

# An audio scrambler

Carlos B. Crespo

**Escuela Técnica Superior de Ingenieros de Telecomunicación.  
Universidad Politécnica de Madrid.  
January, 2000**

*"Big Brother is watching you"*

*George Orwell, 1984*

*"A digital illiterate is an illiterate"*

*Nicholas Negroponte*

## **Abstract**

This paper describes the design and implementation of a frequency inversion audio scrambler, with an introduction to some electrical engineering topics such as filter design, analog modulation, frequency inversion, LFSR's and Laplace transforms.

## **Introduction**

The scrambling of radio communications may be required by a customer for a multitude of reasons. For example, public agencies such as the police are monitored by persons interested in avoiding contact with such agencies, as well as reporters for leads on news stories and individuals for entertainment, so they might want to keep communication hidden from third parties. For these reasons, and many others, more and more of these parties are electing to scramble their communications, making them indecipherable to outside listeners.

## Voice Security Systems

Voice security systems entail modifying an original signal by some known coding algorithm such that the coded signal does not resemble the original signal. This coding algorithm is controlled by a specific code or "key", and different keys result in different coded signals. This coded signal is then transmitted. At the receiver, the coded signal is modified again by a decoding algorithm under the control of a specific key, which is related to the transmission key. The result is the recovered signal, which resembles the original signal.

In a correctly operating system, the Transmit Key and Receive Key are identical, and the decoding algorithm is the inverse of the coding algorithm. That is, whatever the coding algorithm does, the decoding algorithm undoes. If the Transmit and Receive Keys are different, then the decoding algorithm will not recover the original signal properly. Differences between different security systems appear primarily in the particular coding algorithm used. We will employ frequency inversion to scramble communications in this project.

### Overview of the scrambler design

The circuit consists of three logical parts: the sender, the channel and the receiver. The sender plugs an audio jack connection from the headphone output of a discman/walkman to the input of the circuit. After passing through some filters and amplifiers, the signal is randomly modulated to 6400 or 12800 Hz, while the frequency of the original signal gets inverted. A frequency selection bit decides which of the two possible values should be used. This bit is generated in a pseudo-random manner using a memory buffer as the random number seed. Then the scrambled signal is sent over a cable, along with some other information (encrypted selection bit and system clocks) to the receiver, who demodulates the signal on a similar fashion using the decrypted selection bit. Note that both sender and receiver must agree on which initial seed to use in order to guarantee correct decodification of audio signal. Finally, the signal is presented to a headphone connection.

### Some concepts on audio scrambling

#### The Laplace Transform

Let  $g(t)$  denote a nonperiodic deterministic signal expressed as some function of time.

Fourier transform of the signal  $g(t)$  is given by the integral  $G(f) = \int_{-\infty}^{\infty} g(t) e^{(-j2\pi ft)} dt$

where  $j = \sqrt{-1}$ , and the variable  $f$  denotes frequency. Given the Fourier transform  $G(f)$ , the original signal  $g(t)$  is recovered exactly using the formula for the inverse Fourier transform:

$$g(t) = \int_{-\infty}^{\infty} G(f) e^{(j2\pi ft)} dt$$

For the Fourier transform of a signal  $g(t)$  to exist, it is sufficient, but not necessary, that  $g(t)$  satisfies three conditions known collectively as Dirichlet's conditions:

1. The function  $g(t)$  is single-valued, with a finite number of maxima and minima in any finite time interval.
2. The function  $g(t)$  has a finite number of discontinuities in any finite time interval.
3. The function  $g(t)$  is absolute integrable.

We may safely ignore the question of the existence of the Fourier transform of a time function

$g(t)$  when it is an accurately specified description of a physically realizable signal. In other words, physical realizability is a sufficient condition for the existence of a Fourier transform. Indeed, we may go one step further and state that all energy signals, that is, signals  $g(t)$  for

$$\text{which } \int_{-\infty}^{\infty} |g(t)|^2 dt < \infty$$

are Fourier transformable.

## Continuous Spectrum

By using the Fourier transform operation, a pulse signal  $g(t)$  of finite energy is expressed as a continuous sum of exponential functions with frequencies in the interval  $-\infty$  to  $\infty$ . The amplitude of a component of frequency  $f$  is proportional to  $G(f)$ , where  $G(f)$  is the Fourier transform of  $g(t)$ . Specifically, at any frequency  $f$ , the exponential function  $e^{j2\pi ft}$  is weighted by the factor  $G(f) df$ , which is the contribution of  $G(f)$  in an infinitesimal interval  $df$  centered at the frequency  $f$ . Thus we may express the function  $g(t)$  in terms of the continuous sum of such infinitesimal components, as shown by the integral.

The Fourier transformation provides us with a tool to resolve a given signal  $g(t)$  into its complex exponential components occupying the entire frequency interval from  $-\infty$  to  $\infty$ .

In particular, the Fourier transform  $G(f)$  of the signal defines the frequency-domain representation of the signal in that it specifies relative amplitudes of the various frequency components of the signal. We may equivalently define the signal in terms of its time-domain representation by specifying the function  $g(t)$  at each instant of time  $t$ . The signal is uniquely defined by either representation.

In general, the Fourier transform  $G(f)$  is a complex function of frequency so that we may express it in the form

$G(f) = |G(f)| e^{j\theta(f)}$ , where  $|G(f)|$  is called the continuous amplitude spectrum of  $g(t)$ , and  $\theta(f)$  is called the continuous phase spectrum of  $g(t)$ . Here, the spectrum is referred to as a continuous spectrum because both the amplitude and phase of  $G(f)$  are defined for all frequencies.

