

Creació d'un multiefectes d'àudio sobre Raspberry Pi i Simulink

ARNAU MAÑOSA I MARTÍ

Grau en Enginyeria de Tecnologies i Serveis de Telecomunicació

Aplicacions multimèdia basades en processament del senyal

Què és un efecte d'àudio?

01

Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

Què és un efecte d'àudio?

01

Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

02

Paràmetres de les ones sonores

Amplitud
To
Període

Què és un efecte d'àudio?

01

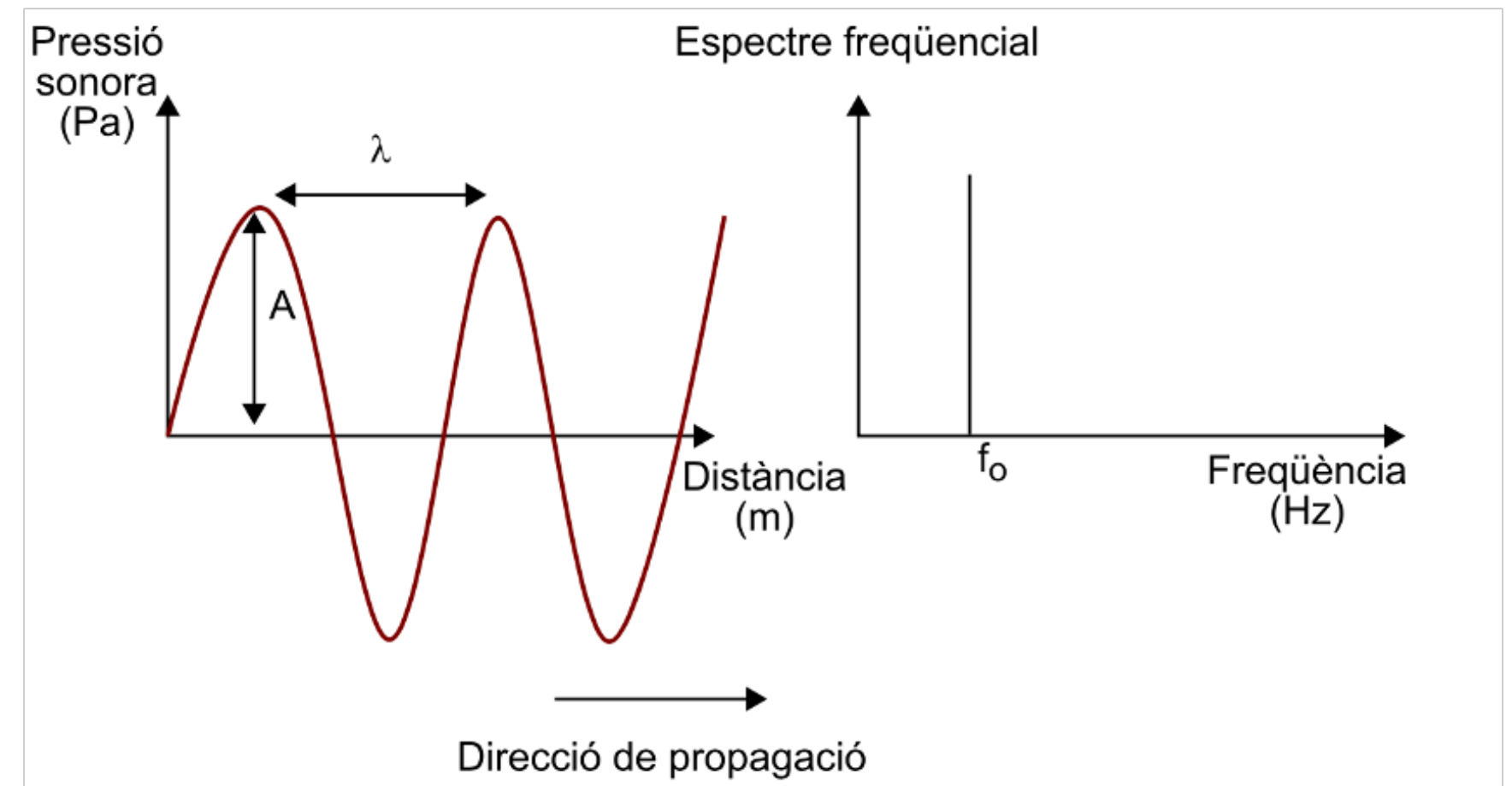
Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

02

Paràmetres de les ones sonores

Amplitud
To
Període



Què és un efecte d'àudio?

01

Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

02

Paràmetres de les ones sonores

Amplitud
To
Període

03

Procès de digitalització

Mostratge
Quantificació
Codificació

Què és un efecte d'àudio?

01

Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

02

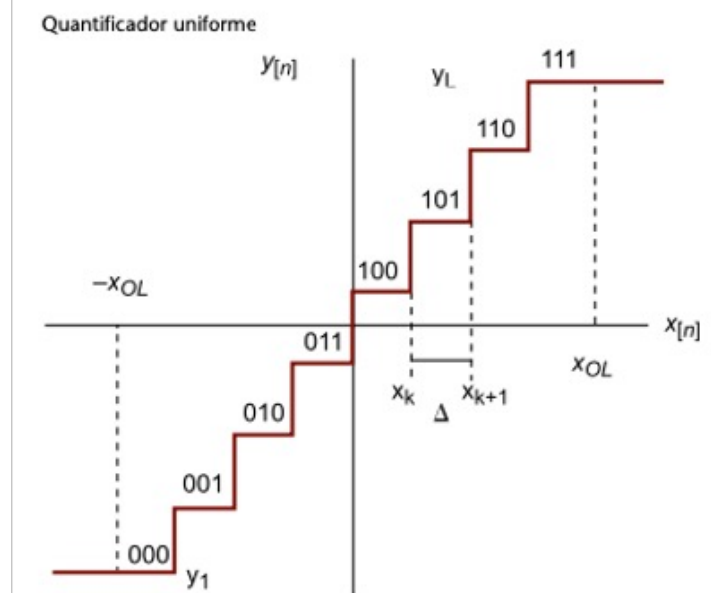
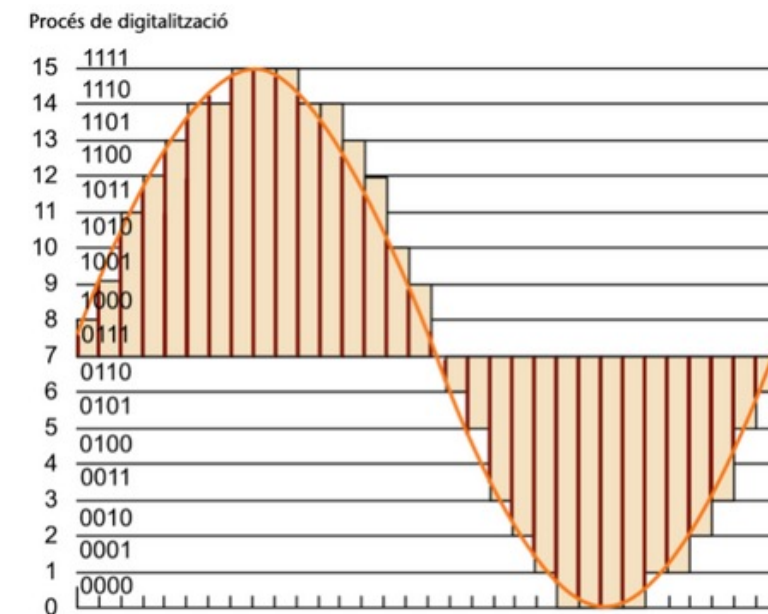
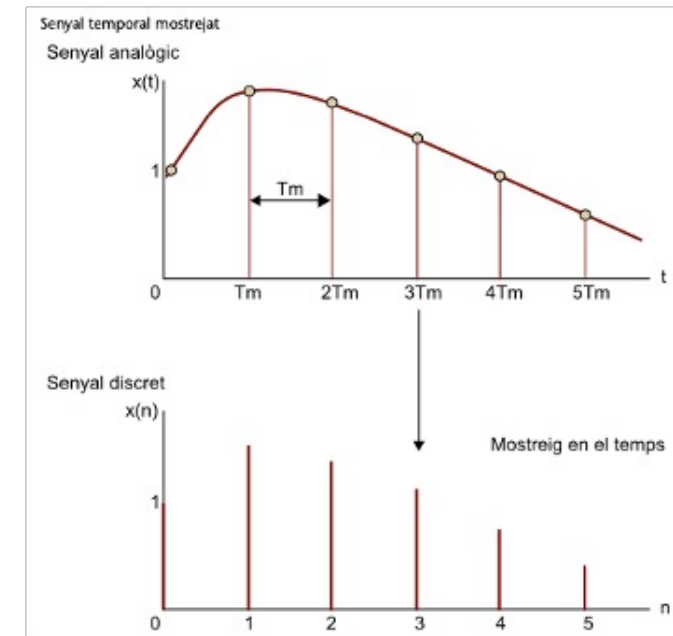
Paràmetres de les ones sonores

Amplitud
To
Període

03

Procès de digitalització

Mostratge
Quantificació
Codificació



Què és un efecte d'àudio?

