

# Hands-On n.3 – DSP

- A library of common DSP functions written in C using intrinsics
  - Source and object (.lib) files provided
  - Separate functions for operating on 8, 16 and 32-bit ints and 32-bit floats
- A collection of 61 algorithms including:
  - Basic maths: vector multiply, add, subtract, scale, shift, negate...
  - Statistics: root mean square, mean, standard deviation...
  - Fast maths: sine, cosine, square root...
  - Complex maths: conjugate, dot product, magnitude, multiply by real...
  - Filters: FIR, IIR, convolution, correlation..
  - Matrix algebra: addition, multiplication, scale...
  - Transforms: Fast Fourier, discrete cosine...
  - Controller: PID motor control, (Inverse)Park transform, (Inverse)Clarke transform...
  - Interpolation: linear and bilinear...
  - Support functions: type conversion, copy, fill...
- Provided free of charge by ARM



# CMSIS 2.0.0: DSP Software Library on Cortex-M3 M4 and M4F (1/2)

3

- **Basic Math Functions**

- Vector Absolute Value
- Vector Addition, Subtraction
- Vector Dot Product
- Vector Multiplication
- Vector Negate
- Vector Offset
- Vector Scale
- Vector Shift

- **Fast Math Functions**

- Cosine, Sine
- Square Root

- **Complex Math Functions**

- Complex Conjugate
- Complex Dot Product
- Complex Magnitude
- Complex Magnitude Squared
- Complex-by-Complex Multiplication
- Complex-by-Real Multiplication

- **Filtering Functions**

- Biquad Cascade IIR Filters Using Direct Form I Structure
- Biquad Cascade IIR Filters Using a Direct Form II Transposed Structure
- High Precision Q31 Biquad Cascade Filter
- Convolution
- Partial Convolution
- Correlation
- Finite Impulse Response (FIR) Decimator
- Finite Impulse Response (FIR) Filters
- Finite Impulse Response (FIR) Lattice Filters
- Finite Impulse Response (FIR) Sparse Filters
- Infinite Impulse Response (IIR) Lattice Filters
- Least Mean Square (LMS) Filters
- Normalized LMS Filters
- Finite Impulse Response (FIR) Interpolator

# CMSIS 2.0.0: DSP Software Library on Cortex-M3 M4 and M4F (2/2)

4

## • Matrix Functions

- Matrix Addition, Subtraction
- Matrix Initialization
- Matrix Inverse
- Matrix Multiplication
- Matrix Scale
- Matrix Transpose

## • Transform Functions

- Complex FFT Functions
- DCT Type IV Functions
- Real FFT Functions

## • Controller Functions (Motor Control)

- Sine Cosine
- PID Motor Control
- Vector Clarke/Inverse Clarke Transform
- Vector Park/Inverse Park Transform