For the special case of a real-valued function  $g(t)$ , we have

$$G(-f) = G^*(f)$$

Where the asterisk denotes complex conjugation. Therefore, it follows that if  $g(t)$  is a real-valued function of time  $t$ , then

$$|G(-f)| = |G(f)|$$

and

$$\theta(-f) = -\theta(f)$$

Accordingly, we may make the following statements on the spectrum of a real-valued signal:

1. The amplitude spectrum of the signal is an even function of the frequency; that is, the amplitude spectrum is symmetric about the vertical axis.
2. The phase spectrum of the signal is an odd function of the frequency; that is, the phase spectrum is antisymmetric about the vertical axis.

These two statements are summed up by saying that the spectrum of a real-valued signal exhibits conjugate symmetry.

The properties of the Fourier transform clearly show that the time-domain and frequency-domain descriptions of a signal are inversely related. In particular, we may make the following important statements:

1. If the time-domain description of a signal is changed, the frequency-domain description of the signal is changed in an inverse manner, and vice versa. This inverse relationship prevents arbitrary specifications of a signal in both domains. In other words, we may specify an arbitrary function of time or an arbitrary spectrum, but we cannot specify both of them together.
2. If a signal is strictly limited in frequency, the time-domain description of the signal will trail on indefinitely, even though its amplitude may assume a progressively smaller value. We say a signal is strictly limited in frequency or strictly band limited if its Fourier transform is exactly zero outside a finite band of frequencies. The sync pulse is an example of a strictly band-limited signal, asymptotically limited in time, which confirms the opening statement we made for a strictly band-limited signal. In an inverse manner, if a signal is strictly limited in time (i.e., the signal is exactly zero outside a finite time interval), then the spectrum of the signal is infinite in extent, even though the amplitude spectrum may assume a progressively smaller value. This behavior is exemplified by both the rectangular pulse and the triangular pulse. Accordingly, we may state that a signal cannot be strictly limited in both time and frequency.

## **Bandwidth**

The bandwidth of a signal provides a measure of the extent of significant spectral content of the signal for positive frequencies. When the signal is strictly band limited, the bandwidth is well defined. For example, the sync pulse has a bandwidth equal to  $W$  when, however, the

signal is not strictly band limited, which is generally the case, we encounter difficulty in defining the band-width of the signal. The difficulty arises because the meaning of "significant" attached to the spectral content of the signal is mathematically imprecise. Consequently, there is no universally accepted definition of bandwidth.

We may formally define the rms bandwidth of a low-pass signal  $g(t)$  with Fourier transform  $G(f)$  as follows:

$$W_{rms} = \sqrt{\frac{\int_{-\infty}^{\infty} f^2 |G(f)|^2 df}{\int_{-\infty}^{\infty} |G(f)|^2 df}}$$

## Modulating signals

The purpose of a communication system is to deliver a message signal from an information source in recognizable form to a user destination. With the source and the user being physically separated from each other. To do this, the transmitter modifies the message signal into a form suitable for transmission over the channel. This modification is achieved by means of a process known as modulation, which involves varying some parameter of a carrier wave in accordance with the message signal. The receiver recreates the original message signal from a degraded version of the transmitted signal after propagation through the channel. This recreation is accomplished by using a process known as demodulation, which is the reverse of the modulation process used in the transmitter. However, owing to the unavoidable presence of noise and distortion in the received signal, we find that the receiver cannot recreate the original message signal exactly. The resulting degradation in overall system performance is influenced by the type of modulation scheme used. Specifically, we find that some modulation schemes are less sensitive to the effects of noise and distortion than others.

We may classify the modulation process into continuous-wave modulation and pulse modulation. In continuous-wave (CW) modulation, a sinusoidal wave is used as the carrier. When the amplitude of the carrier is varied in accordance with the message signal, we have amplitude modulation (AM), and when the angle of the carrier is varied, we have angle modulation. The latter form of CW modulation may be further subdivided into frequency modulation (FM) and phase modulation (PM), in which the instantaneous frequency and phase of the carrier, respectively, are varied in accordance with the message signal. In pulse modulation, on the other hand, the carrier consists of a periodic sequence of rectangular pulses. Pulse modulation can itself be of an analog or digital type. In analog pulse modulation, the amplitude, duration, or position of a pulse is varied in accordance with sample values of the message signal. In such a case, we speak of pulse-amplitude modulation (PAM), pulse-duration modulation (PDM), and pulse-position modulation (PPM). The standard digital form of pulse modulation is known as pulse-code modulation (PCM) that has no CW counterpart. PCM starts out essentially as PAM, but with an important modification: The amplitude of each modulated pulse (i.e., sample of the original message signal) is quantized or rounded off to the nearest value in a prescribed set of discrete

amplitude levels and then coded into a corresponding sequence of binary symbols. The binary symbols 0 and 1 are themselves represented by pulse signals that are suitably shaped for transmission over the channel. In any event, as a result of the quantization process, some information is always lost and the original message signal cannot therefore be reconstructed exactly. However, provided that the number of quantizing (discrete amplitude) levels is large enough, the distortion produced by the quantization process is not discernible to the human ear in the case of a speech signal or the human eye in the case of a two-dimensional image. Among all the different modulation schemes, pulse-code modulation has emerged as the preferred method of modulation for the transmission of analog message signals for the following reasons:

- Robustness in noisy environment by regenerating the transmitted signal at regular intervals.
- Flexible operation.
- Integration of diverse sources of information into a common format.
- Security of information in its transmission from source to destination.

In introducing the idea of modulation, we stressed its importance as a process that ensures the transmission of a message signal over a prescribed channel. There is another important benefit, namely, multiplexing, that results from the use of modulation. Multiplexing is the process of combining several message signals for their simultaneous transmission over the same channel. Two commonly used methods of multiplexing are as follows:

- Frequency division multiplexing (FDM) in which CW modulation is used to translate each message signal to reside in a specific frequency slot inside the pass-band of the channel by assigning it a distinct carrier frequency; at the receiver, a bank of filters is used to separate the different modulated signals and prepare them individually for demodulation.
- Time-division multiplexing (TDM), in which pulse modulation is used to position samples of the different message signals in non-overlapping time slots.

