(Draft -- May 28, 1998 -- Draft)

Specifications for the Analog to Digital Conversion of Voice by 2,400 Bit/Second Mixed Excitation Linear Prediction

1. INTRODUCTION

This standard describes the interoperability requirements relating to the conversion of analog voice to 2,400 bits/s digitized voice by a method known as Mixed Excitation Linear Prediction (MELP) and reconversion back to analog voice. An algorithm description is also included to aid implementation as well as a performance verification process to verify an implementation.

2. CONVENTIONS AND DEFINITIONS

2.1 Frame Size

A MELP frame interval is 22.5 ms ± 0.01 percent in duration and contains 180 voice samples (8,000 samples/s).

2.2 Analog Specification

The recommended analog requirements for the MELP coder are for a nominal bandwidth ranging from 100 Hz to 3800 Hz. Although the MELP coder will operate with a more band limited signal, performance degradation will result. To ensure proper operation of the MELP coder, the A/D conversion process should produce peak values of (or near) -32768 and 32767. Additionally, the coder should have unity gain, which means that the output speech level should match that of the input speech.

3. ALGORITHM DESCRIPTION

3.1 Coder Overview

The Mixed Excitation Linear Prediction coder is based on the traditional Linear Prediction Coding (LPC) parametric model, but also includes five additional features [1][2]. These are: mixed excitation, aperiodic pulses, adaptive spectral enhancement, pulse dispersion, and Fourier magnitude modeling. These features are illustrated in the MELP decoder block diagram shown in Figure 1.

The mixed excitation is implemented using a multi-band mixing model. This model can simulate frequency-dependent voicing strength using an adaptive filtering structure implemented with a fixed filter bank. The primary effect of this mixed excitation is to reduce the buzz usually associated with LPC vocoders, especially in broadband acoustic noise.

When the input speech is voiced, the MELP coder can synthesize using either periodic or aperiodic pulses. Aperiodic pulses are used most often during transition regions between voiced and unvoiced segments of the speech signal. This feature enables the decoder to reproduce erratic glottal pulses without introducing tonal sounds.

The adaptive spectral enhancement filter is based on the poles of the linear prediction synthesis filter. Its use enhances the formant structure of the synthetic speech and improves the match between the synthetic and natural bandpass waveforms. It also gives the synthetic speech a more natural quality.

Pulse dispersion is implemented using a fixed filter based on a spectrally-flattened triangle pulse. This filter spreads the excitation energy within a pitch period, reducing some of the harsh quality of the synthetic speech.

The first ten Fourier magnitudes are determined from the peaks of the Fourier transform of the prediction residual signal. The information in these coefficients improves the accuracy of the speech production model at the perceptually-important lower frequencies. This increases the quality of the synthetic speech, particularly for male speakers and when background noise is present.

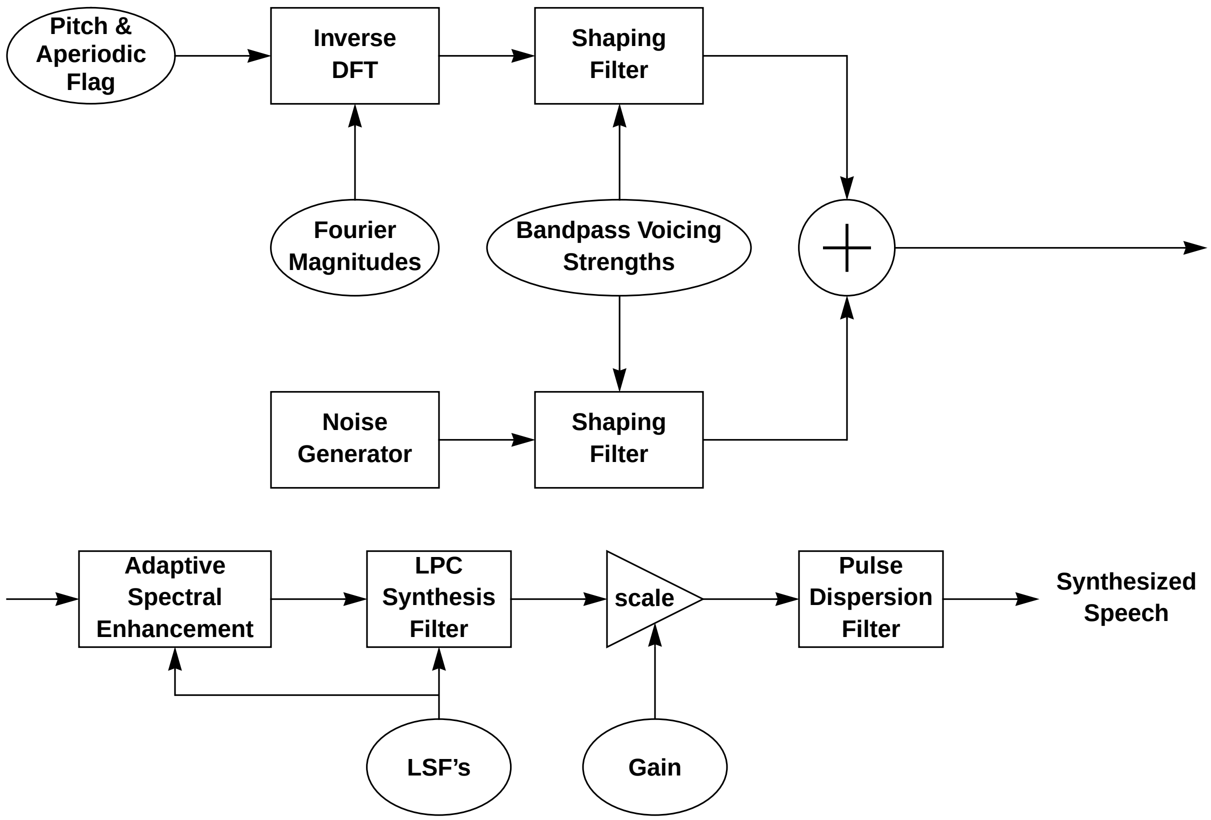

Figure 1. MELP Decoder Block Diagram

3.2 Encoder

Input speech is encoded by performing the following steps in the order given.

3.2.1 **Low Frequency Removal.** The first step in the encoding process is to remove any low frequency energy which may be present in the input signal. This is accomplished with a 4^{th} order Chebychev type II highpass filter, having a cutoff frequency of 60 Hz and a stopband rejection of 30 dB. The filter output is referred to as the input speech signal throughout the following encoder description.

A buffer containing the most recent samples of the input speech signal is maintained in the encoder. One of these samples is designated the last sample in the current frame. The buffer extends beyond this sample into the past and future to contain the samples needed for the encoding process. The last sample in the current frame serves as a reference point for many of the encoder calculations.