## Statistics Functions

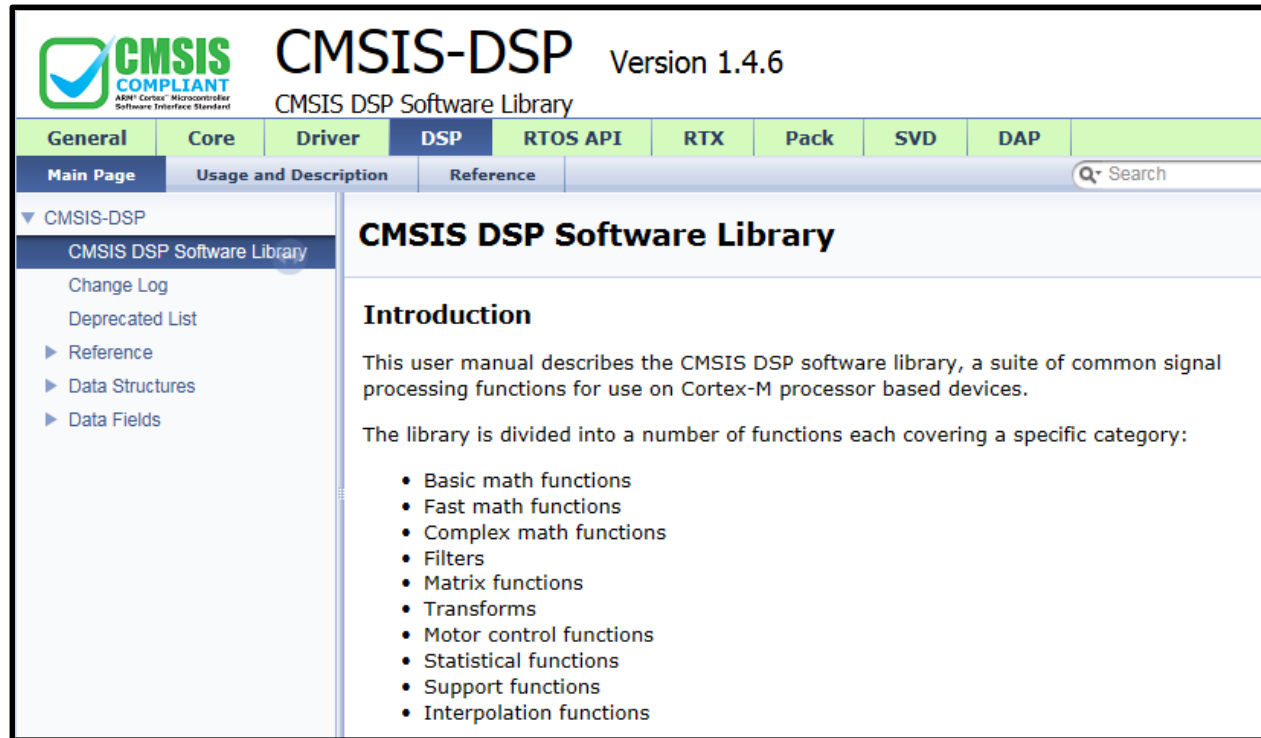
- Maximum
- Mean Minimum
- Power
- Root mean square (RMS)
- Standard deviation
- Variance

## Support Functions

- Convert 16-bit Integer value
- Convert 32-bit Integer value
- Convert 32-bit floating point value
- Convert 8-bit Integer value
- Vector Copy
- Vector Fill

## Interpolation Functions

- Linear Interpolation
- Bilinear Interpolation



**CMSIS DSP Software Library**

**ARM Cortex-M7 Processor Technical Reference Manual**

**Lab-in-a-Box for Digital Signal Processing (DSP) Course**

This Hands-On is composed of project

- **Audio Player**

The Audio Player exploits the STM32F7 audio peripheral to play (optionally filtered) audio traces