01

Un efecte d'àudio és qualsevol modificació que li apliquem a un àudio

A qualsevol dels seus paràmetres

02

Paràmetres de les ones sonores

Amplitud
To
Període

03

Procès de digitalització

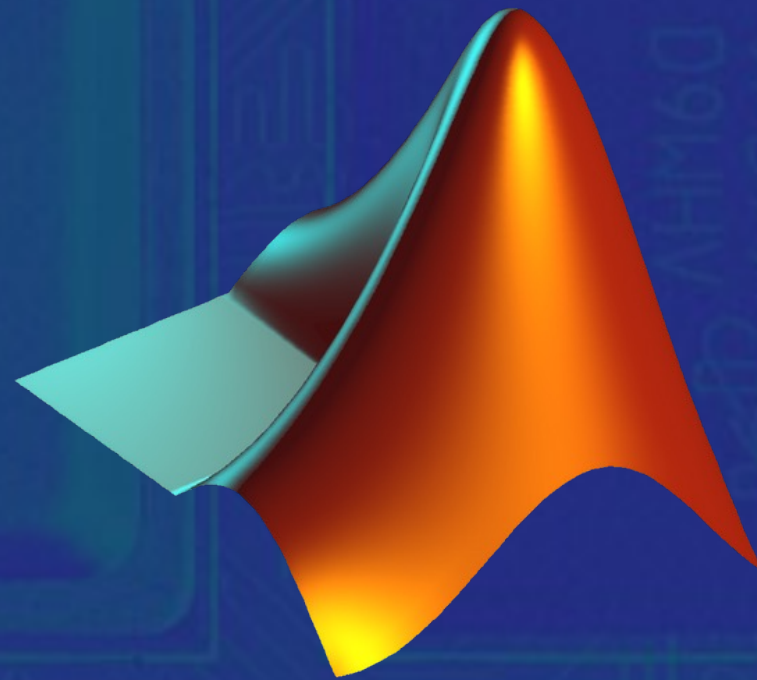
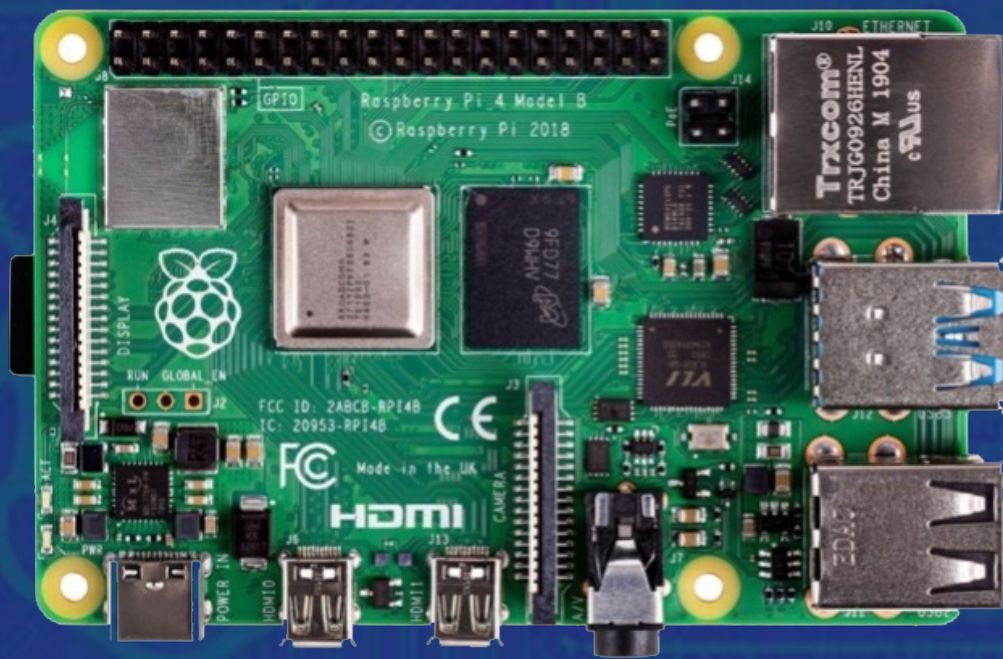
Mostratge
Quantificació
Codificació

04

Classificació dels efectes

Modificadors del temps
Modificadors del contingut freqüencial
Modificadors de la dinàmica

Plataforma de desenvolupament



Bypass

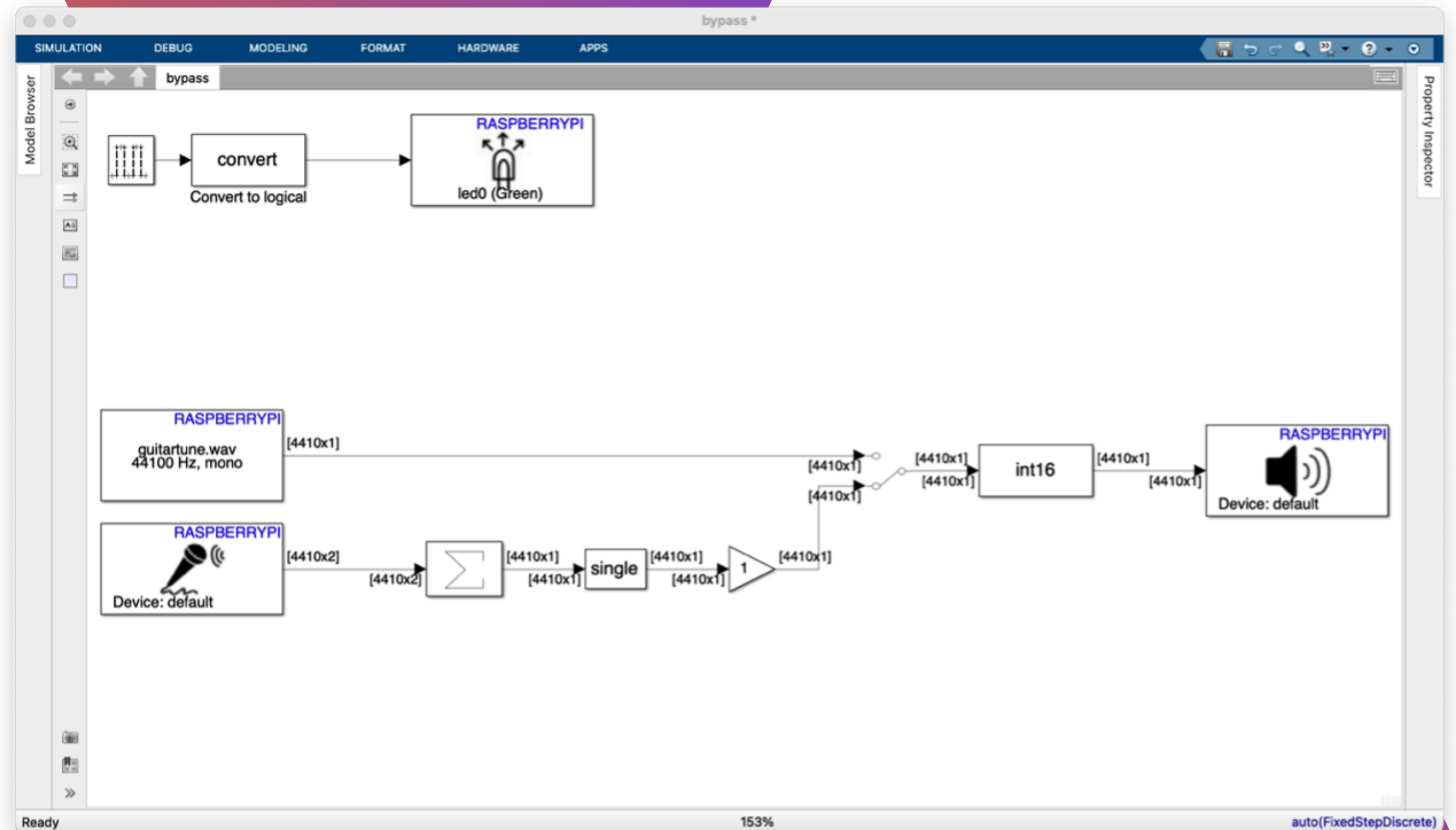
Cadena simple,
només
digitalizta i
reprodueix

Treballem amb una freqüència de mostreig de 44.1kHz i 16 bits de profunditat.

A la part superior veiem “la mosca” per saber si el model s'executa correctament.

Abans del mòdul de reproducció tornem a codificar el senyal a enter de 16 bits.