Thus, in FDM the message signals overlap with each other in time, raising the possibility of cross-talk due to nonlinearity of the channel. On the other hand, in TDM the message signals exploit the full passband of the channel, but on a time-shared basis.

## Modulation using a train of pulses

We will be using a train of 50% duty cycle bipolar squared pulses to simulate DBL modulation in our circuit.

Their spectrum is  $P(\omega) = 2\pi \left( \sum_{k=-\infty}^{\infty} a_k \delta\left(\omega - \frac{k 2\pi}{T_0}\right) \right)$  where  $a_k$  are the coefficients of each signal period, which can be derived from  $a_k = \frac{\int x(t) e^{(-jk\omega_0 t)} dt}{T_0}$ . For our  $P(\omega)$  this yields

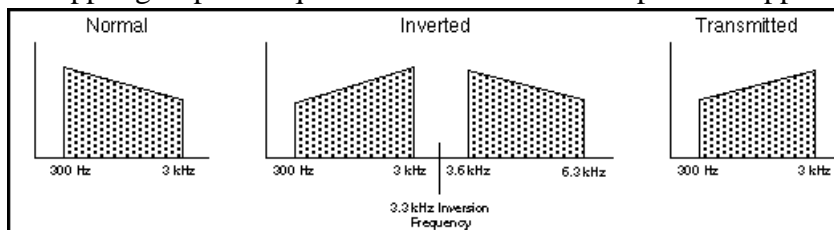
$$a_k = \frac{2 \sin\left(\frac{k \pi}{2}\right)}{k \pi}, \text{ so } a_k = \left[0, \frac{2}{\pi}, 0, \frac{2}{3\pi}, 0, \dots\right]. \text{ Since } y(t) = x(t) p(t), \text{ we have its equivalent in}$$

the frequency domain:  $Y(w) = \frac{X(w)}{2\pi} * P(w)$

## **Frequency inversion**

### Inversion Scrambling Background

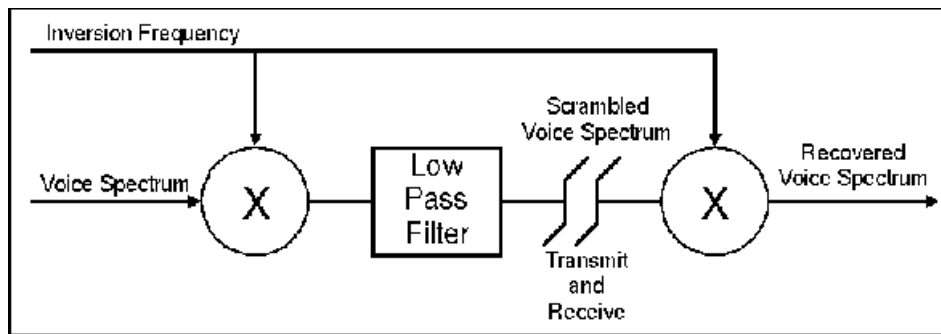
We will employ an analog voice scrambling technique known as frequency inversion. Inversion takes the existing frequency spectrum and inverts it. The standard voice spectrum is shown in the figure at the left. The human voice spectrum, transmitted over radio, has a frequency spectrum ranging from about 300 to 3000 Hz. The majority of the "power" in a voice is at the lower frequencies. The power decreases as the frequency increases, thus the downward slope of the spectrum in Figure 1. When inversion is done, the spectrum in the middle of the figure appears. Note now that the original voice spectrum is inverted in shape. The frequencies have moved with this inversion. The frequencies that were at 300 Hz are now at 3000 Hz. The original spectrum is present in the inverted spectrum but shifted up to 3600 Hz. After inversion takes place, a filter is applied to the inverted spectrum that removes the upper group of frequencies. The transmitted spectrum appears to the right in Figure 1.



**Figure 1**

### ***Inversion Implementation***

To invert a signal, some processing of the audio signal must take place. Figure 2 shows the process in block diagram form. Figure 2 shows the voice spectrum entering a multiplier function. The voice spectrum is multiplied with the inversion frequency. After the upper spectrum is filtered off, the scrambled signal is transmitted. The receiving radio receives the scrambled signal and inverts it. If the two scramblers are operating at the same inversion frequency, de-scrambling process will be performed correctly.



**Figure 2**

The inversion frequency also has an effect on the location of the inverted frequencies. A graphic of the frequency spectrum, with a 3.3 kHz inversion frequency, is shown in Figure 1 before and after inversion. Notice in the second chart the two spectrums that are pictured. Frequencies at both the sum of the original frequency and the inversion frequency as well as the difference between the inversion frequency and the original frequency result. Thus in Figure 1, the frequency components that were at 500 Hz in the normal spectrum are now mapped into the inverted spectrum at 2800 Hz and 3800 Hz. As long as the transmitting and receiving radios are at the same inversion frequency, and in sync, the audio can be encoded and decoded with good quality. This results because inversion is a balanced process.

***Hopping Inversion Frequency***

A higher level of security is created by pseudo-randomly changing the inversion frequency. This includes changing the direction of the inversion frequency step as well as the rate at which it changes. A diagram of this type of inversion is shown in Figure 3. To do this requires that both the sending and receiving radios change inversion frequency simultaneously. We could do this by sending an initial packet of data that tells the receiving scrambler where to start and what algorithm to use to control which frequency is switched to. To help maintain sync for long transmissions, we could also send brief update packets at predetermined intervals.

**Rolling Code Inversion**

Rolling code inversion can be viewed as a constantly changing inversion rate. While this is not exactly the case, the inversion frequency is changing rapidly enough to consider it constantly changing. The frequency moves from one inversion frequency to the next adjacent one. It continues in an upward inversion frequency direction until it reaches the upper limit. At this time, the inversion frequency starts back down until it reaches its lower limit. This creates a sawtooth type waveform if the inversion frequency is plotted over time as shown in Figure 4.