3.2.2 **Integer Pitch Calculation.** For this pitch calculation, the input speech signal is first processed with a 1 kHz, 6^{th} order Butterworth lowpass filter. The integer pitch value, P_1, is the value of τ, $\tau = 40, 41, \ldots, 160$, for which the normalized autocorrelation function, $r(\tau)$, is maximized. This function is defined by:

$$r(\tau) = \frac{c_\tau(0,\tau)}{\sqrt{c_\tau(0,0)c_\tau(\tau,\tau)}}, \tag{1}$$

where

$$c_\tau(m,n) = \sum_{k=-\lfloor\tau/2\rfloor-80}^{-\lfloor\tau/2\rfloor+79} s_{k+m}s_{k+n}, \tag{2}$$

and $\lfloor\tau/2\rfloor$ represents truncation to an integer value. The center of the pitch analysis window is at sample s_0 in Eq. (2). For the integer pitch calculation, this window is centered on the last sample in the current frame. The lowpass filter output is sample s_0 when its input is the last sample in the current frame. The time index k in the autocorrelation preserves the pitch analysis window alignment around its center point; the normalization compensates for changing signal amplitudes. The final pitch calculation (Section 3.2.9) extends the pitch range to a lag of 20 samples.

3.2.3 **Bandpass Voicing Analysis.** This portion of the encoder determines the five bandpass voicing strengths, Vbp_i, $i = 1, 2, \ldots, 5$. It also refines the integer pitch measurement and the corresponding normalized autocorrelation value. The bandpass voicing analysis begins by filtering the input speech signal into five frequency bands. These filters are 6^{th} order Butterworth, with passbands of 0-500, 500-1000, 1000-2000, 2000-3000, and 3000-4000 Hz.

A refined pitch measurement is made using the 0-500 Hz filter output signal. This measurement is centered on the filter output produced when its input is the last sample in the current frame. Two pitch candidates are considered in this refinement, namely the integer pitch values (P_1) from the current and previous frames. For each candidate, Eq. (1) is used to perform an integer pitch search over lags from 5 samples shorter to 5 samples longer than the candidate, and a fractional pitch refinement (Section 3.2.4) is performed around the optimum integer pitch lag. This produces two fractional pitch candidates and their corresponding normalized autocorrelation values. The candidate having the higher normalized autocorrelation is selected as the fractional pitch, P_2. The corresponding normalized autocorrelation, $r(P_2)$, is saved as the lowest band voicing strength, Vbp_1. P_2 is saved for use in determining the voicing strength for the remaining frequency bands. It is also used in the final pitch calculation (Section 3.2.9) and gain calculation (Section 3.2.11).

For each remaining band, the bandpass voicing strength is the larger of $r(P_2)$ as determined by the fractional pitch procedure for the bandpass signal and the time envelope of the bandpass signal, where $r(P_2)$ for the time envelope is first decremented by 0.1 to compensate for an experimentally observed bias (due to the smoothness of the time envelope signals). The envelopes are calculated by full-wave rectification followed by a smoothing filter. This filter consists of a zero at DC in cascade with a complex pole pair at 150 Hz with a radius of 0.97. For each calculation of $r(P_2)$, the analysis window is centered on the last sample in the current frame, as was the case for the first band.

3.2.4 **Fractional Pitch Refinement.** This procedure, which is used at several places in the encoding process, utilizes an interpolation formula to increase the accuracy of an input pitch value. This value is first rounded to the nearest integer. Assume that this integer has a value of T samples. The interpolation formula presumes that $r(\tau)$ has a maximum between lags of T and T+1. Hence, $c_T(0, T-1)$ and $c_T(0, T+1)$ are computed and compared to determine if the maximum is more likely to fall between T and T+1 or between T-1 and T. If $c_T(0, T-1) > c_T(0, T+1)$, then the maximum probably falls between T-1 and T and the pitch, T, is decremented by one prior to interpolation. The fractional offset, Δ, is then computed by the interpolation equation:

$$\Delta = \frac{c_T(0,T+1)c_T(T,T) - c_T(0,T)c_T(T,T+1)}{c_T(0,T+1)[c_T(T,T) - c_T(T,T+1)] + c_T(0,T)[c_T(T+1,T+1) - c_T(T,T+1)]}, \tag{3}$$

where $c_T(m, n)$ is defined by Eq. (2). In some cases, this formula produces an offset outside the range of 0.0 to 1.0, so the offset is clamped between -1 and 2. The fractional pitch is $T + \Delta$ and is clamped between 20 and 160.

The normalized autocorrelation at the fractional pitch value is given by:

$$r(T + \Delta) = \frac{(1 - \Delta)c_T(0, T) + \Delta c_T(0, T + 1)}{\sqrt{c_T(0, 0)[(1 - \Delta)^2 c_T(T, T) + 2\Delta(1 - \Delta)c_T(T, T + 1) + \Delta^2 c_T(T + 1, T + 1)]}} . \tag{4}$$

The fractional pitch refinement procedure is based on work presented in [3]. Equations (3) and (4) produce the fractional offset and corresponding normalized autocorrelation which would be obtained if the input signal had been linearly interpolated to obtain values between the actual sampling times.

3.2.5 **Aperiodic Flag.** The aperiodic flag is set to 1 if $Vbp_1 < 0.5$ and set to 0 otherwise. The Vbp_1 value determined by bandpass voicing analysis (Section 3.2.3) is used for this comparison. When set, this flag tells the decoder that the pulse component of the excitation should be aperiodic, rather than periodic. Section 3.3.1 describes the use of the aperiodic flag.

3.2.6 **Linear Prediction Analysis.** A 10th order linear prediction analysis is performed on the input speech signal using a 200 sample (25 ms) Hamming window centered on the last sample in the current frame. The traditional autocorrelation analysis procedure is implemented using the Levinson-Durbin recursion. In addition, a bandwidth expansion coefficient of 0.994 (15 Hz) is applied to the prediction coefficients, a_i, $i = 1, 2, \ldots, 10$, where each coefficient is multiplied by 0.994^i.

3.2.7 **Linear Prediction Residual Calculation.** The linear prediction residual signal is calculated by filtering the input speech signal with the prediction filter whose coefficients were determined by linear prediction analysis (Section 3.2.6). The residual window is centered on the last sample in the current frame, and is made wide enough for use by the final pitch calculation (Section 3.2.9).

3.2.8 **Peakiness Calculation.** The peakiness of the residual signal is calculated over a 160 sample window centered on the last sample in the current frame. The peakiness value is the ratio of the L2 norm to the L1 norm of the residual signal, r_n, in the window:

$$peakiness = \frac{\sqrt{\frac{1}{160}\sum_{n=1}^{160} r_n^2}}{\frac{1}{160}\sum_{n=1}^{160} |r_n|} . \tag{5}$$

If the peakiness exceeds 1.34, then the lowest band voicing strength, Vbp_1, is forced to 1.0. If the peakiness exceeds 1.6, then the lowest three band voicing strengths, Vbp_i, $i = 1, 2, 3$, are all forced to 1.0. This is the only use of the peakiness measure.