**Goals of this Hands-On are release to you a kit of functions**

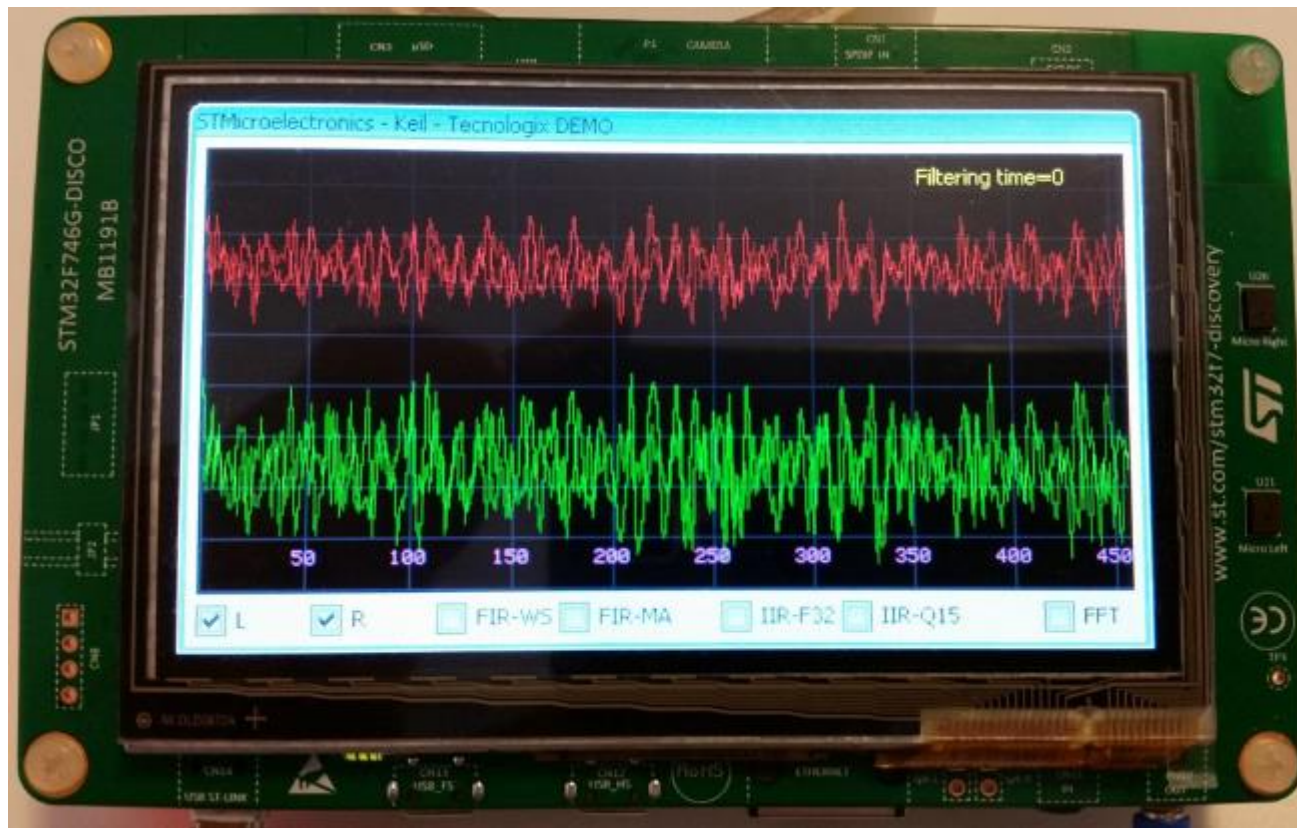
- **FFT** computation
- Show the better quality of a (long) convolution **FIR** filter w.r.t. **IIR** filters
- Show the better performance (time) of **IIR** filters w.r.t. a (long) convolution **FIR** filter
- Show how **IIR-Q15** filter dramatically increase the performance (**SIMD** instructions)
- Show how filters cut-off frequencies

- Plays audio traces previously sampled at **16KHz**
- Traces are transferred to audio peripheral from internal Flash memory through DMA
- A double buffer strategy is implemented
- The left and right channels are shown on the LCD screen
- The user can choose the filter to apply to the audio trace
- The filtering time is shown on the screen
- Optionally the Fourier Transform of the playing audio is shown on the screen at real time



# Hands-On n.3 – DSP, Audio Player

9



Connect your headphones  
to hear the audio  
generated.  
(Green Jack)

## Four targets are available

- Two tones
- Three tones
- Shower
- Jethro Tull

Each target enables a specific filter implementation, for each of these filter types

- FIR Windowed SincQ15 (FIR-WS)
- FIR Moving Average(FIR-MA)
- IIR BiquadCascade F32(IIR-F32)
- IIR BiquadCascade Q15(IIR-Q15)