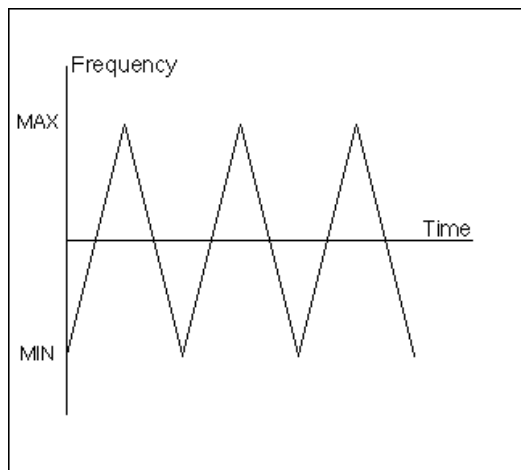


Figure 4

## [-] Implementing the audio scrambler

### [-] Filtering and modulation of the audio signal

#### [-] Line adaptor

We will be connecting the headphone output of a walkman to the audio-in jack of our circuit (see circuit schematics). Therefore a line adaptor is needed in order to provide adequate input impedance, as well as a highpass filter to get rid of any DC signal.

$Z_{in}$  must be around 300 ohms, and  $Z_{in} = \frac{1}{j\omega C} + R_{eq}$ , where  $R_{eq} = R1 // R2$ . We will be using 1  $\mu F$  capacitors so  $Z_c$  will be approximately 100 $\Omega$ , and, therefore,  $R_{eq} < 200$ . For

$$R_{eq} = 100, \frac{R1 R2}{R1 + R2} = 100 \Rightarrow (R2 = 1k\Omega) R1 = 110\Omega.$$

Since the low cutoff frequency must be 20 Hz, it follows that  $2 \pi \nu = \frac{1}{R_{eq} C} = 40 \pi$ , hence

$$R_{eq} C = \frac{1}{40 \pi}, \text{ and since } R_{eq} = 100 \Omega, C = \frac{1}{4 \pi 10^4} \mu F = 8 \mu F.$$

In short,  **$R1 = 100\Omega$ ,  $R2 = 1k\Omega$ ,  $C = 10 \mu F$ .**

#### [-] Sallen-Key Low-Pass filter

As seen in the circuit schematics, we can derive the following two node equations:

$$1) \frac{V_{in} - V1}{R1} + (V0 - V1) j \omega C = \frac{V1 - V0}{R2}$$

$$2) (V0 - 0) j \omega C = \frac{V1 - V0}{R2}$$

Manipulating this expressions:

$$V_0 j \omega C = \frac{V_1}{R_2} - \frac{V_0}{R_2} \Rightarrow V_0 \left( j \omega C + \frac{1}{R_2} \right) = \frac{V_1}{R_2}$$

$$\frac{V_{in}}{R_1} - \frac{V_1}{R_1} + V_0 j \omega C - V_1 j \omega C = \frac{V_1}{R_2} - \frac{V_0}{R_2}$$

$$\frac{V_0}{V_{in}} = \frac{1}{(j \omega R_2 C + 1) \left( 1 + j \omega R_1 C + \frac{R_1}{R_2} \right) - \frac{R_1}{R_2} - j \omega R_1 C} =$$

$$= \frac{1}{j \omega (R_1 C + R_2 C) - \omega^2 R_1 R_2 C^2 + 1}, \text{ and substituting } s=j \omega, \text{ we have}$$

$\frac{1}{s(R_1 C + R_2 C) + s^2 R_1 R_2 C^2 + 1}$ . We want to have a 40 dB fall, so that means we need a double pole. The discriminant of the second grade equation must be zero, therefore  $(R_1 C + R_2 C)^2 - 4 R_1 R_2 C^2 = 0$  which lets  $R_1 + R_2 = 2 \sqrt{R_1 R_2}$ , which has a trivial solution by making  $R_1=R_2=R$ . If this is done we have the following frequency response:

$$\frac{V_0}{V_{in}} = \frac{1}{1 + s 2 R C + s^2 R^2 C^2}. \text{ The canonical form of the low-pass Sallen Key filter is}$$

$$\text{given by } \frac{V_o}{V_{in}} = \frac{A_{vm}}{\frac{s^2}{\omega_0^2} + \frac{s}{\omega_0 Q} + 1}. \text{ Identifying terms: } 2RC = \frac{1}{\omega_0 Q}, Q = \frac{1}{2}, \omega_0 = \frac{1}{RC}. \text{ As we}$$

can see,  $Q = \frac{1}{2}$  follows from the 40 dB restriction. Subsequently we can calculate the

band of the filter as  $\frac{2}{RC}$ .

$$|H(s)| = \sqrt{H(s) H(-s)} \mid (s=j\omega) = \frac{1}{1 + R^2 C^2 \omega^2}.$$

$$\text{whc} = \omega / |H(j\omega)| = \frac{1}{\sqrt{2}} A_{vm}. \text{ Since } A_{vm} = 1, \text{ it follows that } \frac{1}{\sqrt{2}} = \frac{1}{1 + R^2 C^2 \omega^2} \text{ and}$$

$$\text{whc} = \frac{\sqrt{\sqrt{2} - 1}}{RC}.$$

### Sallen-Key High-Pass filter

As seen in the circuit schematics, we can derive the following two node equations:

$$1) (V_{in} - V_1) j \omega C + \frac{V_0 - V_1}{R_2} = (V_1 - V_0) j \omega C$$

$$2) (V_1 - V_0) j \omega C = \frac{V_0}{R_1}$$

Manipulating this expressions:

$$V_{in} j \omega C - V_0 \left( -\frac{1}{R_2} + \left( \frac{1}{R_1} + j \omega C \right) \left( \frac{1}{j \omega R_2 C} + 2 \right) - j \omega C \right) = 0$$

$$\frac{V_0}{V_{in}} = \frac{1}{1 + \frac{2}{j \omega R_1 C} - \frac{1}{\omega^2 R_1 R_2 C^2}}$$