3.2.9 **Final Pitch Calculation.** The final pitch measurement uses the lowpass filtered residual signal, where the filter is a 6th order Butterworth, with a 1 kHz cutoff. Eq. (1) is used to perform an integer pitch search over lags from 5 samples shorter to 5 samples longer than P_2, rounded to the nearest integer. This measurement is centered on the filter output produced when its input is the last residual sample in the current frame. A fractional pitch refinement (Section 3.2.4) is then made around the optimum integer pitch lag. This produces tentative values for the final pitch, P_3, and for the corresponding normalized autocorrelation, $r(P_3)$.

If $r(P_3) \geq 0.6$, the pitch doubling check procedure (Section 3.2.10) is performed on the filtered residual, using P_3 as the candidate pitch, and doubling threshold $D_{th} = 0.75$ if $P_3 \leq 100$, or $D_{th} = 0.5$ otherwise. The doubling check procedure may produce new values for P_3 and $r(P_3)$.

The else action for the preceding if is as follows. A fractional pitch refinement around P_2 is performed using the input speech signal. This measurement is centered on the last sample in the current frame and produces new values for P_3 and $r(P_3)$. If $r(P_3) < 0.55$, then P_3 is replaced by P_{avg}, the long-term average pitch (Section 3.2.12). Otherwise, the pitch doubling check procedure is performed on the input speech signal, using P_3 as the candidate pitch, and doubling threshold $D_{th} = 0.9$ if $P_3 \leq 100$, or $D_{th} = 0.7$ otherwise. The doubling check procedure may produce new values for P_3 and $r(P_3)$.

Finally, if $r(P_3) < 0.55$, then P_3 is replaced by P_{avg}.

The following pseudo code shows the final pitch algorithm:

```
inputs: the input speech signal; the residual signal; P2; Pavg
outputs: P3, cor_P3

fresid buffer = filter the residual with a 1 kHz Butterworth
P3 = best integer pitch on fresid over the range P2-5 to P2+5
P3, cor_P3 = frac_pitch(fresid, P3)
if (cor_P3 >= 0.6)
    Dth = 0.5
    if (P3 <= 100) Dth = 0.75
    P3, cor_P3 = double_ck(fresid, P3, Dth)
else
    P3, cor_P3 = frac_pitch(input, P2)
    if (cor_P3 < 0.55)
        P3 = Pavg
    else
        Dth = 0.7
        if (P3 <= 100) Dth = 0.9
        P3, cor_P3 = double_ck(input, P3, Dth)
    endif
endif
if (cor_P3 < 0.55) P3 = Pavg
```

3.2.10 **Pitch Doubling Check.** The pitch doubling check procedure looks for and corrects pitch values which are multiples of the actual pitch. This procedure takes a signal, a candidate pitch P, and a doubling threshold D_{th}, and returns the checked pitch P_c, and the corresponding correlation, $r(P_c)$. All fractional pitch calculations are made using the signal given to the doubling check procedure.

This procedure begins with a fractional pitch refinement around P. This produces tentative values for P_c and $r(P_c)$. Next, the largest value of k is found for which $r(P_c/k) > D_{th} r(P_c)$, where $(P_c/k) \geq 20$ and $k = 8, 7, \ldots, 2$. $r(P_c/k)$ is calculated in two steps: 1) a fractional pitch refinement around P_c/k, producing P_k; and 2) a double verification, if $P_k < 30$. If such a k is found, then a fractional pitch refinement around P_k is performed, producing new values for P_c and $r(P_c)$.

Finally, if P_c is less than 30 samples, then double verification is performed.

The following pseudo code shows the pitch double check procedure:

```
inputs: signal; P; Dth
outputs: Pc, cor_Pc

Pc, cor_Pc = frac_pitch(signal, P)
for (k=8; k>=2; k--)
    Pk = Pc/k
    if (Pk >= 20)
        Pk, cor_Pk = frac_pitch(signal, Pk)
        if (Pk < 30) cor_Pk = double_ver(Pk, cor_Pk)
        if (cor_Pk > Dth * cor_Pc)
            Pc, cor_Pc = frac_pitch(signal, Pk)
            break
        endif
    endif
endfor
if (Pc < 30) cor_Pc = double_ver(Pc, cor_Pc)
```

For inputs P and $r(P)$, the double verification procedure returns the smaller of $r(P)$ and $r(2P)$, where $r(2P)$ is determined by the fractional pitch procedure around $2P$. The use of double verification in the double check procedure provides robustness against spurious short pitch values.

3.2.11 **Gain Calculation.** The input speech signal gain is measured twice per frame using a pitch-adaptive window length. This length is identical for both gain measurements and is determined as follows. When $Vbp_1 > 0.6$, the window length is the shortest multiple of P_2 which is longer than 120 samples. If this length exceeds 320 samples, it is divided by 2. When $Vbp_1 \le 0.6$, the window length is 120 samples. The gain calculation for the first window produces G_1 and is centered 90 samples before the last sample in the current frame. The calculation for the second window produces G_2 and is centered on the last sample in the current frame. The gain is the RMS value, measured in dB, of the signal in the window, s_n:

$$G_i = 10\log_{10}\left(0.01 + \frac{1}{L}\sum_{n=1}^{L} s_n^2\right), \tag{6}$$

where L is the window length. The 0.01 term prevents the log argument from going too close to zero. If a gain measurement is less than 0.0, it is clamped to 0.0. The gain measurement assumes that the input signal range is -32768 to 32767 (Section 2.2).

3.2.12 **Average Pitch Update.** The long-term average pitch, P_{avg}, is updated with a simple smoothing procedure. If $r(P_3) > 0.8$ and $G_2 > 30$ dB, then P_3 is placed into a buffer containing the three most recent strong pitch values, p_i, $i = 1, 2, 3$. Otherwise, all three pitch values in the buffer are moved toward a default pitch, $P_{default} = 50$ samples, according to:

$$p_i = 0.95 p_i + 0.05 P_{default}\ ,\ i = 1, 2, 3\ . \tag{7}$$

The average pitch is then updated as the median of the three values in the buffer. P_{avg} is used in the final pitch calculation (Section 3.2.9).

