



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

[Interaction](#)
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

[Tools](#)
[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

[Print/export](#)
[Create a book](#)
[Download as PDF](#)
[Printable version](#)

[Languages](#)
[Català](#)
[Deutsch](#)
[Español](#)
[Français](#)
[한국어](#)
[日本語](#)
[Русский](#)
[Tiếng Việt](#)
[Yorùbá](#)

[Edit links](#)

[Create account](#) [Log in](#)

[Article](#) [Talk](#)

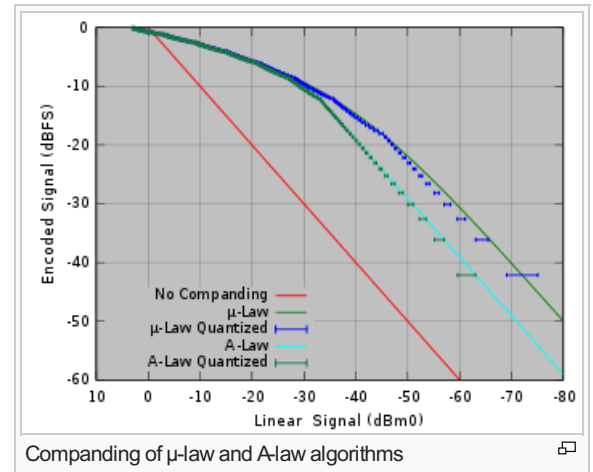
[Read](#) [Edit](#) [View history](#)

μ-law algorithm

From Wikipedia, the free encyclopedia
(Redirected from [Mu-law algorithm](#))

The **μ-law algorithm** (sometimes written "mu-law", often [approximated](#) as "u-law") is a [companding](#) algorithm, primarily used in 8-bit [PCM digital telecommunication](#) systems in [North America](#) and [Japan](#). Companding algorithms reduce the [dynamic range](#) of an audio [signal](#). In analog systems, this can increase the [signal-to-noise ratio](#) (SNR) achieved during transmission; in the digital domain, it can reduce the quantization error (hence increasing signal to quantization noise ratio). These SNR increases can be traded instead for reduced [bandwidth](#) for equivalent SNR.

It is similar to the [A-law algorithm](#) used in regions where digital telecommunication signals are carried on E-1 circuits, e.g. Europe.



Contents

- [1 Algorithm types](#)
 - [1.1 Continuous](#)
 - [1.2 Discrete](#)
- [2 Implementation](#)
- [3 Usage justification](#)
- [4 Comparison with A-law](#)
- [5 See also](#)
- [6 References](#)
- [7 External links](#)

Algorithm types [\[edit\]](#)

There are two forms of this algorithm—an analog version, and a quantized digital version.

Continuous [\[edit\]](#)

For a given input *x*, the equation for μ-law encoding is^{[\[1\]](#)}

$$F(x) = \operatorname{sgn}(x) \frac{\ln(1 + \mu|x|)}{\ln(1 + \mu)} \quad -1 \leq x \leq 1$$

where *μ* = 255 (8 bits) in the North American and Japanese standards. It is important to note that the [range](#) of this function is −1 to 1.

μ-law expansion is then given by the inverse equation:

$$F^{-1}(y) = \operatorname{sgn}(y) (1/\mu) ((1 + \mu)^{|y|} - 1) \quad -1 \leq y \leq 1$$

The equations are culled from [Cisco's Waveform Coding Techniques](#) [\[2\]](#).

Discrete [\[edit\]](#)

This is defined in ITU-T Recommendation [G.711](#).^{[\[2\]](#)}

G.711 is unclear about how to code the values at the limit of a range (e.g. whether +31 codes to 0xEF or 0xF0). However, G.191 provides example C code for a μ-law encoder which gives the following encoding. Note the difference between the positive and negative ranges, e.g. the negative range corresponding to +30 to +1 is −31 to −2. This is accounted for by the use of [1's complement](#) (simple bit inversion) rather than [2's complement](#) to convert a negative value to a positive value during encoding.

Quantized μ -law algorithm

14 bit Binary Linear input code	8 bit Compressed code
+8158 to +4063 in 16 intervals of 256	0x80 + interval number
+4062 to +2015 in 16 intervals of 128	0x90 + interval number
+2014 to +991 in 16 intervals of 64	0xA0 + interval number
+990 to +479 in 16 intervals of 32	0xB0 + interval number
+478 to +223 in 16 intervals of 16	0xC0 + interval number
+222 to +95 in 16 intervals of 8	0xD0 + interval number
+94 to +31 in 16 intervals of 4	0xE0 + interval number
+30 to +1 in 15 intervals of 2	0xF0 + interval number
0	0xFF
-1	0x7F
-31 to -2 in 15 intervals of 2	0x70 + interval number
-95 to -32 in 16 intervals of 4	0x60 + interval number
-223 to -96 in 16 intervals of 8	0x50 + interval number
-479 to -224 in 16 intervals of 16	0x40 + interval number
-991 to -480 in 16 intervals of 32	0x30 + interval number
-2015 to -992 in 16 intervals of 64	0x20 + interval number
-4063 to -2016 in 16 intervals of 128	0x10 + interval number
-8159 to -4064 in 16 intervals of 256	0x00 + interval number

Implementation [\[edit\]](#)

There are three ways of implementing a μ -law algorithm:

Analog

Use an amplifier with non-linear gain to achieve companding entirely in the analog domain.