We want to have a 40 dB fall, so that means we need a double pole. The discriminant of the second grade equation must be zero, therefore  $\frac{V_0}{V_{in}} = \frac{s^2 R_1 R_2 C^2}{s^2 R_1 R_2 C^2 + 2 s R_2 C + 1}$ , so

$4 R_2 C^2 (R_2 - R_1) = 0$  which lets  $R_1=R_2=R$ . If this is done we have the following frequency response:

$$\frac{V_0}{V_{in}} = \frac{1}{1 + \frac{2}{s R C} + \frac{1}{s^2 R^2 C^2}}. \text{ The canonical form of the high-pass Sallen Key filter is}$$

$$\text{given by } \frac{V_0}{V_{in}} = \frac{A_{vm}}{\frac{\omega_0^2}{s^2} + \frac{\omega_0}{s Q} + 1}. \text{ Identifying terms: } Q = \frac{1}{2}, \omega_0 = \frac{1}{R C}. \text{ As we can see, } Q = \frac{1}{2}$$

follows from the 40 dB restriction. Subsequently we can calculate the band of the filter

$$\text{as } \frac{2}{R C}.$$

$$|H(s)| = \sqrt{H(s) H(-s)} \big|_{(s=j\omega)} = \frac{1}{\sqrt{\left(1 - \frac{1}{R^2 C^2 \omega^2}\right)^2 + \left(\frac{2}{\omega R C}\right)^2}}$$

$$\omega_{lc} = \omega / |H(j\omega)| = \frac{1}{\sqrt{2}} A_{vm}. \text{ Since } A_{vm} = 1, \text{ it follows that}$$

$$\frac{1}{\sqrt{2}} = \frac{1}{\sqrt{\left(1 - \frac{1}{R^2 C^2 \omega^2}\right)^2 + \left(\frac{2}{\omega R C}\right)^2}} \text{ and}$$

$$\omega_{lc} = \frac{\sqrt{\sqrt{2} + 1}}{R C}.$$

### **Band limitation of the signal**

Our input signal must be first preprocessed before modulating it. We can do so by passband filtering the signal left after the line adapter.

A passband filter might be implemented with a low-pass Sallen-Key filter cascaded with a high-pass one. We've already devoted some time to calculate the necessary relation between R,C and  $\omega$ . So let's just substitute our design values:

· For the low-pass filter  $f_{hc} < \min(f_1, f_2) < 6400$  Hz. The reason why this frequency must be less than the minimum of the two working frequencies is because if it were greater it would overlap with it's negative counterpart. We will choose a value of 4000 Hz for  $f_{hc}$ .

$$RC = \frac{\sqrt{\sqrt{2}-1}}{2\pi \cdot 4000} = 2.56 \cdot 10^{-5}. \text{ If } C=10 \text{ nf, } R=2k5.$$

· The high-pass filter will eliminate any DC component, so a cutoff frequency of 200 Hz will be suitable for our purposes. Therefore,  $RC = \frac{\sqrt{1+\sqrt{2}}}{2\pi \cdot 200} = 1.23 \cdot 10^{-3}$ . We can choose **C=100nf, and R=12k3**.

### **Inverter amplifier**

As seen in the circuit:

$$(1) \frac{V_{in} - V_1}{Z_1} = \frac{V_1 - V_0}{R_2}, \text{ and since there is no current circulating through } R_3 \text{ } V_1=0.$$

We've called  $Z_1=C_1+R_1$ .

$$\frac{V_0}{V_{in}} = -\frac{R_2}{R_1 + \frac{1}{j\omega C_1}}, \text{ and } |V_0/V_{in}| = -\frac{R_2}{\sqrt{R_1^2 + \frac{1}{\omega^2 C_1^2}}}.$$

At medium frequencies we can suppose that  $Z_c$  tends to zero, and  $A_{vm} = -\frac{R_2}{R_1}$ .

· **Gain** must be 0 dB, so  $20 \log(A_{vm}) = 0 \Rightarrow |A_{vm}|=1 \Rightarrow R_1=R_2=R$ .

$$\cdot |A_{vb}(\omega)| = |A_{vm}| \frac{1}{\sqrt{2}} = \frac{1}{\sqrt{2}}. \text{ Therefore } -\frac{R_2}{\sqrt{R_1^2 + \frac{1}{\omega^2 C_1^2}}} = \frac{1}{\sqrt{2}} \text{ and}$$

$$\omega = \frac{1}{RC}.$$

· **Zin** should be of at least 100kΩ, and  $Z_{in} = \frac{V_i}{I_i} = R + \frac{1}{j\omega C_1}$ . At medium frequencies

$\frac{1}{j\omega C_1}$  tends to zero, so  $Z_{in} = R = 100 \text{ k}\Omega$ .

$$\cdot \omega = \frac{1}{RC_1} \Rightarrow C_1 = \frac{1}{2\pi \nu R} = \frac{1}{2\pi \cdot 160 \cdot 100 \cdot 10^3} = 9.94 \text{ nf} = 10\text{nf}$$

· **DC Compensation:** To compensate the effect of the leakage current we must assure there's no output voltage  $V_0$  in DC.

$$V_0 = G(V_p - V_n) = 0 \Rightarrow V_p = V_n.$$

$V_n = R_2 I_{bias}$ ,  $V_p = R_3 I_{bias} \Rightarrow R_3=R_2=100k\Omega$ .  $C_2$  is a decoupling capacitor so any value in the range 100nf to 1 μF is suitable. We will choose **C=100nf**.