3.2.13 **Quantization of Prediction Coefficients.** First, the linear prediction coefficients a_i, $i = 1, 2, \ldots, 10$, are converted into line spectrum frequencies (LSF's). Details of the conversion algorithm can be found in [4]. Next, a process which forces the LSF components to be in ascending order with a minimum separation of 50 Hz is performed. This process begins by checking all adjacent pairs of the LSF components and swapping any pair not in ascending order. This step is repeated as many as ten times, if necessary. The minimum separation criterion is then applied by correcting each pair, f_i and f_{i+1}, for which $d = f_{i+1} - f_i$ is less than 50 Hz, Δ_{min}, as shown in the following pseudo code. The LSF components and frequency-related constants are in Hertz; scaling in other implementations may differ. The minimum separation process is repeated ten times.

```
dmin = 50
for (i=1; i<10; i++)
    d = f[i+1] - f[i]
    if (d < dmin)
        s1 = s2 = (dmin-d)/2
        if (i == 1 and f[i] < dmin) s1 = f[i]/2
        else if (i > 1)
            tmp = f[i] - f[i-1]
            if      (tmp < dmin)   s1 = 0
            else if (tmp < 2*dmin) s1 = (tmp-dmin)/2
        endif
        if (i == 9 and f[i+1] > 4000-dmin) s2 = (4000-f[i+1])/2
        else if (i < 9)
            tmp = f[i+2] - f[i+1]
            if      (tmp < dmin)   s2 = 0
            else if (tmp < 2*dmin) s2 = (tmp-dmin)/2
        endif
```

```
        f[i]   = f[i]   - s1
        f[i+1] = f[i+1] + s2
    endif
endfor
```

The resulting LSF vector, $\boldsymbol{f}$, is then quantized using a multi-stage vector quantizer (MSVQ). The MSVQ codebook consists of four stages of 128, 64, 64, and 64 levels respectively. The quantized vector, $\hat{\boldsymbol{f}}$, is the sum of the vectors selected by the search process, with one vector selected from each stage. The MSVQ search finds the codebook vector which minimizes the square of the weighted Euclidean distance, d^2, between the unquantized and quantized LSF vectors:

$$d^2(\boldsymbol{f}, \hat{\boldsymbol{f}}) = \sum_{i=1}^{10} w_i (f_i - \hat{f}_i)^2, \tag{8}$$

where

$$w_i = \begin{cases} P(f_i)^{0.3}, & 1 \le i \le 8 \\ 0.64P(f_i)^{0.3}, & i = 9 \\ 0.16P(f_i)^{0.3}, & i = 10 \end{cases}, \tag{9}$$

f_i is the i^{th} component of the unquantized LSF vector, and $P(f_i)$ is the inverse prediction filter power spectrum evaluated at frequency f_i. The search procedure is an M-best approximation to a full search, in which the M=8 best code vectors from each stage are saved for use with the next stage; reference [5] has additional details. The process to ensure ascending order and minimum separation (described in the first part of this section) is then applied to the quantized LSF vector. The resulting vector is used in the Fourier magnitude calculation (Section 3.2.17).

3.2.14 **Pitch Quantization.** The final pitch value, P_3, is quantized on a logarithmic scale with a 99-level uniform quantizer ranging from 20 to 160 samples. These pitch values are then mapped to a 7-bit codeword using a look-up table, as shown in Section 4.1.1. The all-zero codeword represents the unvoiced state, and is sent if $Vbp_1 \le 0.6$. All 28 codewords with Hamming weight of 1 or 2 are reserved for error protection. The uniform quantizer details are described in Section 4.1.7.

3.2.15 **Gain Quantization.** The two gain values are quantized as follows. G_2 is quantized with a 5-bit uniform quantizer ranging from 10 to 77 dB. G_1 is quantized to 3 bits using the following adaptive algorithm. If G_2 for the current frame is within 5 dB of G_2 for the previous frame, and G_1 is within 3 dB of the average of the G_2 values for the current and previous frames, then the frame is steady-state and a special code (all zero) is sent to indicate that the decoder should set G_1 to the mean of the G_2 values for the current and previous frames. Otherwise, the frame represents a transition and G_1 is quantized with a 7-level uniform quantizer ranging from 6 dB below the minimum of the G_2 values for the current and previous frames to 6 dB above the maximum of those G_2 values. The quantizer range is clamped to 10 and 77 dB. The uniform quantizer details are described in Section 4.1.7. Pseudo code for the adaptive quantization of G_1 is shown below.

```
if (|G2 - G2p| < 5.0 and |G1 - 0.5 *(G2 + G2p)| < 3.0)
    quantizer_index = 0
else
    gain_max = max(G2p, G2) + 6.0
    gain_min = min(G2p, G2) - 6.0
    if (gain_min < 10.0) gain_min = 10.0
    if (gain_max > 77.0) gain_max = 77.0
    quantizer_index values 1 to 7 are determined by quantizing G1 with a 7-level,
    uniform quantizer ranging from gain_min to gain_max
endif
```

3.2.16 **Bandpass Voicing Quantization.** When $Vbp_1 \le 0.6$ (unvoiced), the remaining voicing strengths, Vbp_i, $i = 2, 3, 4, 5$, are quantized to 0. When $Vbp_1 > 0.6$, the remaining voicing strengths are quantized to 1 if their value exceeds 0.6, and quantized to 0 otherwise. There is one exception. If the quantized values of Vbp_i, $i = 2, 3, 4, 5$ are 0001, respectively, then Vbp_5 is quantized to 0.

3.2.17 **Fourier Magnitude Calculation and Quantization.** This analysis measures the Fourier magnitudes of the first 10 pitch harmonics of the prediction residual generated by the quantized prediction coefficients. It uses a 512-point Fast Fourier Transform (FFT) of a 200 sample window centered at the end of the frame. First, a set of quantized predictor coefficients is calculated from the quantized LSF vector (Section 3.2.13). Then the residual window is generated using the quantized prediction coefficients. Next, a 200 sample Hamming window is applied, the signal is zero-padded to 512 points, and the complex FFT is performed. Finally, the complex FFT output is transformed into magnitudes, and the harmonics are found with a spectral peak-picking algorithm.

The peak-picker finds the maximum within a width of $512/\hat{P}_3$ frequency samples centered around the initial estimate for each pitch harmonic, where $\hat{P}_3$ is the quantized pitch. This width is truncated to an integer. The initial estimate for the location of the i^{th} harmonic is $512i/\hat{P}_3$. The number of harmonic magnitudes searched for is limited to the smaller of 10 or $\hat{P}_3/4$. These magnitudes are then normalized to have an RMS value of 1.0. If fewer than 10 harmonics are found, the remaining magnitudes are set to 1.0.

The 10 magnitudes are quantized with an 8-bit vector quantizer. The codebook is searched using a perceptually weighted Euclidean distance, with fixed weights that emphasize low frequencies over higher frequencies. The weights are given by:

$$w_i = \left[\frac{117}{25 + 75\left(1 + 1.4\left(\frac{f_i}{1000}\right)^2\right)^{0.69}}\right]^2, \; i = 1, 2, \ldots, 10, \qquad (10)$$

where $f_i = 8000i/60$ is the frequency in Hz corresponding to the i^{th} harmonic for a default pitch period of 60 samples. The weights are applied to the squared difference between the input Fourier magnitudes and the codebook values.

