, in seconds, from start, before start
#recording

index = 0

#index of possible recordings

#that are made on the same day, each recording
#increases the index by one unit

A directory structure will be created depending on the current date and the file names will depend on the current hour and minute ending with the file number.

modify the configuration file to record

nano /media/ensor/Ensor384/ensor.conf

[DEFAULT]

record = yes #enable recording or not <yes, no>

. . . #by default, it does not record

And try recording

sudo /home/ensor/Ensor/eea.py

It is necessary to use “sudo” to modify the system date.

the program has a delay of 10 seconds to give time for the USB flash drive to mount correctly when we automatically record when the system starts.

12.- Record when the system starts

we will use the rc-local.service service

we will modify the /etc/rc.local file

```
sudo nano /etc/rc.local
```

insert the following line, so that the recorder.sh script is executed when the system starts

```
/home/ensor/Ensor/recorder.sh
```

Also, add the following line so that GPIO4 (PIN 7) is set to high level when the system starts and the Ensor-384 card is enabled

```
/usr/bin/pinctrl -p GPIO4 op dh pn
```

13.- Script program /home/ensor/Ensor/recorder.sh

```
nano /home/ensor/Ensor/recorder.sh
```

```
#!/bin/sh
```

```
sudo /home/ensor/Ensor/eea.py
```

```
sudo shutdown -h now
```

what this script does is start recording according to the ensor.conf configuration and when it finishes, it shuts down the system so that it does not consumes power.

14.- Backup the card

windows: Win32DiskImager

linux: `dd if=/dev/sda of=/image/path`

from raspi: `sudo dd if=/dev/mmcblk0 of=/media/ensor/Ensor384/raspbian.img bs=1M status=progress`

15.- Prepare Raspberry Pi to record at 384000 samples/s and 32 bits

The system will be prepared with everything necessary to compile the kernel

`sudo apt install git bc bison flex libssl-dev make`

The current kernel source files will be downloaded

`git clone --depth=1 https://github.com/raspberrypi/linux`

16.- The "menuconfig" tool requires ncurses

`sudo apt install libncurses5-dev libncursesw5-dev`

17.- We will use spdif-receiver as a digital audio interface

Since spdif-receiver supports a maximum sampling rate of 192000 samples/s, we will have to modify spdif-receiver to support 384000 samples/s

The file is located in `/home/ensor/Ensor/linux/sound/soc/codecs`

`nano /home/ensor/Ensor/linux/sound/soc/codecs/spdif_receiver.c`

Where it says 192000, put 384000 instead and compile the kernel

```
#define STUB_RATES SNDRV_PCM_RATE_8000_384000
```

```
#define STUB_FORMATS (SNDRV_PCM_FMTBIT_S16_LE | \ SNDRV_PCM_FMTBIT_S20_3LE | \  
SNDRV_PCM_FMTBIT_S24_LE | \ SNDRV_PCM_FMTBIT_S32_LE | \  
SNDRV_PCM_FMTBIT_IEC958_SUBFRAME_LE)
```

18.- Compile kernel 6.6.59-v8-16k+ Raspberry Pi OS Lite (64) for Raspberry Pi 5

```
cd linux
```

```
KERNEL=kernel_2712
```

```
make bcm2712_defconfig
```

```
make menuconfig
```

```
make -j6 Image.gz modules dtbs
```

```
sudo make -j6 modules_install
```

```
sudo cp /boot/firmware/$KERNEL.img /boot/firmware/$KERNEL-backup.img
```

```
sudo cp arch/arm64/boot/Image.gz /boot/firmware/$KERNEL.img sudo
```

```
cp arch/arm64/boot/dts/broadcom/*.dtb /boot/firmware/
```

```
sudo cp arch/arm64/boot/dts/overlays/*.dtb* /boot/firmware/overlays/
```

```
sudo cp arch/arm64/boot/dts/overlays/README /boot/firmware/overlays/
```

```
sudo reboot
```

11.- Check that the card records audio at a sampling rate of 384000 samples per second and 32 bits

The recording program is located in /home/ensor/Ensor/eea.py

Edit the configuration file to record at 384000 m/s and 32 bits for 10 seconds

```
nano /media/ensor/Ensor384/ensor.conf
```

```
[DEFAULT]
```

```
record = yes          #enable whether to record or not <yes, no>
                        # by default, it does not record
```

```
date = 01/12/2024
```

```
file_name = prueba_384Ks_32bits
```

```
[sound.format]
```

```
sampling = 384000
```

```
bits = 32
```

```
[record.time]
```

```
total_time = 10      # total recording time in seconds
```

```
file_time =10        # since the files are very large,
                      # it is better to divide them into several files
```

```
delay = 0            # wait time, in seconds, from # the start, before starting to record
```

```
index = 0            # incremental index in successive recordings
                      # made on the same date
```

try to record

```
sudo /home/ensor/Ensor/eea.py
```

If the system records perfectly, perform another recording test, but this time automatically from the start of the system, to do this,

simply leave the configuration file as it is and restart the system

```
sudo reboot
```

When the system boots, it will start recording as indicated in the configuration file.

Since the recording configuration file, ensor.conf, is on the USB flash drive, it can be modified with any text editor in both Windows and Linux

In this simple way, automatic recordings can be made.

Simply set the configuration that is needed at any given time.

21.- Backing up the card

windows: Win32DiskImager

linux: `dd if=/dev/sda of=/path/to/image`

from RPI5: `sudo dd if=/dev/mmcblk0`

`of=/media/ensor/Ensor384/raspbian_64_6_6_59_v8_16k.img bs=1M status=progress`