- **FIR-WS** coefficients (**128**) have been computed using a Kaiser-Bessel filter designer available at <http://www.arc.id.au/FilterDesign.html>
- **FIR-MA** impulse response is always a rectangular pulse of ten points and unit area
- **IIR F32** and **Q15** coefficients have been computed using an IIR filter designer available at <http://www.earlevel.com/main/2013/10/13/biquad-calculator-v2/>
- Other tools can be used, e.g. [Scilab](#)

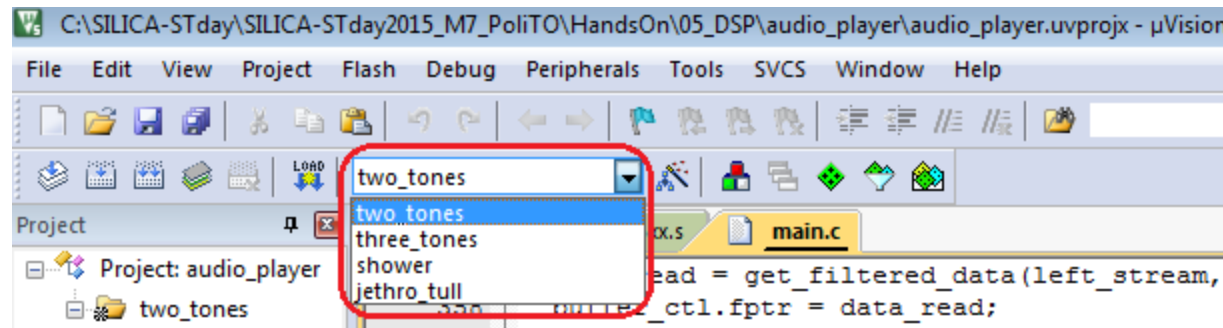
- The audio trace is composed of two superposed tones
  - A pure tone at 440Hz
  - A pure tone at 1000Hz
- All the filter types implements a low pass filter at a cut-off frequency of 500Hz

Open the example:

**C:\...\SILICA-STday2015\_M7\_PoliTO\HandsOn\05\_DSP\audio\_player**

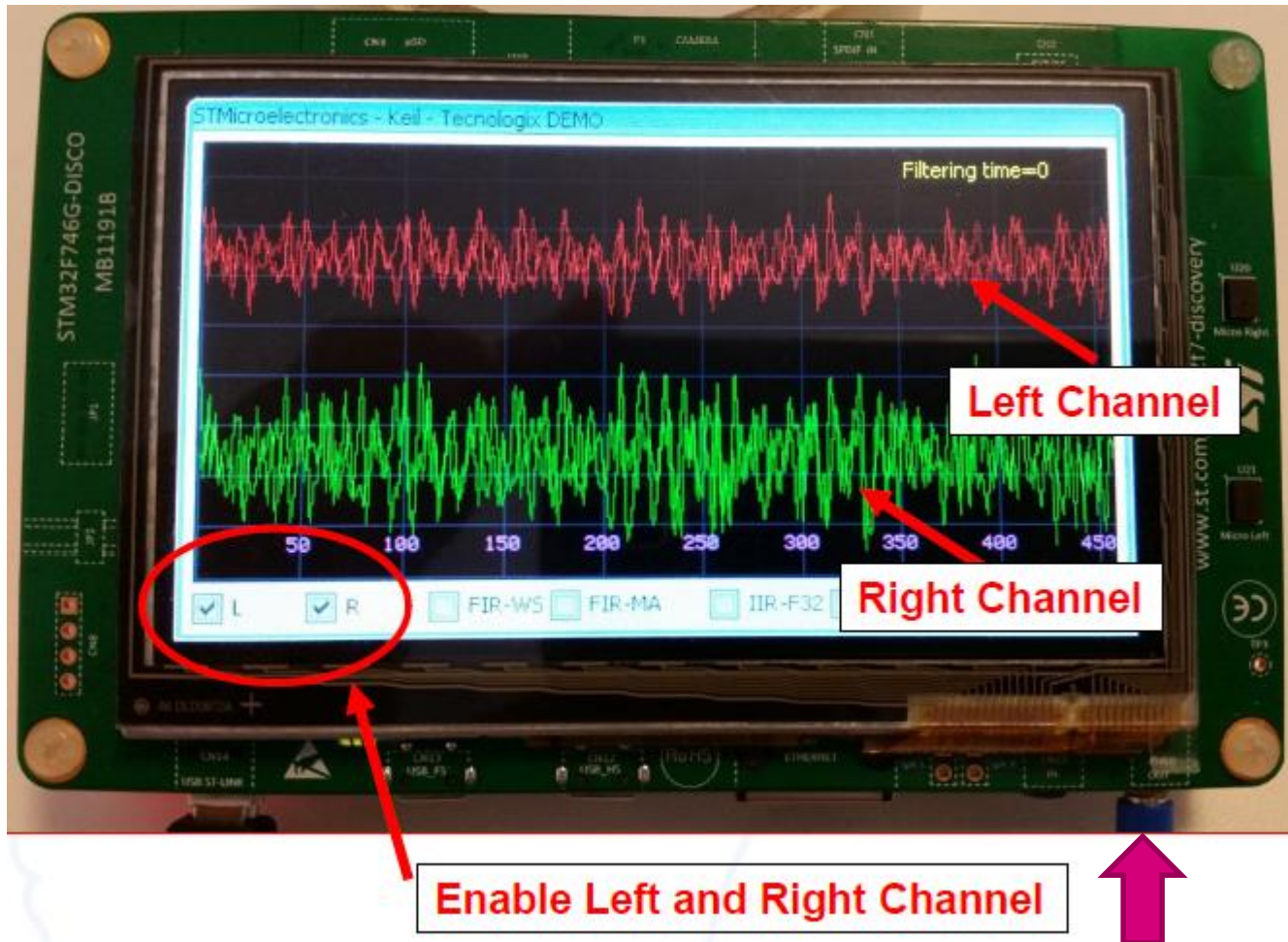
Select: **two\_tones**

**Compile and Run**



# Hands-On n.3 – DSP Two Tones

13

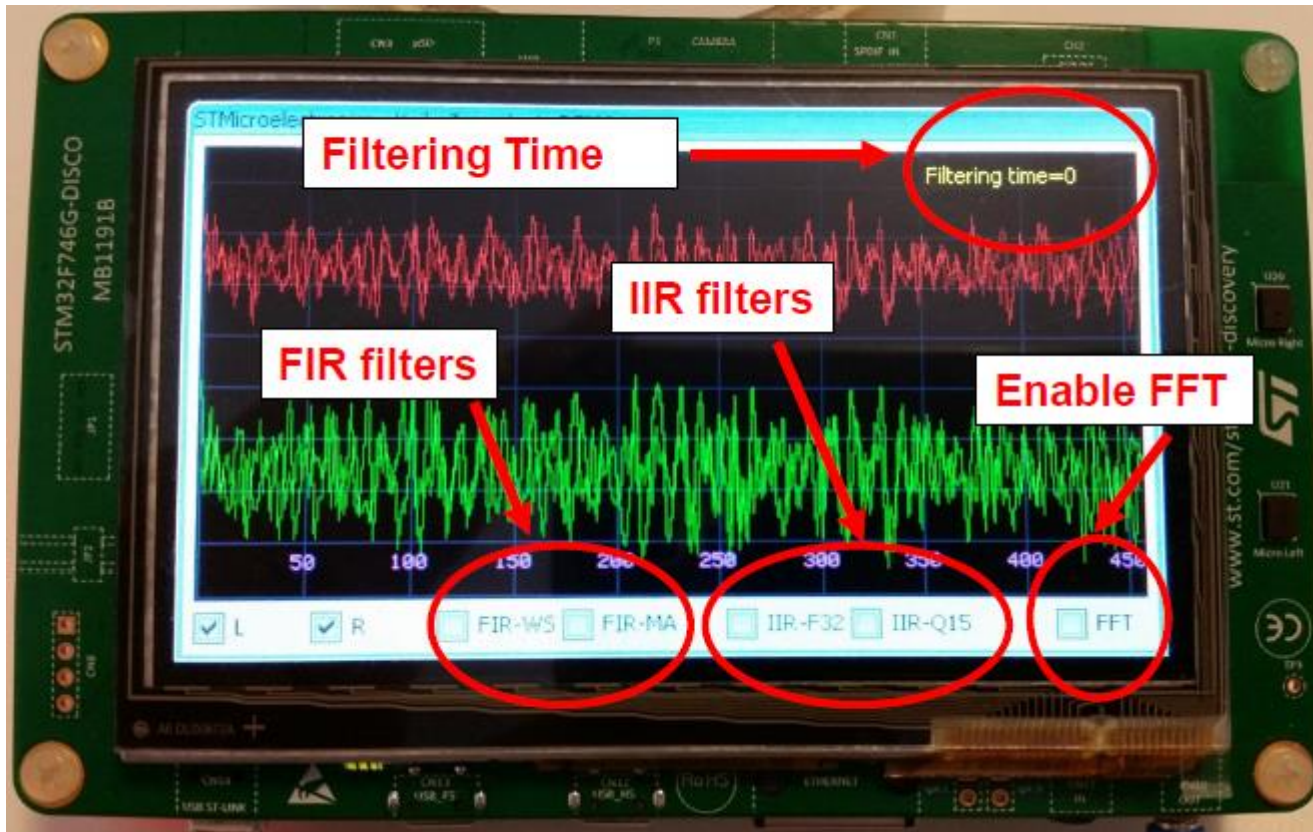


Connect your headphones to hear the audio generated.  
