Omer Ifrach (oi38)

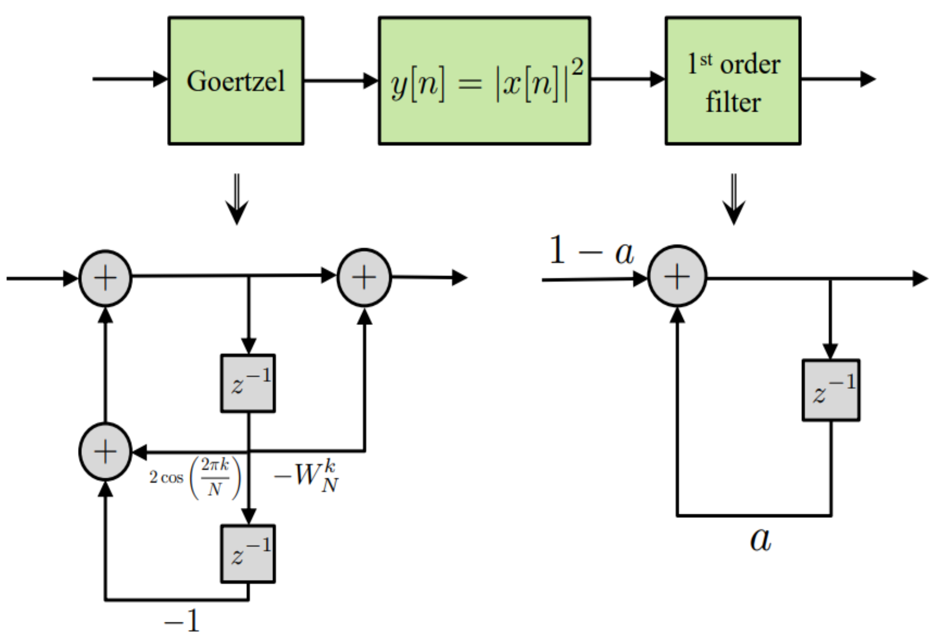
DSP Midterm – DTMF Decoder

The Goertzel Algorithm is derived from the Discrete Fourier Transform (DFT). It exploits the phase periodicity factor $e^{-\frac{j2\pi k}{N}}$ and could reduce computational complexity associated with the FFT in instances where only a few samples of the DFT are needed.

Dual Tone Multi Frequency (DTMF), also referred to as touch-tone, is the signal generated by telephone touch keys. A DTMF decoder recognizes the sequence of DTMF tones.

The task was to implement a DTMF decoder with a robust second harmonic tone detection.

The Decoder Block Diagram



$x[n] \rightarrow input$

$s[n] \rightarrow temp\ value$

$y[n] \rightarrow output$

$a \rightarrow alpha$

$n \rightarrow sample$

$N \rightarrow bin\ size$

$w = \frac{2\pi k}{N}$

$c_{f1} = 2 \cdot \cos(w)$

$c_{f2} = e^{-\frac{2\pi k}{N}j}$

$F_s = 8kHz$

Stage 1 - The Goertzel Algorithm

The first stage of this system could be expressed using the following two equations. Each with it's own transfer function:

$$\boldsymbol{s[n] = x[n] + (2 \cdot \cos(w) \times s[n-1]) - s[n-2]} \rightarrow Z(s[n]) = \frac{S(z)}{X(z)} = \frac{1}{(1 - 2 \times \cos(w) z^{-1} + z^{-2})}$$

$$\rightarrow \frac{1}{\left(1 - z^{-1}\left(\cos(w) + j\sin(w)\right)\right)\left(1 - z^{-1}\left(\cos(w) - j\sin(w)\right)\right)} \rightarrow \boldsymbol{\frac{1}{(1 - z^{-1}(e^{jw}))(1 - z^{-1}(e^{-jw}))}} \ldots \mathbf{(1)}$$

$$\boldsymbol{y[n] = s[n] + W_N^k \times s[n-1]} \rightarrow Z(y[n]) = \frac{Y(z)}{S(z)} = \boldsymbol{1 - z^{-1}(e^{-jw})} \ldots \mathbf{(2)}$$

To evaluate the transfer function of the Goertzel filter we must multiply both stages together:

$$\boldsymbol{H(z) = \frac{Y(z)}{X(z)} = \frac{1}{1 - z^{-1}(e^{jw})}} \ldots \mathbf{(3)}$$

From here, we can easily figure out the time domain using the inverse z-transform:

$$Z^{-1}(H[z]) = h(n) = \frac{y[n]}{x[n]} = \frac{y[n]}{y[n] - y[n-1]e^{jw}} \rightarrow \boldsymbol{y[n] = x[n] + y[n-1]e^{jw}} \ldots \mathbf{(4)}$$

We can now clearly see the resemblance to the discrete Fourier transform. Accounting for the recursive summation of $y[n-1]$ terms, and substituting for ω.

$$DFT \rightarrow \sum_{k=0}^{N-1} f[k] \times e^{-\frac{j2\pi k}{N}} \ldots \mathbf{(5)}$$

This is the reason the algorithm works. Proof based on – 4 & 5.

Stage 2 – Absolute Square

The output generated by the first stage, is now being processed through an intermediate quadratic block:

$$\boldsymbol{y[n] = |x[n]|^2}$$

The reasoning behind this operation is that the output signal from the Goertzel filter could take positive, negative or complex form. To compare the power coefficients of the sinusoidal frequencies – we need to make sure they are all real and positive. Hence, the system takes the output magnitude of each sample from the first stage, then squares it.

Stage 3 - First Order Filter

The final stage of the DTMF decoder is a first order, low-pass filter. This filter is implemented to *damp* the effect of the previous squaring block. In other words, this prevents the system from generating high frequency, added noise to the output signals.

The filter achieves this by keeping a 'weighted average' of output samples. Each output is multiplied by a constant scalar $\boldsymbol{a' = 1 - a}$ *where,* $\boldsymbol{0 < a < 1}$, then summed with all the previous outputs. Finally, the total value of the outputs is multiplied by the reciprocal of $\boldsymbol{a}$ (so that the system remains stable). This guarantees a gradual dampening of "older" signals. The closer to 1 the value of $\boldsymbol{a}$ – the more gradual the change in the output resultant.