**R1=R2=R3=100k, C1=10nf, C2=100nf**

### **Non-Inverter amplifier**

As seen in the circuit:

$$(1) (V_{in} - V) j \omega C I = \frac{V}{R_4}. \text{ We've called } Z_1 = R_3 // C_2$$

$$(2) \frac{V_0 - V}{R_2} = \frac{V}{R_1}.$$

$$\frac{V_0}{V_{in}} = \frac{1 + \frac{R_2}{R_1}}{1 + \frac{1}{j \omega R_4 C I}}. \text{ At medium frequencies } \frac{1}{j \omega R_4 C I} \text{ tends to zero, and}$$

$$\frac{V_0}{V_{in}} = 1 + \frac{R_2}{R_1}.$$

• **Gain** must be 0 dB, so  $20 \log(A_{vm}) = 0 \Rightarrow |A_{vm}| = 1 \Rightarrow 1 + \frac{R_2}{R_1} = 1 \Rightarrow R_2 = 0$

•  $|A_{vb}(\omega)| = |A_{vm}| \frac{1}{\sqrt{2}} = \frac{1}{\sqrt{2}}$ . Therefore  $\frac{1}{\sqrt{1 + \frac{1}{\omega^2 R_4 C I^2}}} = \frac{1}{\sqrt{2}}$  and

$$\omega = \frac{1}{R_4 C I}.$$

• **Z<sub>in</sub>** should be of at least 100kΩ, and  $Z_{in} = \frac{V_i}{I_i} = R_4 = 100 \text{ k}\Omega$ .

•  $\omega = \frac{1}{R_4 C I} \Rightarrow C I = \frac{1}{2 \pi \nu R_4} = \frac{1}{2 \pi 100000 160} = 10 \text{ nf}$

• **DC Compensation:** To compensate the effect of the leakage current we must assure there's no output voltage V<sub>0</sub> in DC.

$$V_0 = G (V_p - V_n) = 0 \Rightarrow V_p = V_n.$$

$$V_n = (R_3 + R_1 // R_2) I_{bias} \text{ (since } V_0 = 0), V_p = R_4 I_{bias} \Rightarrow G I_{bias} (R_4 - (R_3 + R_{eq})) = 0$$

$$R_4 = R_3 + R_1 // R_2 = R_3 \Rightarrow R_3 = R_4 = 100 \text{ k}\Omega.$$

• C<sub>2</sub> is a decoupling capacitor so any value in the range 100nf to 1 μF is suitable. We will choose C=100nf.

• R<sub>1</sub> we can choose it in any value, so let's make it 100 kΩ.

**R<sub>1</sub>=100k, R<sub>2</sub>=0k, R<sub>3</sub>=100k, R<sub>4</sub>=100k, C<sub>1</sub>=10nf, C<sub>2</sub>=100nf.**

### DC supression

We can supress DC components by adding an RC high pass filter in between some of the circuit stages. The cutoff frequency should be around 200 Hz.

$$\omega = \frac{1}{R C} \Rightarrow 2 \pi \nu = \frac{1}{R C} \Rightarrow R C = \frac{1}{2 \pi 200} = 7.96 10^{(-4)}. \text{ So we can choose:}$$

**R=8kΩ, C=100nF.**

### Modulating the signal

We've chosen a **74VHC4053** analog multiplexor to modulate the squared bipolar signal (see "Modulating signals" for a more detailed analysis). This chip has three built-in multiplexers with 2 inputs and one control signal. For our purposes a +5V/-5V feeding is needed to multiply the signal with the squared carrier. We will select one of the two unipolar squared signals generated by the circuit clock in the control input, depending on which bit is decoded on the digital decoding unit (see "selecting a random frequency" section for more details). This frequency selection is done with another multiplexor, so at least two of them are needed in the modulation process. The audio signal will pass through a unit-gain-non-inverting amplifier and a unit-gain-inverting amplifier respectively, each of them being connected to the inputs of the multiplexor/modulator.

### **–** *Lowpass filtering of the modulated signal: frequency inversion*

After modulation we have to filter the signal to eliminate all of the spectral replications to obtain frequency inversion. This should be done with two different low pass filters, one for the 6400 Hz signal, and the other for the 12800 one. But this is not quite an accurate way of achieving spectral suppression, because double circuitry would be needed. A more efficient way would be to use a single lowpass filter with a cutoff frequency of 12800 Hz, valid for both 12800 and 6400 spectrums. The only problem is 6400 Hz frequency will not be inverted, only replicated, but this is not such a big problem since audio encryption is also achieved by replicating the spectral components (this audio signal will not be intelgible in neither way). So with this design decission we calculate the values of the Sallen-Key filter for a fhc of 12800 Hz, which gives  $R C = 8.13 \cdot 10^{(-6)}$ . If **C=10 nF then R=812 Ω**.

### **–** *Transmission line*

The transmission line will be simulated by five cables: the modulated and inverted audio signal, the 6400 Hz clock, the 12800 Hz clock, the digital 2 Hz clock, and the codified frequency. Therefore, no additional sincronization is needed. We should notice though that relevant circuit noise will affect bit sincronism. For example, manipulating connections with the circuit connected to power will cause bit decoding errors. The only solution is resetting the circuit.

### **–** *Demodulation process*

Demodulating follows the same steps as modulating. We will point here a few differences. After demodulating the signal with the 74VHC4053 multiplexor attacked by the inverting and none-inverting amplifiers the signal should be lowpass filtered in order to eliminate undesired frequency components. This can be achieved using our now friend the lowpass Sallen-Key filter for a cutoff frequency of 4000 Hz. Subsitiuting in the equation we obtain:

$$R C = \frac{\sqrt{\sqrt{2} - 1}}{2 \pi 4000} = 2.56 \cdot 10^{(-5)}. \text{ } \mathbf{C=10 \text{ nf}, R=2k5.}$$

In the modulation process we decided to use only one filter to filter out both 6400 and 12800 frecuencies. There is a little price to pay for simplification: 6400 Hz information

will overlap at baseband resulting on an audio signal of double amplitude than its 12800 counterpart. This can be corrected by a simple attenuator made out of resistors. The value of this resistors is not very important, and 2k resistors have been used for convenience. This attenuator will only have to work for the 6400 Hz, and there is where the multiplexer comes again, to select which way will the signal have to travel. The signal is then again DC filtered and then passed to the power amplification stage

### **- The final step: power amplification**

We have chosen a 26dB power gain configuration from the National catalog. This particular configuration does not alter bass nor does it provide any special effects. If more gain is needed a different configuration can be chosen at will.

