Pulse-density modulation

From Academic Kids

Pulse-density modulation, or **PDM**, is a form of modulation used to represent an analog signal in the digital domain. In a **PDM** signal, specific amplitude values are not encoded into pulses as they would be in PCM or PWM. Instead it is the relative density of the pulses that corresponds to the analog signals amplitude.

Contents

Basics

In a pulse-density modulation bitstream a 1 corresponds to a pulse and a 0 corresponds to the absence of a pulse. A run consisting of all 1's would correspond to a positive amplitude value, all 0's would correspond to a negative amplitude value, and alternating 1's and 0's would correspond to a zero amplitude value. This has the remarkable effect of the **PDM** bitstream actually looking like the wave it represents.

Analog-to-digital conversion

A **PDM** bitstream is encoded from an analog signal through the process of Sigma-delta modulation. This process uses a one bit quantizer that produces either a 1 or 0 depending on the amplitude of the analog signal. A 1 or 0 corresponds to a signal that is all the way up or all the way down, respectively. Because in the real world analog signals are rarely all the way in one direction there is a quantization error, the difference between the 1 or 0 and the actual amplitude it represents. This error is fed back negatively in the sigma-delta modulation process loop. In this way every error successively influences every other quantization measurement and its error. This has the effect of averaging out the quantization error.

Digital-to-analog conversion

The process of decoding a **PDM** signal into an analog one is amazingly simple. One only has to pass that signal through an analog low-pass filter. This works because the function of a low-pass filter is essentially to average the signal. The density of pulses is measured by the average amplitude of those pulses over time, thus a low pass filter is the only step required in the decoding process.

Examples/algorithm

A single period of the trigonometric sine function, sampled 100 times and represented as a **PDM** bitstream, is:

0101011011110111111111111111111111011111101101101010100100100000010000000000000000000001000010010101

Two periods of the sine wave would appear as:

0101101111111111111101101010010000000000000100010011011101111111111111011010100100000000000000100101

As you can see in **Pulse-*density* modulation** a high *density* of 1's occurs at the peaks of the sine wave, while a low *density* of 1's occurs at the troughs of the sine wave.

The following algorithm can be used to replicate the above examples.

```
//Produce s samples of p periods of a sine wave
function sample(int s, int p)
  s := s - 1
  var real ω := p × π × 2 ÷ s
  var real[0..s] pcm
  for i from 0 to s
      pcm[i] := sin(ω × i)
  return pcm

//Encode samples into pulse-density modulation
function encode(int[0..s] pcm)
  var real[0..s] pdm
  var real r := 1
  for i from 0 to s
      r := pcm[i] - r
      if r > 0
          pdm[i] := 1
          r := 1 - r
      else
          pdm[i] := 0
          r := -1 - r
  return pdm
```