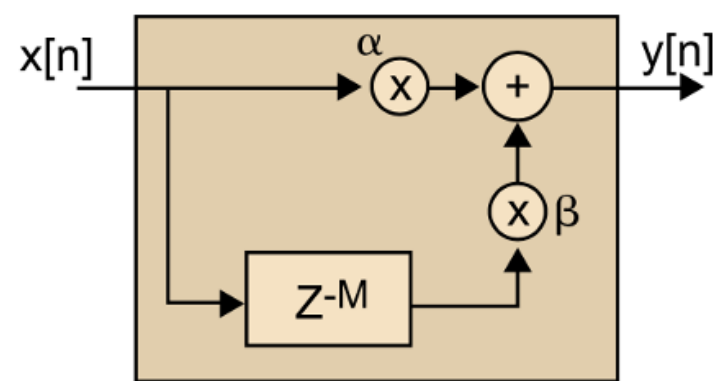
Al fer els càlculs de la latència s'observen els primers problemes.



Delay

L'efecte més senzill que podem aplicar

$$y[n] = x[n] + x[n - m]$$

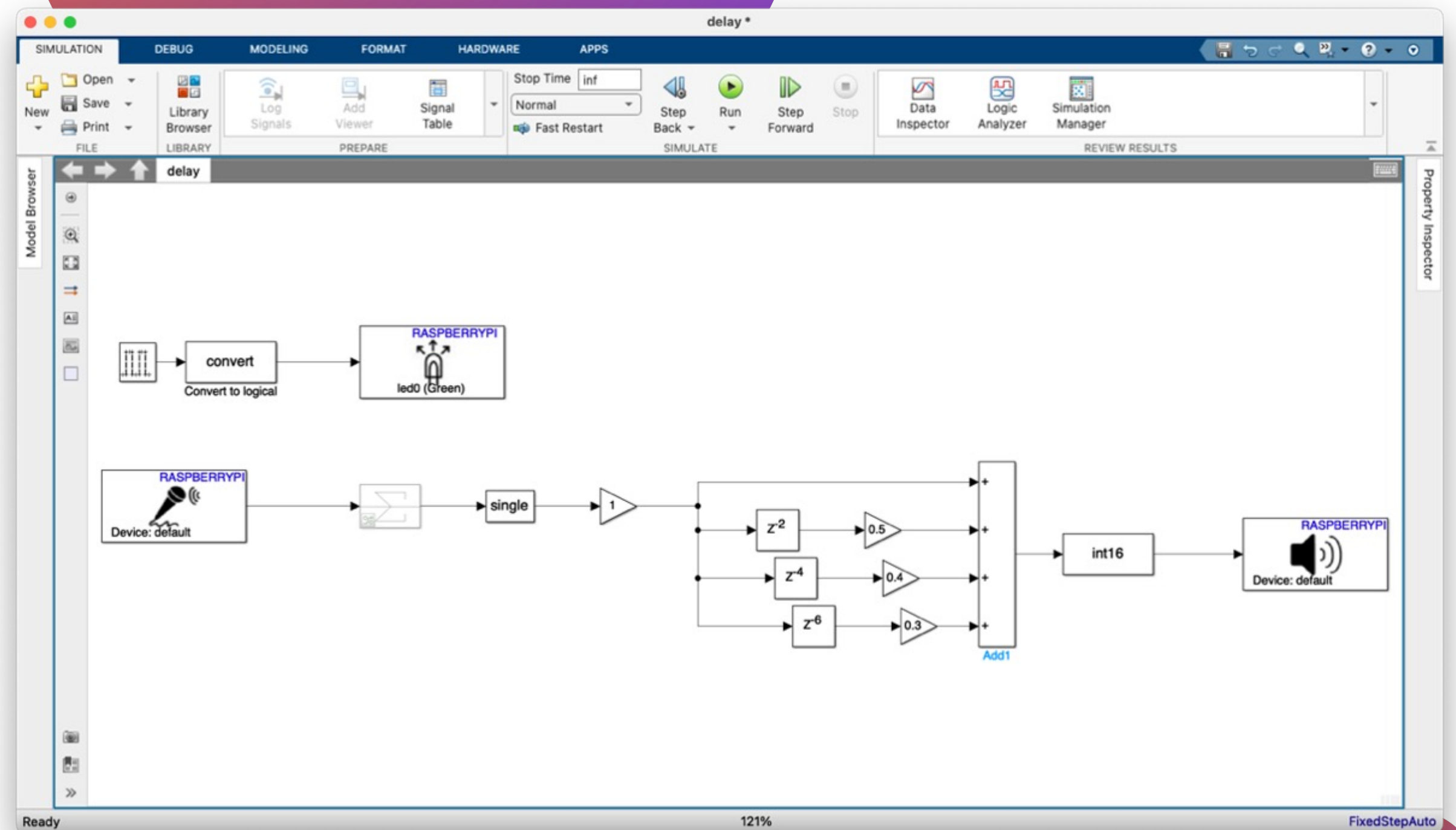


Apliquem un retràs de n mostres (per tant, un retràs temporal).

En funció del número de senyals i els temps tenim un estil de delay o altre.

Sistema lineal i invariant en el temps.

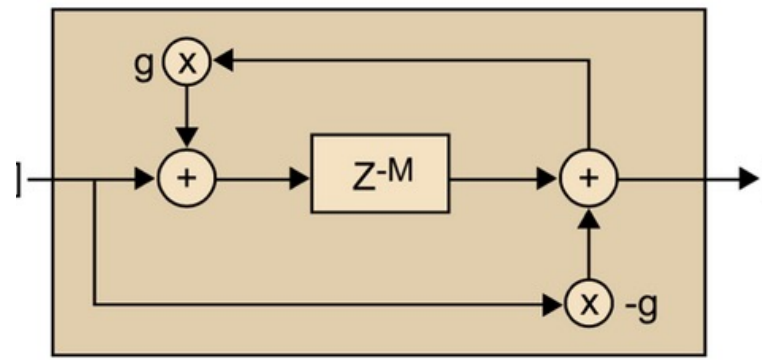
És un efecte modificador del temps.



Reverberació

Situació de la font sonora a l'espai

$$y[n] = -gx[n] + x[n - m] + gy[n - m]$$

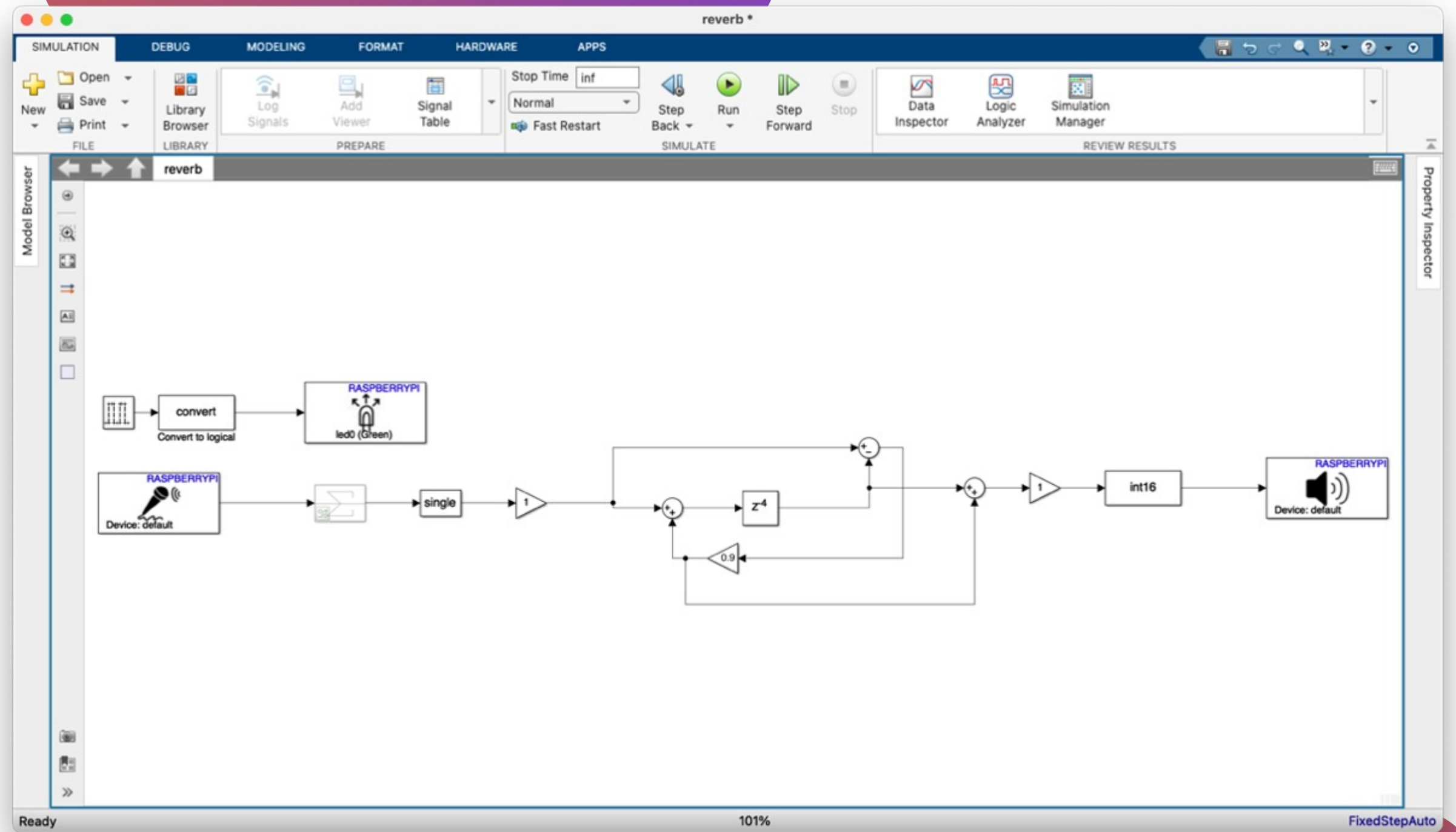


Generem versions retardades del senyal amb una atenuació determinada imitant les reflexions a l'espai.