3.2.18 **Error Protection and Bit Packing.** The table in Section 4.3.2 shows the bit allocation for the MELP coder. To improve performance in channel errors, the unused coder parameters for the unvoiced mode are replaced with forward error correction. Three Hamming (7,4) codes and one Hamming (8,4) code are used. The (7,4) code corrects single bit-errors, while the (8,4) code in addition detects double bit-errors. The (8,4) code is applied to the 4 most significant bits (MSB's) of the first MSVQ index, and the 4 parity bits are written over the bandpass voicing. The remaining 3 bits of the first MSVQ index along with a reserved bit (set to zero), are covered by a (7,4) code with the resulting 3 parity bits written to the MSB's of the Fourier series VQ index. The 4 MSB's of the G_2 codeword are protected with 3 parity bits which are written to the next 3 bits of the Fourier magnitudes. Finally, the LSB of the second gain index and the 3 bit G_1 codeword are protected with 3 parity bits written to the 2 LSBs of the Fourier magnitudes and the aperiodic flag.

The bit transmission order is given in Section 4.3.3.

3.3 **Decoder**

3.3.1 **Bit Unpacking and Error Correction.** The received bits are unpacked from the channel and assembled into the parameter codewords. Parameter decoding is different for voiced and unvoiced modes. The pitch is decoded first, since it contains the mode information. If the pitch code is all-zero or has only one bit set, then the unvoiced mode is used. If two bits are set, a frame erasure is indicated. Otherwise, the pitch value is decoded and the voiced mode is used.

In the unvoiced mode, the (8,4) Hamming code is decoded to correct single bit errors and detect double errors. If an uncorrectable error is detected, a frame erasure is indicated. Otherwise, the (7,4) Hamming codes are decoded, correcting single errors but without double error detection.

If any erasure is detected in the current frame, by the Hamming code, by the pitch code, or directly signaled from the channel, then a frame repeat mechanism is implemented. All of the parameters for the current frame are replaced with the parameters from the previous frame. In addition, the first gain term is set equal to the second gain term so that no gain transitions are allowed.

If an erasure is not indicated, the remaining parameters are decoded. The LSF's are checked for ascending order and minimum separation as described in Section 3.2.13. In the unvoiced mode, default parameter values are used for the pitch, jitter, bandpass voicing, and Fourier magnitudes. The pitch value is set to 50 samples, the jitter is set to 25%, all of the bandpass voicing strengths are set to 0, and the Fourier magnitudes are set to 1. In the voiced mode, Vbp_1 is set to 1; jitter is set to 25% if the aperiodic flag is a 1; otherwise jitter is set to 0%. The bandpass voicing strength for the upper four bands is set to 1 if the corresponding bit is a 1; otherwise the voicing strength is set to 0. There is one exception. If 0001 is received for Vbp_i, $i = 2, 3, 4, 5$, respectively, then Vbp_5 is set to 0.

When the special all-zero code for the first gain parameter, G_1, is received, some errors in the second gain parameter, G_2, can be detected and corrected. This correction process provides improved performance in channel errors. The decoding for the two gain parameters is shown in the following pseudo code.

```
inputs: G1_index, G2_index
outputs: G1, G2
internal: G2p, G2p_error

G2 = decode(G2_index)              32 levels; range: 10 to 77 dB
if (G1_index == 0)                 special G1 code: use mean of G2 and G2p
   if (|G2 - G2p| > 5)             G2_index probably in error
      if (G2p_error == 0)          G2p is correct
         G2 = G2p                  replace the erroneous G2 with past value
      endif
      G2p_error = 1
   else                            G2_index probably correct
      G2p_error = 0
   endif
   G1 = 0.5 * (G2 + G2p)           mean of G2 and G2p
else
   G1 = decode(G1_index)           7 levels; range: min(G2,G2p)-6 to max(G2,G2p)+6
   G2p_error = 0                             (above range is clamped to 10 to 77 dB)
endif
G2p = G2                           save for use as past value
```

3.3.2 **Noise Attenuation.** For quiet input signals, a small amount of gain attenuation is applied to both decoded gain parameters using a power subtraction rule. This attenuation is a simplified, frequency invariant case of the Smoothed Spectral Subtraction noise suppression method [6].

Before determining the attenuation for the first gain term, G_1, a background noise estimate, G_n, is updated as follows. If $G_1 > G_n + C_{up}$ then $G_n = G_n + C_{up}$. If $G_1 < G_n - C_{down}$ then $G_n = G_n - C_{down}$. Otherwise, $G_n = G_1$. $C_{up} = 0.0337435$ and $C_{down} = 0.135418$, so that the noise estimator moves up by 3 dB per second and down by 12 dB per second for the gain update rate of 88.9 updates per second. The noise estimate is clamped between 10 and 80. Noise estimation is disabled for repeated frames to prevent repeated attenuation. The background noise estimate is also used in the adaptive spectral enhancement calculation (Section 3.3.5).

Gain G_1 is then modified by subtracting a (positive) correction term, G_{att}, given in dB by

$$G_{att} = -10\log_{10}(1 - 10^{0.1[G_n + 3 - G_1]}), \tag{11}$$

where G_n is the background noise estimate (in dB), and G_1 is the first gain term (in dB). The correction is clamped to a maximum value of 6 dB to avoid fluctuations and signal distortion. To ensure that the attenuation is applied only to quiet signals, the G_n value as used in Eq. (11) is clamped at an upper limit of 20 dB.

The noise estimation and gain modification steps are then repeated for the second gain term, G_2. Noise estimation and gain attenuation are disabled for repeated frames.

3.3.3 **Parameter Interpolation.** All MELP synthesis parameters are interpolated pitch-synchronously for each synthesized pitch period. The interpolated parameters are the gain (in dB), LSF's, pitch, jitter, Fourier magnitudes, pulse and noise coefficients for mixed excitation, and spectral tilt coefficient for the adaptive spectral enhancement filter. Gain is linearly interpolated between the second gain of the prior frame, G_{2p}, and the first gain of the current frame, G_1, if the starting point, t_0, $t_0 = 0, 1, \ldots, 179$, of the new pitch period is less than 90; otherwise, gain is interpolated between G_1 and G_2. Normally, the other parameters are linearly interpolated between the past and current frame values. The interpolation factor, int, for these parameters is based on the starting point of the new pitch period:

$$int = t_0/180. \tag{12}$$

There are two exceptions to this interpolation procedure. First, if there is an onset with a high pitch frequency, pitch interpolation is disabled and the new pitch is immediately used. This condition is met when G_1 is more than 6 dB greater than G_{2p} and the current frame's pitch period is less than half the prior frame's pitch period. The second exception also involves a gain onset. If G_2 differs from G_{2p} by more than 6 dB, then the LSF's, spectral tilt, and pitch are interpolated using the interpolated gain trajectory as a basis, since the gain is transmitted twice per frame and has a more accurate interpolation path. In this case, the interpolation factor is given by

$$int = \frac{G_{int} - G_{2p}}{G_2 - G_{2p}}, \tag{13}$$

where G_{int} is the interpolated gain. This interpolation factor is then clamped between 0 and 1.