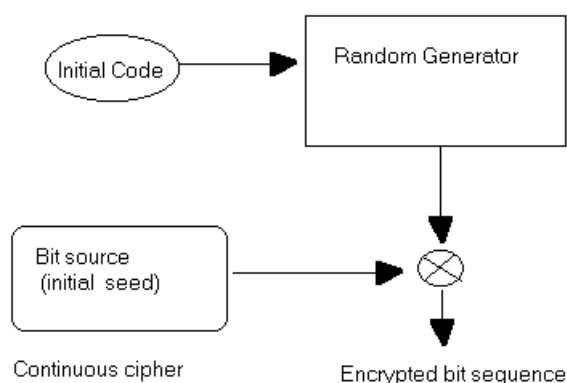
### **- Selecting a random frequency**

#### **- Introducing the initial random seed and the inversion frequency**

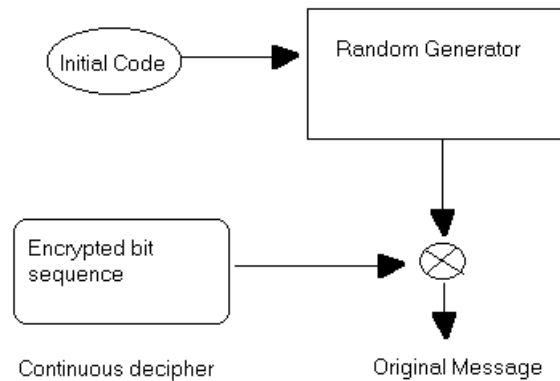
The initial random seed is selected using a 4-bit long microswitch, and the frequency selection bit is chosen using an 8-bit one. This microswitches are connected to +5V on one side and on the other they are both plugged to ground through a 100-300 Ohm resistor, and to the inputs of their respective chip. The 4-bit microswitch is also connected to a 7-segment display through a SN74LS248 BCD-TO-SEVEN-SEGMENT decoder (see *Visual Display* below), while the 8-bit microswitch is wired to an 8-bit circular buffer (74HC165).

#### **- Random Generator**

The easiest way to encrypt a bit sequence is by XORing it with another random sequence.

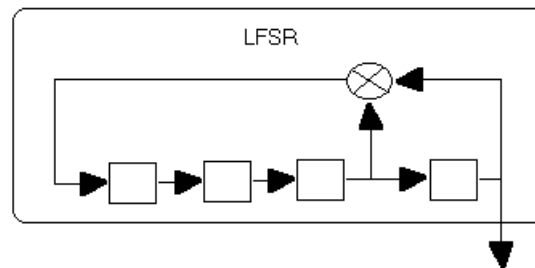


To decrypt the sequence we XOR it back with the original random sequence.



It is important to notice that **BOTH** random sequences must be completely equal to one another. If a pseudo-random generator is used that can be achieved by feeding it with the same initial seed.

We will now explore how to construct a pseudo-random generator with the aid of Linear Feedback Shift Registers (LFSR). The architecture of LFSR's is shown in the following figure.



LFSR's have a period of  $2^m - 1$ . In our circuit,  $m=4$  so there are 15 different pseudo-random output bits. We have chosen  $m=4$  because we are using a very simple model.

### **Visual Display**

The visual display of this circuit consists of the following: 4 LEDs to show the contents of the LFSR, 1 LED to show the codified frequency selection bit in the emitter and 1 in the receptor, 1 LED to show the uncoded frequency selection bit in the emitter and 1 in the receptor and 1 LED more to show the system clock. Also, as said above, there is a 7-segment display that shows the initial content of the LFSR.

### **Clock Generator**

The system clock is implemented using a RC oscillator, with an oscillation frequency given by  $\frac{1}{1.386 RC}$ . Although this is not a very precise scheme, it is very easy to implement and there is no need of high accuracy. We have chosen an oscillation frequency of about 2 Hz.

### ***Resetting the circuit***

Circuit resetting is needed in order to change the initial contents of the LFSR and to provide a starting point for synchronism. We would find it very weird that a channel would be randomly selected whenever we turned on the TV. Similarly, the initial contents of the LFSR cannot be initiated at random, so appropriate resetting should be done when plugging on the circuit. This can be done by a typical RC circuit scheme with a constant time  $\tau$  of at least twice the system clock period. This yields  $\tau > 1$  second. We have chosen  $R=2k7$  and  $C=1mF$  so  $\tau = 2.7$  seconds. We also need to be able to manually reset the circuit, so a push button and a debouncing subcircuit is needed. We find a small problem here. Both subcircuits shall be connected to the same reset inputs of the chips. While one of them may put a logical 1, the other one may be putting a 0. What shall we interpret, then? Well, a 0 means "reset now" while 1 means "don't reset". Therefore, if at least one of them says "reset now" a circuit resetting should be done, so we can implement this by wiring the two subcircuits using an AND gate. The output of the AND gate is the output of the RESET circuit.

### ***Emitter frequency ciphering***

The inversion frequency transmitted bit must be codified, and this is done using the output of our Random Generator XORed with the frequency selection bit (which is the output bit of the 8-bit circular buffer). The inputs to this subsystem are: the reset signal, the system clock, the inversion frequency bit and the initial random seed. The output is the ciphered bit.