Podem generar models més o menys complexes en funció del número de versions (reflexions) que tinguem.

Sistema lineal i invariant en el temps.

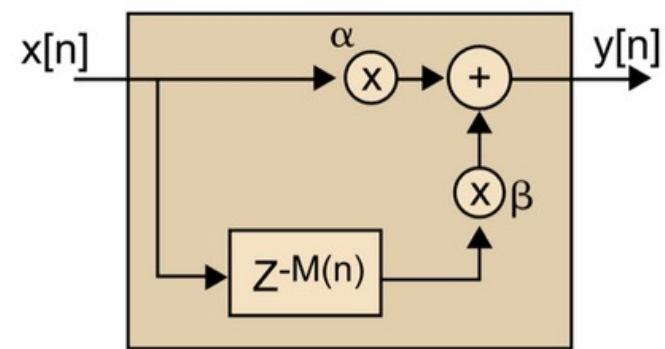
És un efecte modificador del temps.



Flanger

Retard governat per una ona sinusoidal

$$y[n] = x[n] + g[x[n - M(n)]]$$

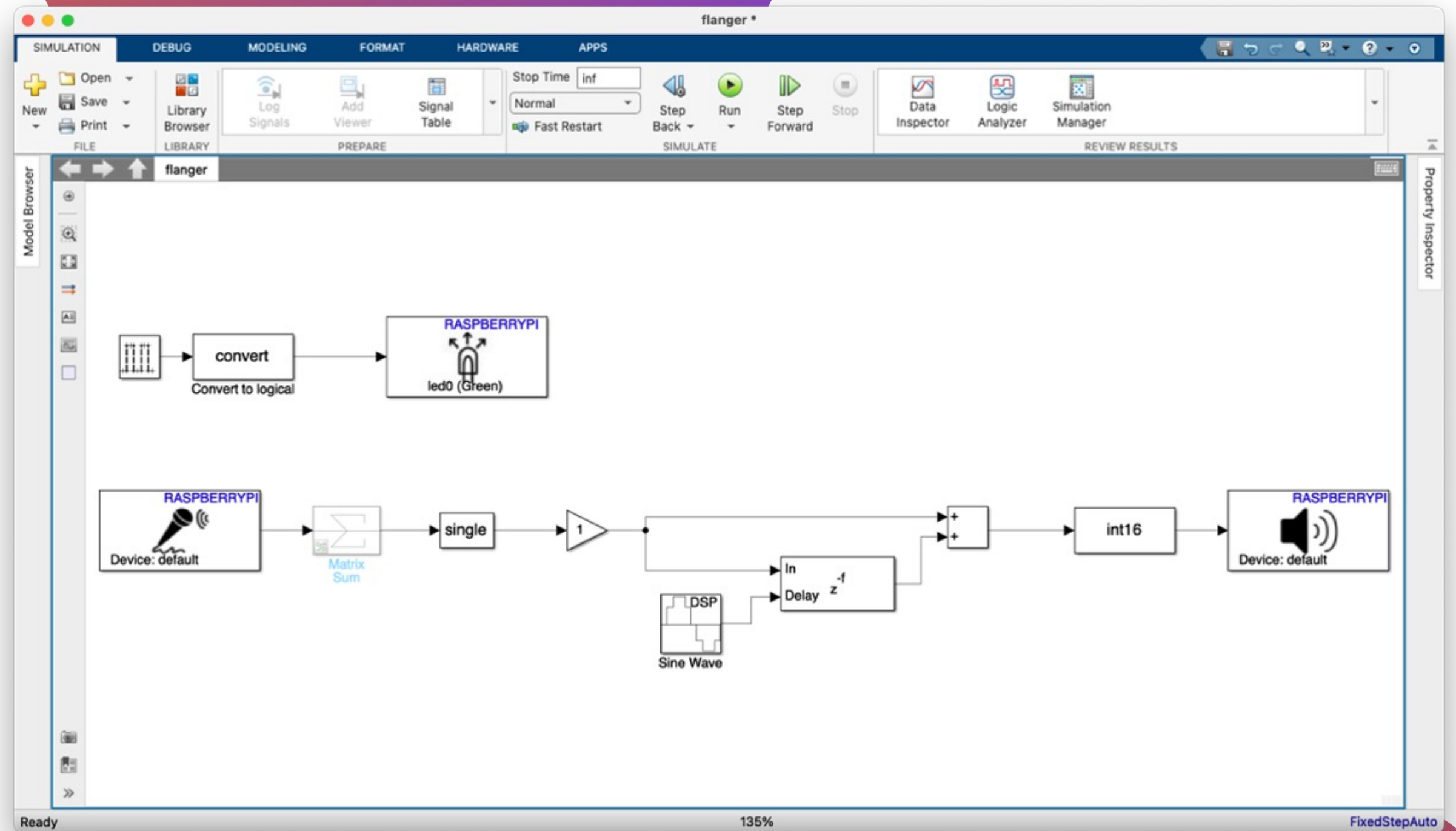


Generem una sola versió retardada però el temps de retard (i per tant, el número de mostres) serà variable en el temps.

Necessitem d'una ona sinusoidal (normalment entre 100 o 1kHz, és a dir, de baixa freqüència, LFO).

Sistema lineal i variant en el temps.

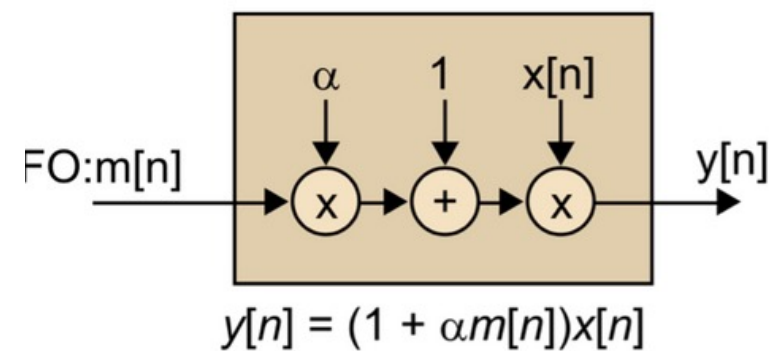
És un efecte modificador del contingut freqüencial.