3.3.4 **Mixed Excitation Generation.** The mixed excitation is generated as the sum of the filtered pulse and noise excitations. The pulse excitation, $e_p(n)$, $n = 0, 1, \ldots, T-1$, is computed using an inverse Discrete Fourier Transform of one pitch period in length:

$$e_p(n) = \frac{1}{T}\sum_{k=0}^{T-1} M(k)e^{j2\pi nk/T}. \tag{14}$$

The pitch period, T, is the interpolated pitch value plus the jitter times the pitch, where the jitter is the interpolated jitter strength times the output of a uniform random number generator between -1 and 1. This pitch period is rounded to the nearest integer and clamped between 20 and 160. All of the phases for the pulse excitation are set to zero, hence $M(k)$ is real. Since $e_p(n)$ is real, the magnitudes obey:

$$M(T-k) = M(k),\ k = 1, 2, \ldots, L, \tag{15}$$

where $L = T/2$ if T is even, and $L = (T-1)/2$, if T is odd. The DC term, $M(0)$, is set to 0. Magnitude terms $M(k)$, $k = 1, 2, \ldots, 10$, are set to the interpolated values of the Fourier magnitudes, and any magnitudes not otherwise specified are set to 1. To prevent rapid changes at the start of the pitch period, the pulse excitation is circularly shifted by ten samples of delay so the main excitation pulse occurs at the tenth sample of the period. The pulse is then multiplied by the square root of the pitch to give a unity RMS signal, and then multiplied by 1000 to give a nominal signal level.

The noise is generated by a uniform random number generator with an RMS value of 1000, and range of -1732 to 1732.

The pulse and noise excitation signals are then filtered and summed to form the mixed excitation. The pulse filter for the current frame is given by the sum of all the bandpass filter coefficients for the voiced frequency bands, while the noise filter is given by the sum of the bandpass filter coefficients for the unvoiced bands. These filter coefficients are interpolated pitch synchronously. The bandpass filter coefficients for each of the five bands are given in Appendix A.

3.3.5 **Adaptive Spectral Enhancement.** The adaptive spectral enhancement filter is applied to the mixed excitation signal. This filter is a tenth order pole/zero filter, with an additional first-order tilt compensation. Its coefficients are generated by bandwidth expansion of the linear prediction filter transfer function, $A(z)$, corresponding to the interpolated LSF's. The transfer function of the enhancement filter, $H_{ase}(z)$, is given by:

$$H_{ase}(z) = \frac{A(\alpha z^{-1})}{A(\beta z^{-1})} \cdot (1 + \mu z^{-1}), \tag{16}$$

where

$$\begin{aligned} \alpha &= 0.5p \\ \beta &= 0.8p \end{aligned}, \tag{17}$$

and tilt coefficient μ is first calculated as $max(0.5k_1, 0)$, then interpolated, then multiplied by p, the signal probability. The first reflection coefficient, k_1, is calculated from the decoded LSF's. By the MELP predictor coefficient sign convention, k_1 is usually negative for voiced spectra. The signal probability p is estimated by comparing the current interpolated gain, G_{int}, to the background noise estimate G_n using the formula:

$$p = \frac{G_{int} - G_n - 12}{18}. \tag{18}$$

This signal probability is clamped between 0 and 1.

3.3.6 **Linear Prediction Synthesis.** The synthesis uses a direct form filter, with the coefficients corresponding to the interpolated LSF's.

3.3.7 **Gain Adjustment.** Since the excitation is generated at an arbitrary level, the speech gain must be introduced to the synthesized speech. The correct scaling factor, S_{gain}, is computed for each synthesized pitch period of length T by dividing the desired RMS value (G_{int} must be converted from dB) by the RMS value of the unscaled synthetic speech signal $\hat{s}_n$:

$$S_{gain} = \frac{10^{G_{int}/20}}{\sqrt{\frac{1}{T}\sum_{n=1}^{T} \hat{s}_n^2}}. \tag{19}$$

To prevent discontinuities in the synthesized speech, this scale factor is linearly interpolated between the previous and current values for the first ten samples of the pitch period.

3.3.8 **Pulse Dispersion.** The pulse dispersion filter is a 65th order FIR filter derived from a spectrally-flattened triangle pulse. The coefficients are listed in Appendix B.

3.3.9 **Synthesis Loop Control.** After processing each pitch period, the decoder updates t_0 by adding T, the number of samples in the period just synthesized. If $t_0 < 180$, synthesis for the current frame continues from the parameter interpolation step (Section 3.3.3). Otherwise, the decoder buffers the remainder of the current period which extends beyond the end of the current frame and subtracts 180 from t_0 to produce its initial value next frame.

4. REQUIREMENTS

4.1 Parameter Quantization and Encoding

The MELP parameters which are quantized and transmitted are the final pitch, P_3; the bandpass voicing strengths, Vbp_i, $i = 1, 2, \ldots, 5$; the two gain values, G_1 and G_2; the linear prediction coefficients, a_i, $i = 1, 2, \ldots, 10$; the Fourier magnitudes; and the aperiodic flag. The use of the following quantization procedures is required for interoperability among various implementations.

4.1.1 **Pitch and Overall Voicing.** The final pitch, P_3, and the low band voicing strength, Vbp_1, are quantized jointly using 7 bits, as follows. If $Vbp_1 \le 0.6$, then the frame is unvoiced and the all-zero code is sent. Otherwise, the log of P_3 is quantized with a 99-level uniform quantizer (Section 4.1.7) ranging from $\log 20$ to $\log 160$. The resulting index (range 0 to 98) is then mapped to the transmitted 7-bit codeword using the encode/decode table below. All 28 codes with Hamming weight of 1 or 2 are reserved for error protection. This table is also used in decoding the 7-bit pitch code to determine if a frame is voiced, unvoiced, or whether a frame erasure is indicated. A frame is determined unvoiced if the pitch code is all zero or has only one bit set. If two bits are set, then a frame erasure is indicated. Otherwise, the voiced mode is used and the pitch index is determined from the received code according to the table below.