(Green Jack)



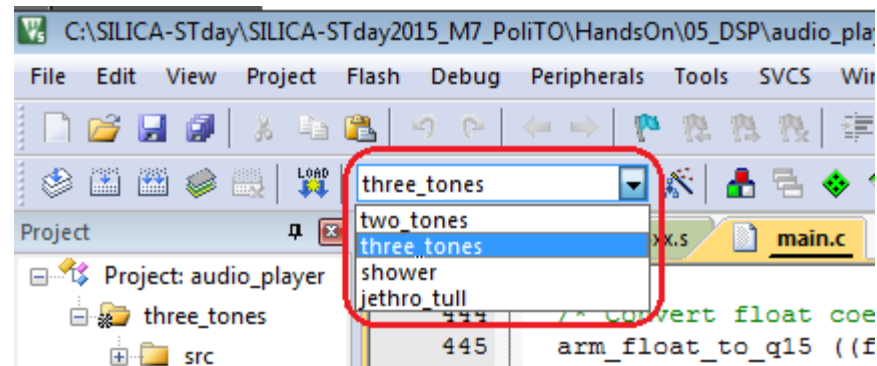
# Hands-On n.3 – DSP Two Tones

14



Connect your headphones  
to hear the audio  
generated.  
(Green Jack)

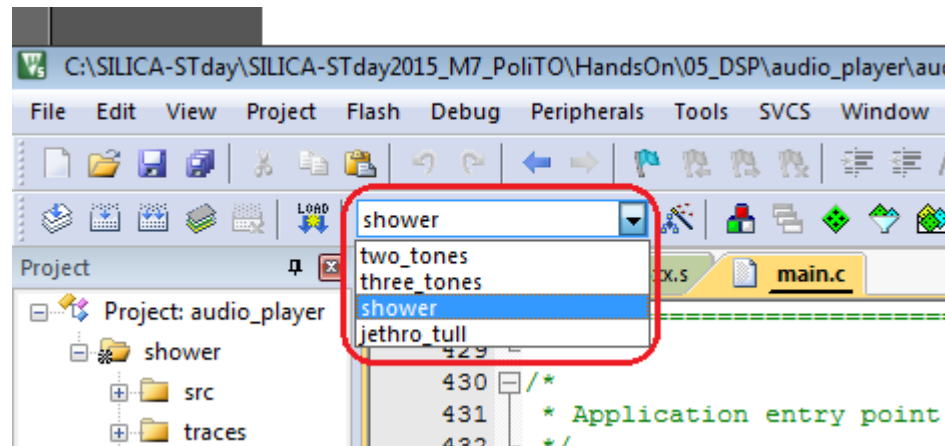
Select: **three\_tones**  
**Compile and Run**