Tremolo

Modificador de la dinàmica gràcies a un LFO

$$y[n] = x[n] \cdot M[n]$$

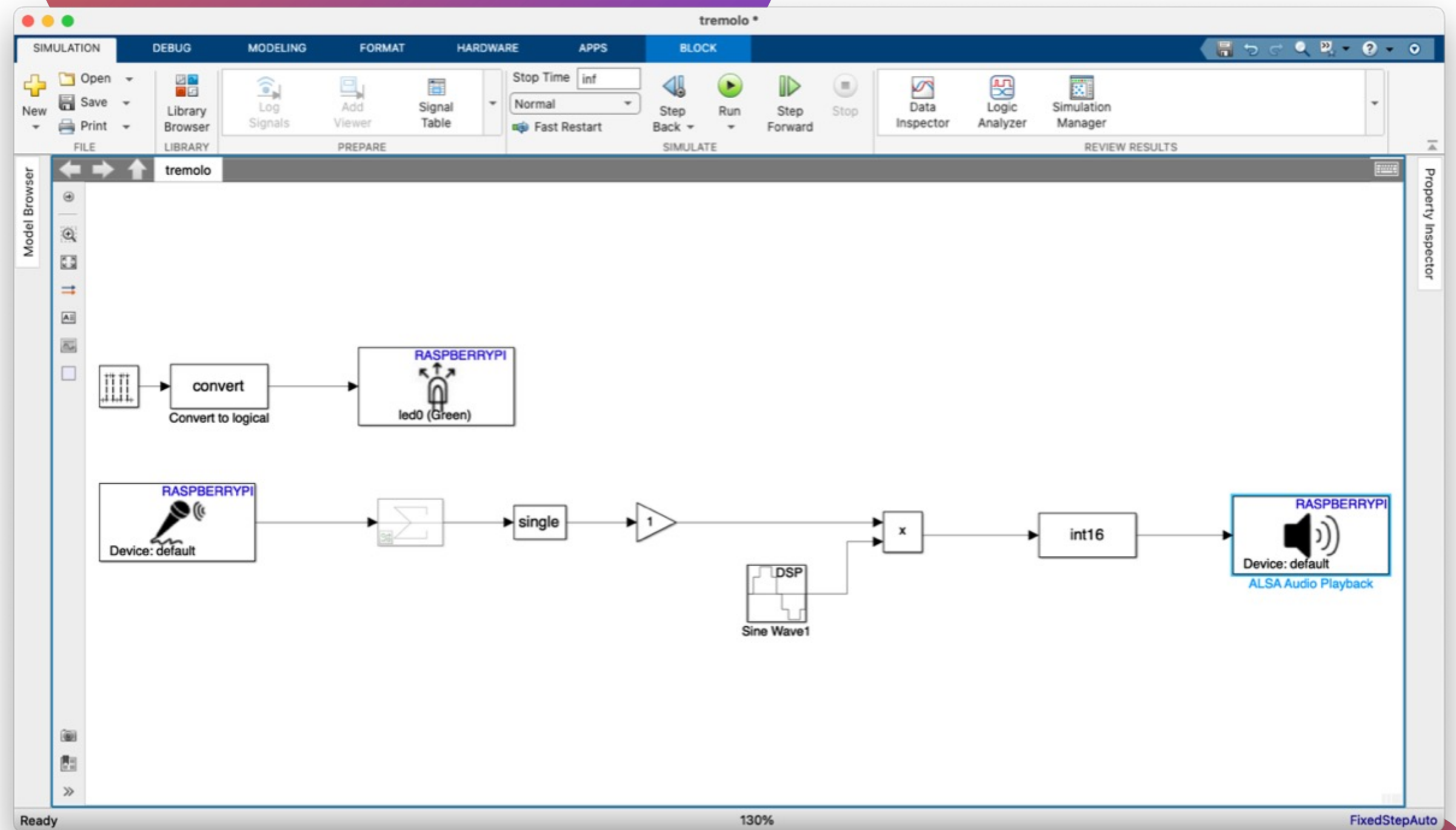


Amb un LFO, modifiquem l'amplitud del senyal original gràcies a multiplicar els dos senyals.

Necessitem una altra vegada d'un LFO.

Sistema lineal i variant en el temps.

És un efecte modificador de la dinàmica.



Equalitzador

Podem modificar el contingut freqüencial del senyal

Es construeixen amb base de filtres on podem decidir la freqüència central, l'ample d'actuació i el nivell d'atenuació.

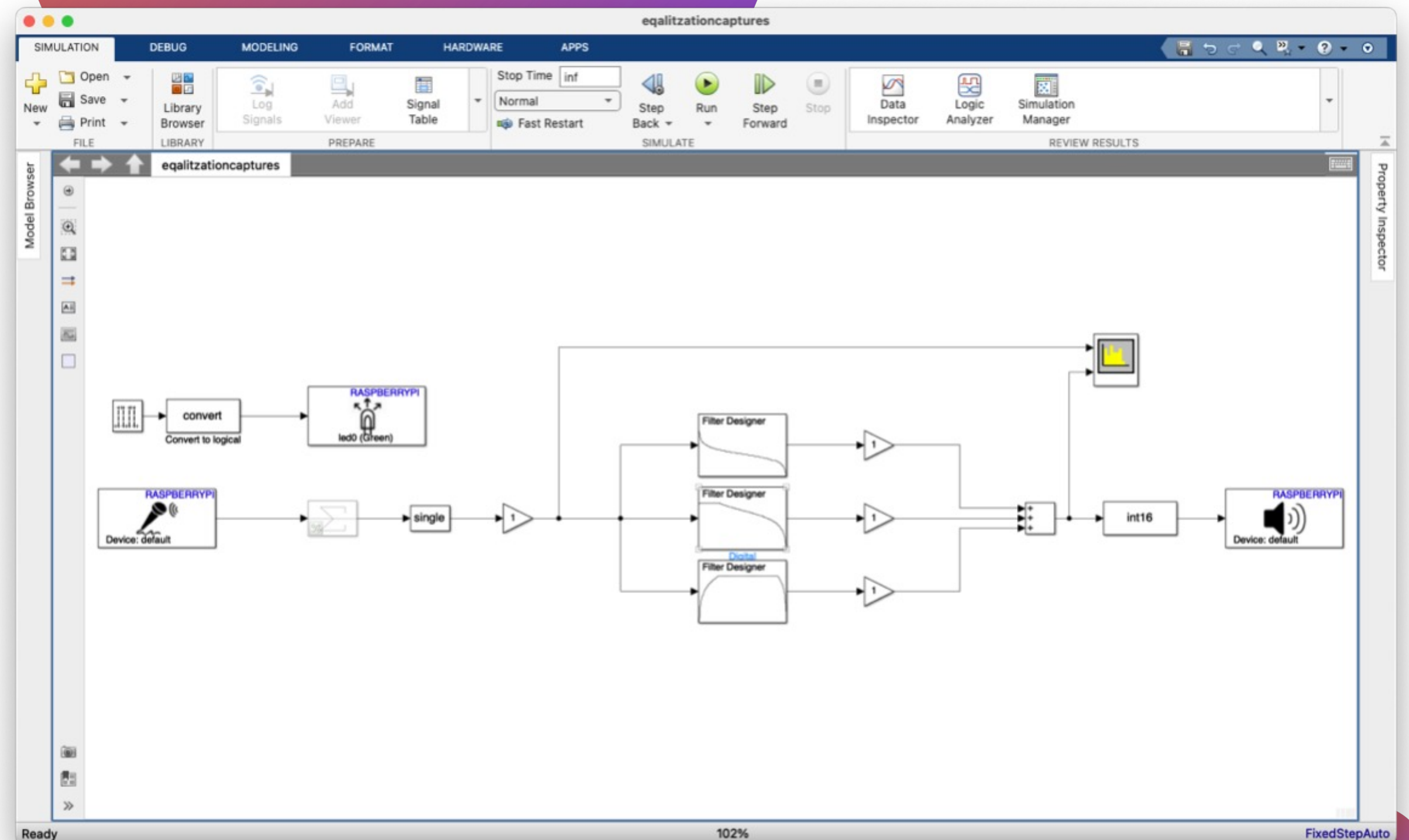
En funció dels paràmetres que puguem modificar tindrem un tipus d'equalitzador.

És l'únic efecte del que se n'ha construït dues versions diferents.

La primera versió utilitza l'eina Filter Digital Designer del Simulink.

El senyal es procesa **en paral·lel**

És un efecte modificador del contingut freqüencial.



Equalitzador

Podem modificar el contingut freqüencial del senyal

Es construeixen amb base de filtres on podem decidir la freqüència central, l'ample d'actuació i el nivell d'atenuació.

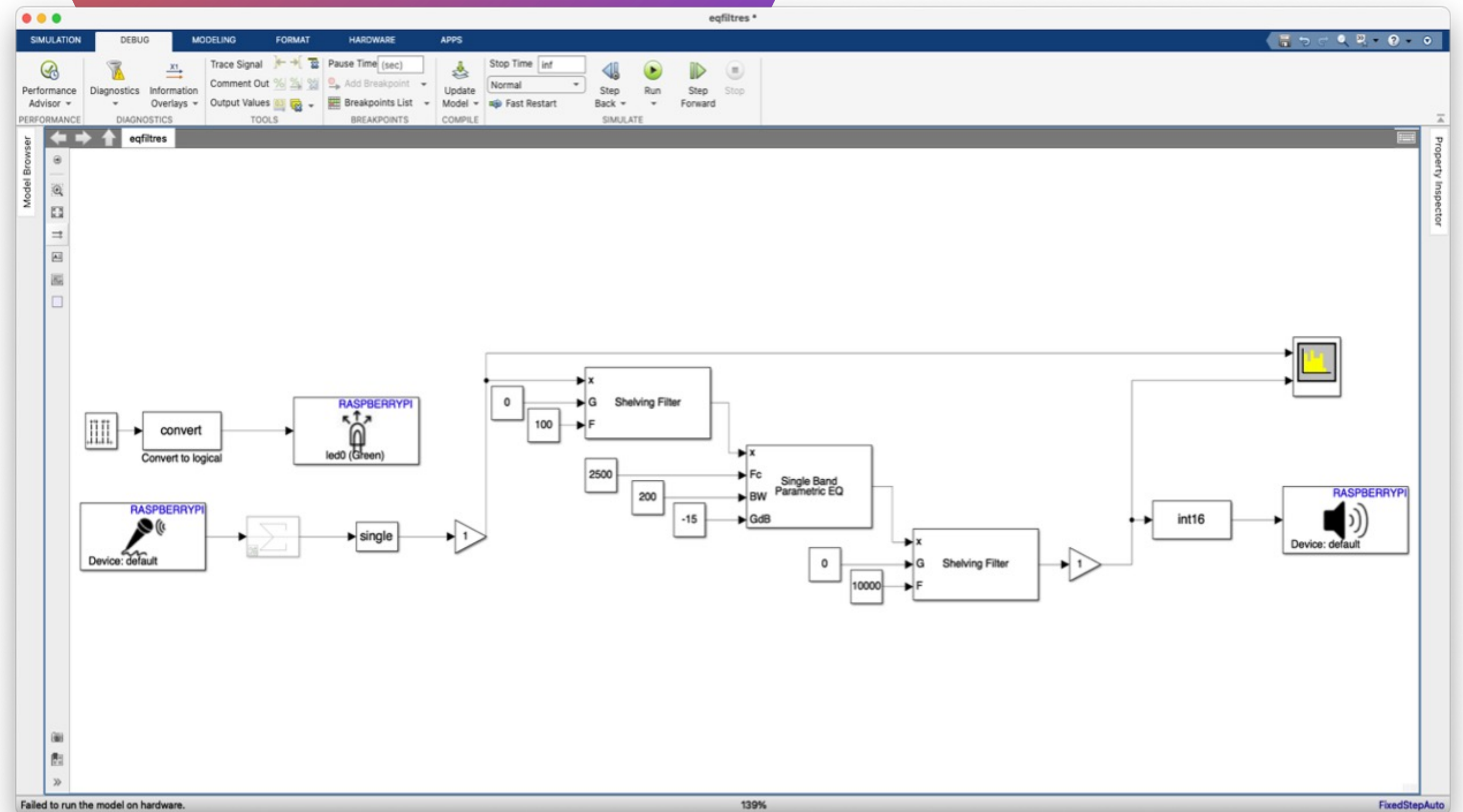
En funció dels paràmetres que puguem modificar tindrem un tipus d'equalitzador.