### ***Receiver frequency deciphering***

To recover the original inversion bit we must XOR the received bit with the output sequence of the Random Generator. The initial random seed must be the same in both the emitter and the receiver, otherwise bit will not be properly unciphered. The inputs to this subsystem are: the reset signal, the system clock, the ciphered inversion frequency bit and the initial random seed. The output is the frequency inversion bit.

## **Improving the audio scrambler**

There are many ways in which this project could be improved. We shall only describe some of them. I originally had the idea of transmitting the signal via IR. This would have the advantage of avoiding a physical connection, being, at the same time, easier to implement than a radio transmission. A modulation frequency of around 100Khz could be used, and special care should be taken with problems such as fading,... Also, sinusoidal modulation would have to be used, and a frequency mixer (instead of our multiplexer solution), so circuit subsystems would have been a little bit more complicated. There's yet another problem. A different type of modulation rather than DBL would have to be used, since there's no point on sending the audio signal along and not

the system clock. Mixing both digital and analog signals is not an easy matter, and will probably fall beyond the scope of this project.

Another way to improve the project is by digitally ciphering the information. An A/D and D/A conversion should be done, in order to apply encryption techniques in the digital domain. This could be achieved by the use of a microcontroller provided it is fast enough (it could be done with the amount of bandwidth used in this project with any of the 68HC11/12 microcontrollers family).

## [-] A few final words

Algorithms used in cryptography are in continuous advance. In 1917 the *Vigenère* encryption code had been described as *unbreakeble* by the prestigious magazine *Scientific American*. Nowadays, a message encrypted using this code can be broken in very few seconds. The analog techniques used in this paper are by no means totally secure, but secure enough for our purposes.

We use scrambling technology when speaking over the phone from India to the Maurice Islands, or when ringing some Nicaraguan guerrilla. It is important to keep a person's right to privacy, and encrypting a few e-mails or telephonic conversations is definteley not enough. Atoms are loosing specific weight in favor of bits, the philosopher's stone of the twenty first century.

Negroponte's *global village*, a golden dream of non-discrimination in the basis of race, sex, religious affiliation or birthplace is dying of cancer. Computer power, along with all of the existing databases which contain personal information, provides any government or *rich-n-wealthy* enough individual with comprehensive information about anyone living on a developed country: complete physical description, buying preferences, sexual inclinations, marital status, past diseases and future potential ones, economic status, political affinities,... anything from what Blockbuster movie do we like to which color of underwear are we using. We will soon become slaves of our electronic portraits, finding a job only if we fit a certain electronic pattern.

Our personal records are out there, waiting for someone to ask for them. And, if things continue to go in the way they seem to be going, human kind shall be heading towards self-destruction. World War III will be fought in the basis of information, and will be much more devastating than traditional wars, because not only will the economy of the nations or their military power be destroyed, but also the personal lifes of its people will be exposed. Our deepest fears will then be known and made real, our worst nightmares will become true. Next time you enter the Internet, beware of Big Brother. He's only one click away.

## [-] Acknowledgements

Any project involves a considerable amount of work carried out by many people, not only the author. I would like to express my deep gratitude to **Alicia Díaz-Chirón** for her generous help in all of the design and development processes of this audio scrambler, and for her unconditional support and her ardent enthusiasm showed in the continuous suggestions and ideas brought to me, which went way beyond friendship. **Je**<sub>Page 18</sub>**martín** spent her time during Christmas Day

pointing out the errors I committed in the implementation of the scrambler, and making constructive suggestions on how to fixing them. **Yazán Senan** let me borrow his oscilloscope and shared with me his opinions on the design. **Ángel Navarro** did some of the drawings of the circuits, and **Carlos Rey** gave me technical support on using *PSPICE* and *Matlab*. **Oscar Pérez** suggested me to use a D-latch to both divide the frequency of the system clock and make a 50% duty cycle. Also, **Isabel Torres** and my lab instructor, **Asst. Prof. Javier Macías** have given me useful remarks and encouraged me throughout the design of the project. I am deeply indebted to them all for their kind help and for the valuable time they have shared with me. Needless to say, without all the above help and support, the writing and production of this project would not have been possible.

## **- Bibliographical references**

### **- Paper references**

- 1] *Feynman, Richard & Leighton, Robert & Sands, Matthew*. The Feynman lectures on Physics. 1963. Ed. Addison-Wesley. Reading, Massachusets (USA)
- 2] *Oppenheim, Alan*. Signals and Systems. 1997. Ed. Prentince Hall. New York (USA)
- 3] *Lacanette, Kerry*. A Basic Introduction to Filters-Active, Passive and Switched-Capacitor. 1995. Ed. National Semiconductors Application Note 779. USA.
- 4] *Haykin, Simon*. Communication Systems. 1994. Ed. Wiley. New York (USA)

### **- Online references**

[1] *Ehlers, Doug*. Scrambling essentials.

WWW Address: <http://www.transcrypt.com/Pages/scrames.html>

[2] *Macías, Javier*. Enunciado de la práctica del Laboratorio de Circuitos Electrónicos (LCEL). Sistema de cifrado y descifrado de audio.

WWW Address: <http://www-gth.die.upm.es/~macias/lcel.html>

[3] *Macías, Javier*. Notas complementarias del Laboratorio de Circuitos Electrónicos (LCEL).

WWW Address: <http://www-gth.die.upm.es/~macias/lcel.html>

[4] *B. Crespo, Carlos*. Satélites de comunicaciones.

WWW Address: <http://www.dat.etsit.upm.es/~cbousono/satcom/index.htm>

[5] *Rivest, Ron*. Cryptographic links.

WWW Address: <http://theory.lcs.mit.edu/~rivest/crypto-security.html>

## **- Appendix A: Design schematics and Bode plots**

## **Appendix B: Catalog of components used**