Code	Index	Code	Index	Code	Index	Code	Index
0x0	UNVOICED	0x20	UNVOICED	0x40	UNVOICED	0x60	ERASURE
0x1	UNVOICED	0x21	ERASURE	0x41	ERASURE	0x61	68
0x2	UNVOICED	0x22	ERASURE	0x42	ERASURE	0x62	69
0x3	ERASURE	0x23	16	0x43	42	0x63	70
0x4	UNVOICED	0x24	ERASURE	0x44	ERASURE	0x64	71
0x5	ERASURE	0x25	17	0x45	43	0x65	72
0x6	ERASURE	0x26	18	0x46	44	0x66	73
0x7	0	0x27	19	0x47	45	0x67	74
0x8	UNVOICED	0x28	ERASURE	0x48	ERASURE	0x68	75
0x9	ERASURE	0x29	20	0x49	46	0x69	76
0xA	ERASURE	0x2A	21	0x4A	47	0x6A	77
0xB	1	0x2B	22	0x4B	48	0x6B	78
0xC	ERASURE	0x2C	23	0x4C	49	0x6C	79
0xD	2	0x2D	24	0x4D	50	0x6D	80
0xE	3	0x2E	25	0x4E	51	0x6E	81
0xF	4	0x2F	26	0x4F	52	0x6F	82
0x10	UNVOICED	0x30	ERASURE	0x50	ERASURE	0x70	83
0x11	ERASURE	0x31	27	0x51	53	0x71	84
0x12	ERASURE	0x32	28	0x52	54	0x72	85
0x13	5	0x33	29	0x53	55	0x73	86
0x14	ERASURE	0x34	30	0x54	56	0x74	87
0x15	6	0x35	31	0x55	57	0x75	88
0x16	7	0x36	32	0x56	58	0x76	89
0x17	8	0x37	33	0x57	59	0x77	90
0x18	ERASURE	0x38	34	0x58	60	0x78	91
0x19	9	0x39	35	0x59	61	0x79	92
0x1A	10	0x3A	36	0x5A	62	0x7A	93
0x1B	11	0x3B	37	0x5B	63	0x7B	94
0x1C	12	0x3C	38	0x5C	64	0x7C	95
0x1D	13	0x3D	39	0x5D	65	0x7D	96
0x1E	14	0x3E	40	0x5E	66	0x7E	97
0x1F	15	0x3F	41	0x5F	67	0x7F	98

4.1.2 **Bandpass Voicing.** When $Vbp_1 > 0.6$, the remaining bandpass voicing strengths are quantized to 1 if their value exceeds 0.6, and quantized to 0 otherwise. There is one exception. If the quantized values of Vbp_i, $i = 2, 3, 4, 5$ are 0001, respectively, then Vbp_5 is quantized to 0. The quantized values are transmitted. When $Vbp_1 \le 0.6$, the bandpass voicing bits are replaced with FEC parity bits.

4.1.3 **Gain.** Two gain parameters, G_1 and G_2, are transmitted each frame. G_2 is quantized using a 32-level uniform quantizer ranging from 10.0 to 77.0 dB. The quantizer index is the transmitted codeword. G_1 is quantized to 3 bits using the adaptive algorithm described in Section 3.2.15. This algorithm determines if the frame is a steady state frame or a transition frame. The all-zero codeword is sent for steady state frames and a 7-bit uniform quantizer is used for transition frames. In this case, the quantizer index plus 1 is the transmitted codeword. Section 4.1.7 describes uniform quantizer operation.

4.1.4 **Linear Prediction Coefficients.** The linear prediction coefficients are converted into LSF's and then quantized by an MSVQ, as described in Section 3.2.13. The MSVQ codebook consists of four stages whose indices have 7, 6, 6, and 6 bits, respectively. The quantized LSF vector is the sum of the vectors selected by the search process, with one vector selected from each stage. The indices of those four vectors are transmitted. The code vectors and corresponding indices are given in Appendix C.

4.1.5 **Fourier Magnitudes.** The ten Fourier magnitudes are coded with an 8-bit vector quantizer. The index of the code vector which minimizes the weighted Euclidean distance between the input and code vectors is transmitted. The weights are fixed and are given by Eq. (10). The code vectors and corresponding indices are given in Appendix D.

4.1.6 **Aperiodic Flag.** The aperiodic flag is a single bit, transmitted as is. Section 3.2.5 describes this flag.

4.1.7 **Uniform Quantization.** The pitch and gain quantization processes employ uniform quantizers which operate as follows. The stepsize for an n-level quantizer ranging from x_1 to x_2 is $s = (x_2 - x_1)/(n-1)$. The n quantizer output values are $x_1 + i \cdot s$, $i = 0, 1, \ldots, n-1$. The threshold values between levels i and $i+1$ are $x_1 + (0.5+i)s$, $i = 0, 1, \ldots, n-2$. The quantizer produces n indices, $0, 1, \ldots, n-1$, which correspond to an increasing value of the parameter being quantized. For example, let $x_1 = 1$, $x_2 = 7$, and $n = 7$. This gives $s = 1$, levels of $1, 2, \ldots, 7$, and thresholds of $1.5, 2.5, \ldots, 6.5$. Index 0 is assigned to input values x, for which $x \le 1.5$; index 1 is assigned to input values for which $1.5 < x \le 2.5$; etc.

4.2 **Error Protection**

Forward Error Correction (FEC) is implemented in the unvoiced mode only, when the Fourier magnitudes, bandpass voicing, and jitter bits need not be transmitted. FEC replaces those 13 bits with the parity bits of three Hamming (7,4) codes and one Hamming (8,4) code. These codes protect the first stage LSF index (7 bits) and both gain indices (8 bits); there is one spare information bit, set to 0.

The protected bits are placed into a column vector, $\boldsymbol{u}$, which post-multiplies the parity generator matrix to produce the n-bit parity vector, $\boldsymbol{p} = [p_0 \;\; p_1 \;\; \ldots \;\; p_{n-1}]^T$, where n is 3 or 4. The parity generator matrix for the Hamming (7,4) code is:

$$G_{7,4} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{bmatrix}. \tag{20}$$

The parity generator matrix for the Hamming (8,4) code is:

$$G_{8,4} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}. \tag{21}$$

The 4 MSB's of the first stage LSF index ($\boldsymbol{u} = [b_6 \;\; b_5 \;\; b_4 \;\; b_3]^T$) are protected by the (8,4) code, with the 4 parity bits written to the LSBs of the bandpass voicing index ($p_0 \;\; p_1 \;\; p_2 \;\; p_3$). The remaining 3 bits of the first stage index and the spare bit ($\boldsymbol{u} = [b_2 \;\; b_1 \;\; b_0 \;\; 0]^T$), are protected with 3 parity bits written to the MSB's of the Fourier magnitude index ($p_0 \;\; p_1 \;\; p_2$). The 4 MSB's of the second gain index ($\boldsymbol{u} = [b_4 \;\; b_3 \;\; b_2 \;\; b_1]^T$) are protected with 3 parity bits written to the next 3 bits of the Fourier magnitude index ($p_0 \;\; p_1 \;\; p_2$). The LSB of the second gain index and the 3 bit first gain index ($\boldsymbol{u} = [b_0 \;\; b_2 \;\; b_1 \;\; b_0]^T$) are protected with 3 parity bits written to the 2 LSBs of the Fourier magnitude index ($p_0 \;\; p_1$) and the aperiodic flag (p_2). The parenthesized groups of parity bits show their placement in the given index, with the right-most bit having the least significance.