- The audio trace is composed of three superposed tones
  - A pure tone at 440Hz
  - A pure tone at 1500Hz
  - A pure tone at 3500Hz
- The **FIR-WS** implements a band pass filter, with pass band within 1200Hz and 1700Hz. It will isolate the tone at 1500Hz
- The **IIR-F32** implements a high pass filter at a cut-off frequency of 3000Hz, isolating the tone at 3500Hz
- The **IIR-Q15** filter implements a low pass filter at a cut-off frequency of 500Hz, isolating the tone at 440Hz.

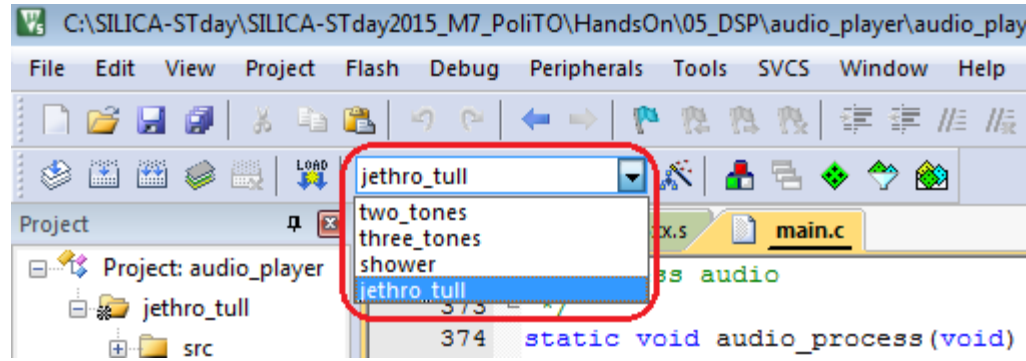


Select: **shower**  
**Compile and Run**



- The audio trace is composed of a voice of a man singing under the shower.  
The shower noise resembles a random noise source
- The **FIR-WS** implements a low pass filter at a cut-off frequency of **500Hz**
- The **FIR-MA** performs the moving average of ten points
- **IIR-F32** and **IIR-Q15** implements a high pass filter at a cut-off frequency of **3500Hz**

Select: **Jethro\_tull**  
**Compile and Run**



- Let's rock !



- The audio trace is an excerpt (the flute solo) of the song “Locomotive Breath” by Jethro Tull

- The **FIR-WS** filter implements a band pass filter with pass band within 1200Hz and 1700Hz, isolating the **FLUTE** from the other instruments (drums, guitars, bass)
- The **IIR-F32** implements a 500Hz low pass filter, isolating (mainly) the **BASS DRUM** from the other instruments.
- The **IIR-Q15** implements a 3500Hz high pass filter, isolating (mainly) the **CHARLESTON DRUM BEAT** from the other instruments

# Need more info ?

22

For more info contact:

**[enrico.marinoni@avnet.eu](mailto:enrico.marinoni@avnet.eu)**

(Digital FAE for STM - MCU, WireLess (IoT), MEMS, PLM, etc)

**[roberto.rossetti@avnet.eu](mailto:roberto.rossetti@avnet.eu)**

(B.D.M.)





# Thank you

[www.st.com/stm32](http://www.st.com/stm32)