És l'únic efecte del que se n'ha construït dues versions diferents.

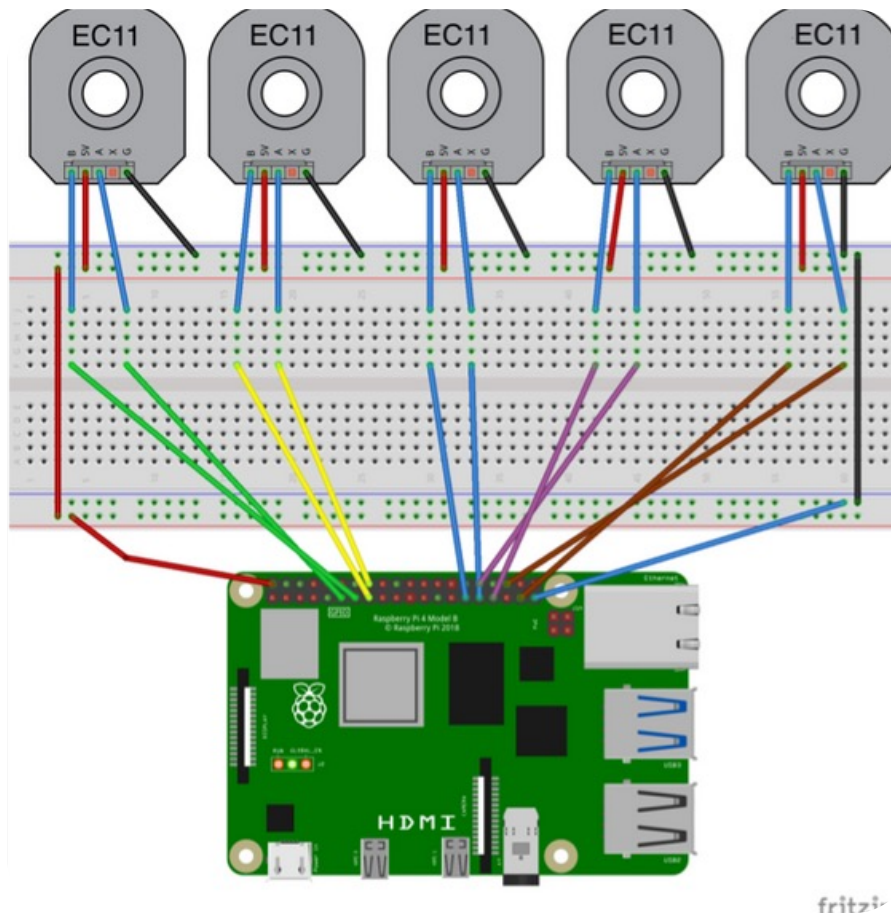
La segona versió utilitza el complement Audio Toolbox del Simulink on ja hi ha blocs d'equalització.

El senyal es procesa **en sèrie**

És un efecte modificador del contingut freqüencial.

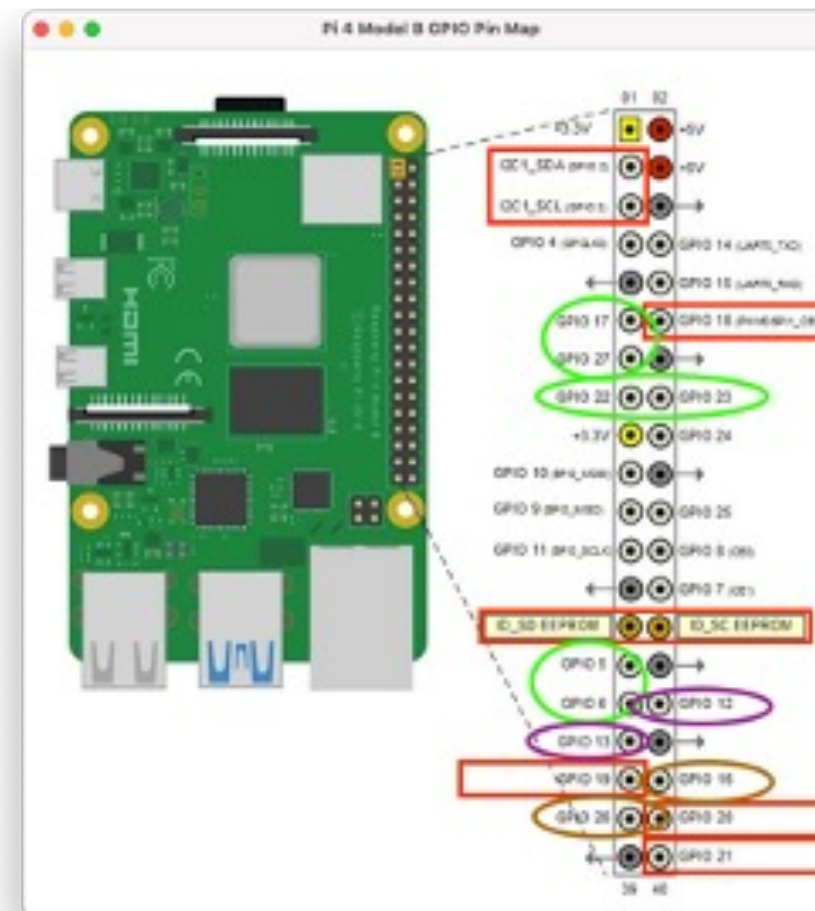


Disseny del hardware



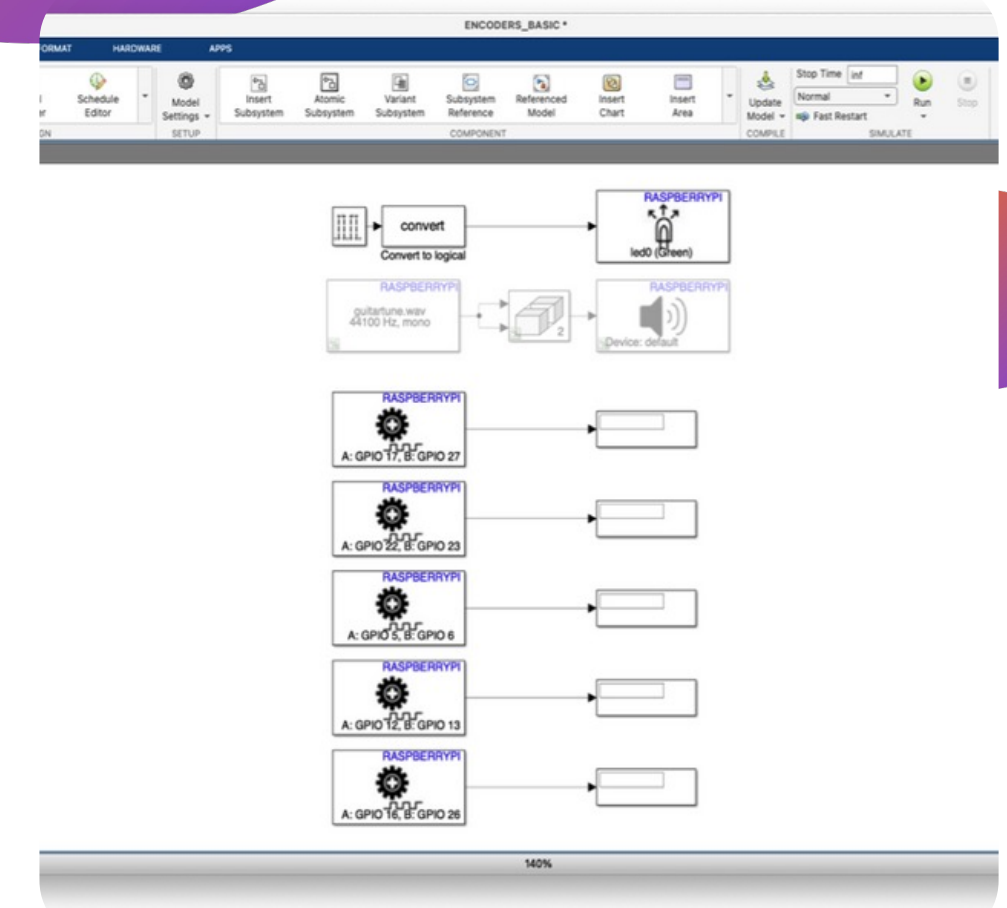
Plantejament inicial

Partint dels esquemes electrònics del RPi i dels potenciòmetres fem el prototipat