Non-linear ADC

Use an [analog-to-digital converter](#) with quantization levels which are unequally spaced to match the μ -law algorithm.

Digital

Use the quantized digital version of the μ -law algorithm to convert data once it is in the digital domain.

Usage justification [\[edit\]](#)

This encoding is used because [speech](#) has a wide [dynamic range](#). In the analog world, when mixed with a relatively constant background noise source, the finer detail is lost. Given that the precision of the detail is compromised anyway, and assuming that the signal is to be perceived as audio by a human, one can take advantage of the fact that the perceived [acoustic intensity level](#) or [loudness](#) is logarithmic by compressing the signal using a logarithmic-response operational amplifier ([Weber-Fechner law](#)). In telecommunications circuits, most of the noise is injected on the lines, thus after the compressor, the intended signal is perceived as significantly louder than the static, compared to an un-compressed source. This became a common solution, and thus, prior to common digital usage, the μ -law specification was developed to define an inter-compatible standard.

In digital systems this pre-existing algorithm had the effect of significantly reducing the number of bits needed to encode recognizable human voice. Using μ -law, a sample could be effectively encoded in as few as 8 bits, a sample size that conveniently matched the symbol size of most standard computers.

μ -law encoding effectively reduced the dynamic range of the signal, thereby increasing the [coding](#) efficiency while biasing the signal in a way that results in a signal-to-[distortion](#) ratio that is greater than that obtained by linear encoding for a given number of bits. This is an early form of perceptual audio encoding.

The μ -law algorithm is also used in the [.au format](#), which dates back at least to the [SPARCstation 1](#) by Sun Microsystems as the native method used by the `/dev/audio` interface, widely used as a de facto standard for sound on Unix systems. The au format is also used in various common audio [APIs](#) such as the classes in the `sun.audio` [Java package](#) in [Java](#) 1.1 and in some [C#](#) methods.

This plot illustrates how μ -law concentrates sampling in the smaller (softer) values. The abscissa represents the

byte values 0-255 and the vertical axis is the 16-bit linear decoded value of μ -law encoding.

Comparison with A-law [edit]

The μ -law algorithm provides a slightly larger dynamic range than the A-law at the cost of worse proportional distortion for small signals. By convention, A-law is used for an international connection if at least one country uses it.

See also [edit]

- [Audio level compression](#)
- [Signal compression \(disambiguation\)](#)
- [G.711](#)

References [edit]

This article incorporates **public domain material** from the [General Services Administration](#) document "[Federal Standard 1037C](#)".

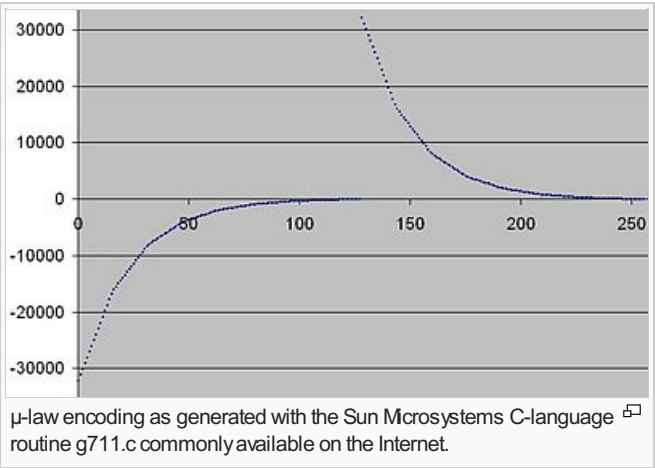
- ↑ "Cisco - Waveform Coding Techniques". Retrieved 2008-07-29.
- ↑ "ITU-T Recommendation G.711".

External links [edit]

- [Waveform Coding Techniques](#) – details of implementation
- [A-Law and mu-Law Companding Implementations Using the TMS320C54x](#) (PDF)
- [TMS320C6000 \$\mu\$ -Law and A-Law Companding with Software or the McBSP](#) (PDF)
- [A-law and \$\mu\$ -law realisation \(in C\)](#) (ctrl-a "highlight all" to see linked black-on-black text).

v t e	Data compression methods	[show]
v t e	Multimedia compression and container formats	[show]

Categories: Audio codecs ITU-T recommendations
--



This page was last modified on 6 August 2014, at 10:15.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