Assignació del GPIO

A aquestes alçades ja treballem amb una targeta d'àudio externa i per tant, hem d'escollir quins pins i reservar-ne alguns per la targeta



Modelat al Simulink

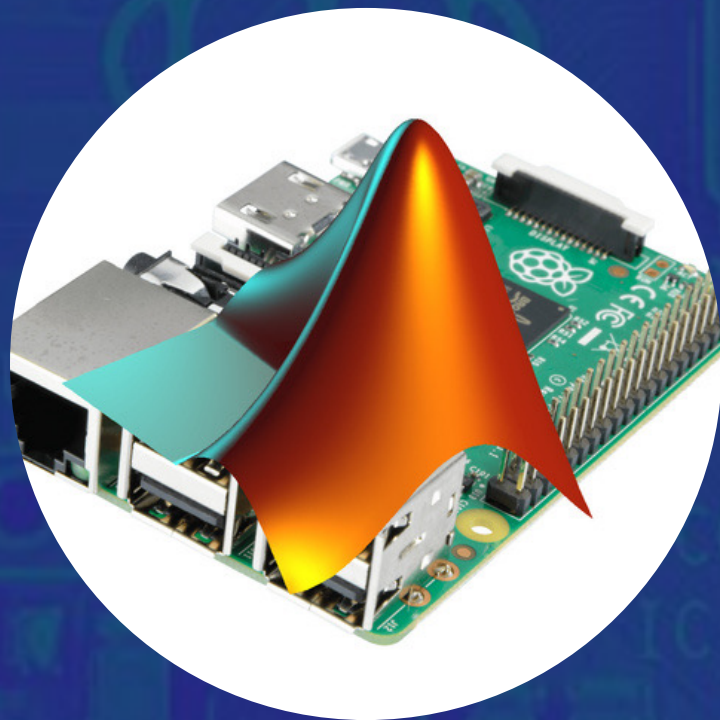
Abans de la implementació amb els efectes es genera un model de prova amb els pins escollits. Apareixen els problemes amb l'XCP

Problemes durant el desenvolupament

01

Compatibilitat amb el sistema operatiu

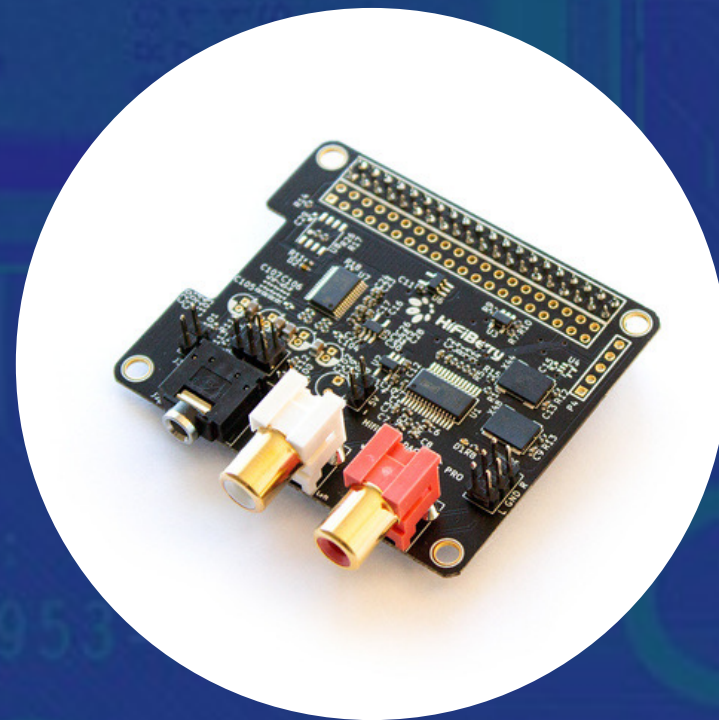
El complement per al RPi que ofereix MathWorks no era compatible amb la última versió del Raspberry Pi OS



02

Captació i reproducció d'àudio

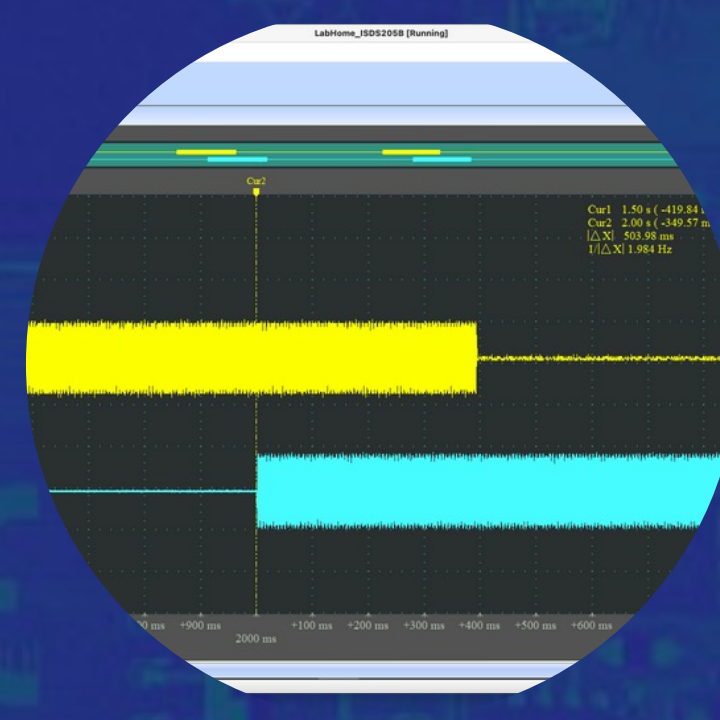
EL RPi no ofereix entrada d'àudio. Necessitem una targeta externa de qualitat superior a la plantejada inicialment



03

Temps real i latència

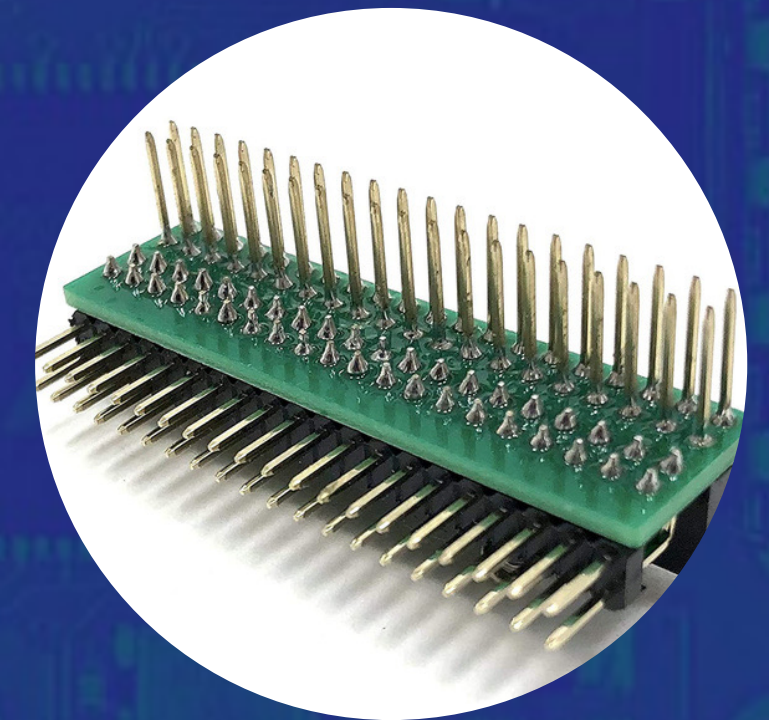
Després de moltes investigacions, arribem a la conclusió que el Simulink sempre treballa a uns 500 ms de latència. Insuficient per aprofitar el fenomen d'emascarament



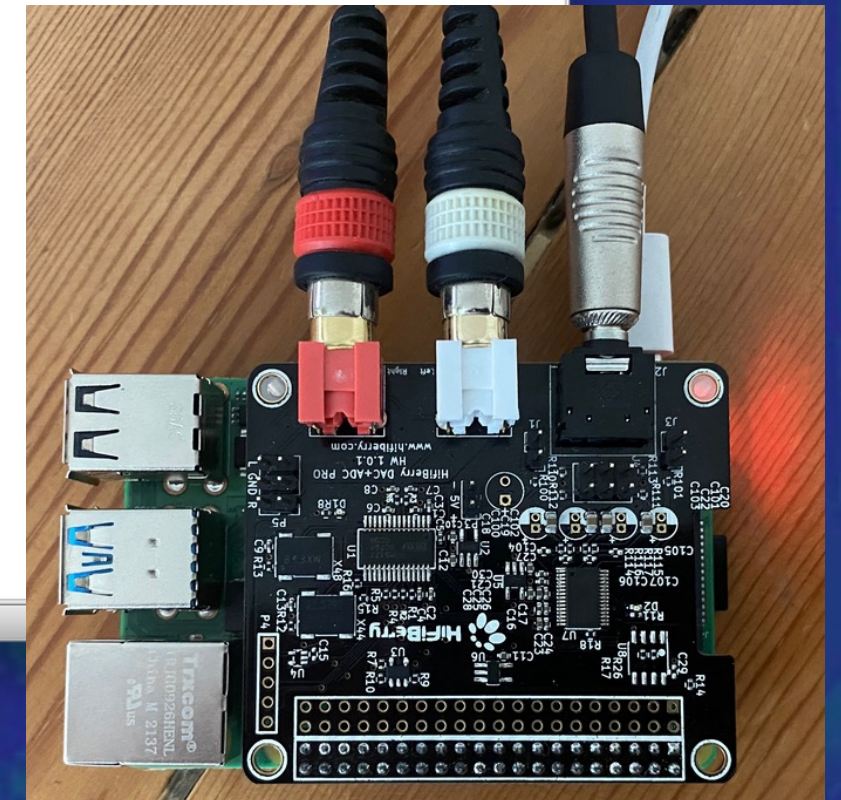
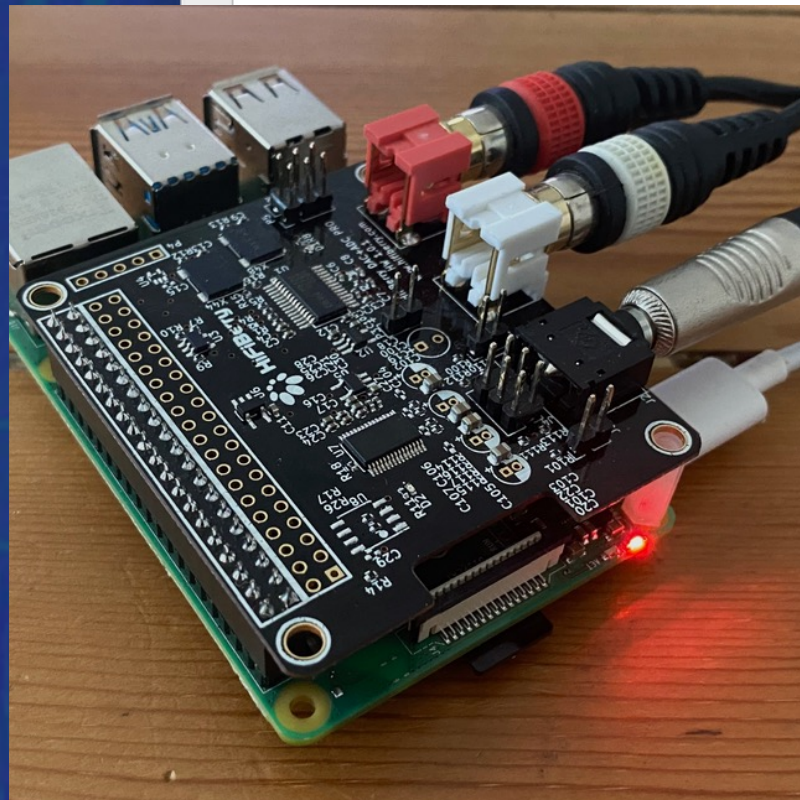
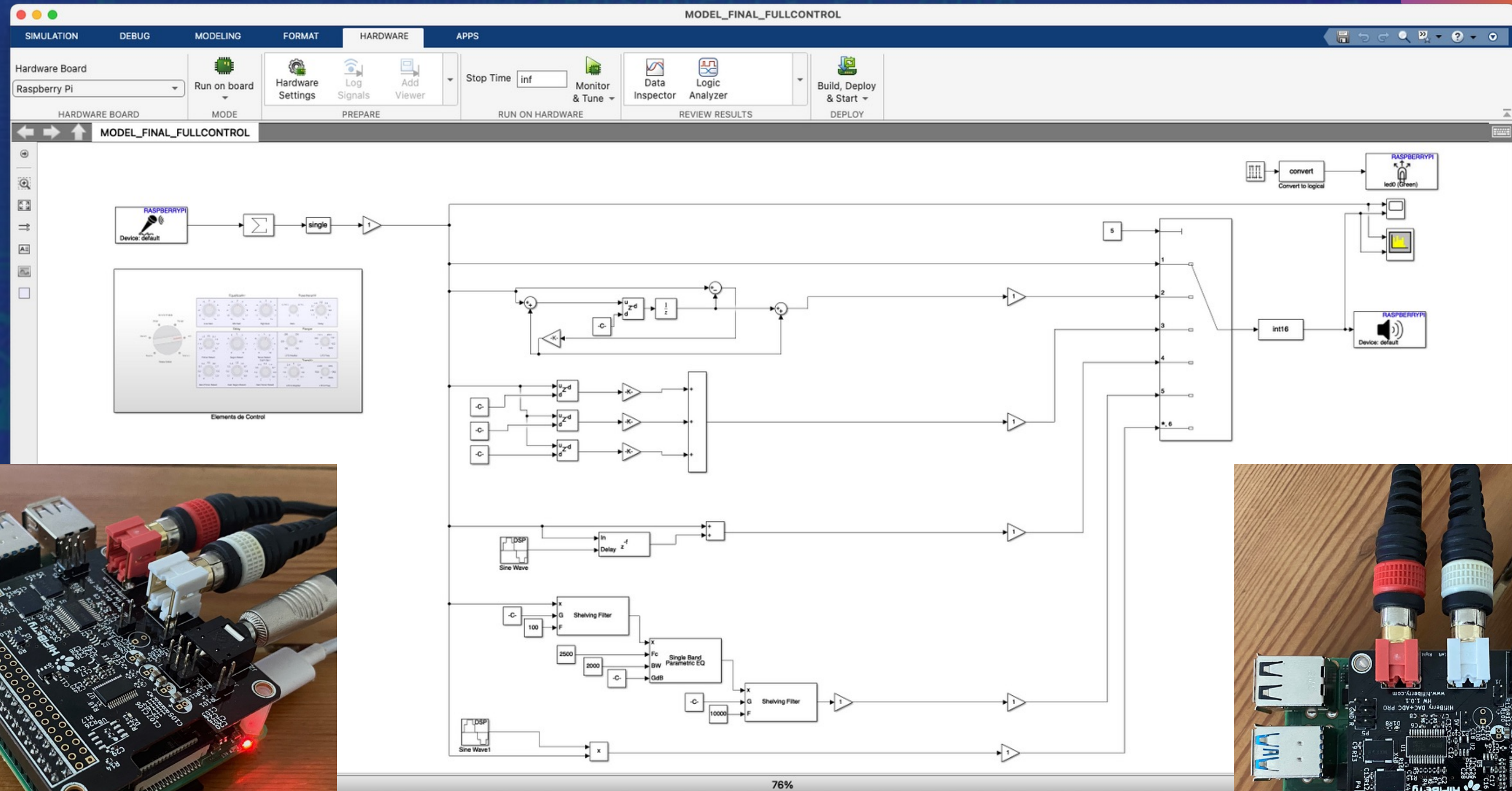
04

XCP i GPIO

El protocol XCPactua de coll d'ampolla i no permet el control de més d'un potenciòmetre via GPIO. Es decideix prescindir d'ells i programar un panell de control a l'ordinador de desenvolupament



Model final



Model final - Panell de control

The screenshot displays a software control panel titled "MODEL_FINAL_FULLCONTROL/Elements de Control" for a Raspberry Pi hardware board. The interface includes a top menu bar with options like SIMULATION, DEBUG, MODELING, FORMAT, HARDWARE, and APPS. Below the menu is a toolbar with icons for hardware settings, logging, and deployment. The main area is divided into several sections:

- Rotary Switch:** A central circular control with a red indicator, labeled with "Constant:Value", "Delay", "Flanger", "EQ", "Tremolo", "Bypass", and "Reverb".
- Equalitzador (Equalizer):** Three knobs for "Low Gain", "Mid Gain", and "High Gain", each with a scale from -15 to 15.
- Reverberació (Reverb):** Two knobs for "Gain" (scale 0 to 1.1) and "Delay" (scale 1 to 10).
- Delay:** Three knobs for "Primer Retard", "Segon Retard", and "Tercer Retard", each with a scale from 0 to 10.
- Flanger:** Two knobs for "LFO Amplitut" (scale 150 to 400) and "LFO Freq" (scale 1 to 5000).
- Tremolo:** Two knobs for "LFO 2 Amplitut" (scale 1 to 5) and "LFO 2 Freq" (scale 10 to 5000).

At the bottom of the window, the status bar shows "Ready", "174%", and "FixedStepAuto".

Vídeo de demostración