4.3 **Transmission Format**

4.3.1 **Transmission Rate.** The transmission rate shall be 2,400 bits/s ±0.01 percent. Since all frames contain 54 bits, the frame length is 22.5 ms ±0.01 percent.

4.3.2 **Bit Allocation.** The following table shows how the 54 bits in an MELP frame are allocated among the parameters.

Parameters	Voiced	Unvoiced
LSF's	25	25
Fourier Magnitudes	8	-
Gain (2 per frame)	8	8
Pitch, overall voicing	7	7
Bandpass Voicing	4	-
Aperiodic Flag	1	-
Error Protection	-	13
Sync Bit	1	1
Total Bits / 22.5 ms Frame	54	54

4.3.3 **Bit Transmission Order.** The following table shows the transmission order for the 54 bits in each MELP frame for both voiced and unvoiced frames. The sync bit alternates between 0 and 1 from frame to frame.

Bit	Voiced	Unvoiced	Bit	Voiced	Unvoiced	Bit	Voiced	Unvoiced
1	G(2)-1	G(2)-1	19	**LSF(1)-7**	LSF(1)-7	37	G(1)-1	G(1)-1
2	BP-1	FEC(1)-1	20	LSF(4)-6	LSF(4)-6	38	BP-3	FEC(1)-3
3	P-1	P-1	21	**P-4**	P-4	39	BP-2	FEC(1)-2
4	LSF(2)-1	LSF(2)-1	22	**LSF(1)-6**	LSF(1)-6	40	LSF(2)-2	LSF(2)-2
5	LSF(3)-1	LSF(3)-1	23	**LSF(1)-5**	LSF(1)-5	41	LSF(3)-4	LSF(3)-4
6	**G(2)-4**	G(2)-4	24	**LSF(2)-6**	LSF(2)-6	42	**LSF(2)-3**	LSF(2)-3
7	**G(2)-5**	G(2)-5	25	**BP-4**	FEC(1)-4	43	LSF(3)-3	LSF(3)-3
8	LSF(3)-6	LSF(3)-6	26	**LSF(1)-4**	LSF(1)-4	44	LSF(3)-2	LSF(3)-2
9	**G(2)-2**	G(2)-2	27	**LSF(1)-3**	LSF(1)-3	45	LSF(4)-4	LSF(4)-4
10	**G(2)-3**	G(2)-3	28	**LSF(2)-5**	LSF(2)-5	46	LSF(4)-3	LSF(4)-3
11	**P-5**	P-5	29	LSF(4)-5	LSF(4)-5	47	**AF**	FEC(4)-3
12	LSF(3)-5	LSF(3)-5	30	FM-1	FEC(4)-1	48	LSF(4)-2	LSF(4)-2
13	**P-6**	P-6	31	**LSF(1)-2**	LSF(1)-2	49	FM-5	FEC(3)-3
14	**P-2**	P-2	32	**LSF(2)-4**	LSF(2)-4	50	FM-4	FEC(3)-2
15	**P-3**	P-3	33	FM-8	FEC(2)-3	51	FM-3	FEC(3)-1
16	LSF(4)-1	LSF(4)-1	34	FM-7	FEC(2)-2	52	FM-2	FEC(4)-2
17	**P-7**	P-7	35	FM-6	FEC(2)-1	53	G(1)-3	G(1)-3
18	**LSF(1)-1**	LSF(1)-1	36	G(1)-2	G(1)-2	54	**SYNC**	SYNC

NOTES: G = Gain
P = Pitch/Voicing
FEC = Forward Error Correction Parity Bits
Bit 1 = least significant bit of data set
Highlighted Bits = 24 Most Significant MELP Bits
BP = Bandpass Voicing
LSF = Line Spectral Frequencies
FM = Fourier Magnitudes
AF = Aperiodic Flag

4.4 **Performance Verification**

A new implementation of the MELP coder must be tested to verify that its performance meets or exceeds that of the MELP reference coder. A new implementation must also meet the same performance standards when interoperating with the MELP reference coder. The testing is accomplished through formal evaluation of intelligibility and quality or by showing bit equivalence between the new implementation and a verified MELP implementation. Both verification methods evaluate an implementation over an extensive set of conditions.

4.4.1 **Formal Evaluation.** The intelligibility of a new implementation is evaluated using the Diagnostic Rhyme Test (DRT). DRT performance thresholds have been set and are discussed later in this section. Quality is evaluated using a direct paired forced choice comparison test, i.e., an A/B test. Performance thresholds for the quality tests have also been set and are based on the percent preference for the new implementation over the MELP reference coder. The Federal Standard CELP coder is also included in the quality test to broaden the context of the test and to include a known difference from MELP. Table 1 summarizes the coder configurations which are evaluated for intelligibility and quality. In Table 1, the “Implementation → MELP Reference” and “MELP Reference → Implementation” cases are “cross-coder” configurations which test interoperability. Table 2 shows the intelligibility and quality test conditions. Six talkers (3 male, 3 female) are used for each condition.

Intelligibility (Encoder → Decoder)	**Quality (Encoder → Decoder)**
Implementation → Implementation	Implementation → Implementation
Implementation → MELP Reference	Implementation → MELP Reference
MELP Reference → Implementation	MELP Reference → Implementation
	MELP Reference → MELP Reference
	CELP → CELP

Table 1: Tested Coder Configurations

Intelligibility Condition (microphone)	**Quality Condition (microphone)**
Quiet (Dynamic)	Quiet (Dynamic)
Quiet (H250)	Quiet (H250)
Office (STU-III)	Office (STU-III)
E3A AWACS (R215)	E3A AWACS (R215)
HMMWV (H250)	HMMWV (H250)
MCE Field Shelter (M87)	MCE Field Shelter (M87)
Car (STU-III cellular)	Car (STU-III cellular)
0.5% Block Error Rate in Quiet (Dynamic)	0.5% Block Error Rate in Quiet (Dynamic)
1.0% Bit Error Rate in Quiet (Dynamic)	1.0% Bit Error Rate in Quiet (Dynamic)
CVSD → Coder in Quiet (Dynamic)	CVSD → Coder in Quiet (Dynamic)
CVSD → Coder → CVSD in Quiet (Dynamic)	CVSD → Coder → CVSD in Quiet (Dynamic)
M2 Bradley (M138)	
CH47 (M87)	
F-15 (M101)	
P3C Orion (EV985)	

Table 2: Intelligibility and Quality Test Conditions

4.4.1.1 **Intelligibility Tests.** The Diagnostic Rhyme Test will be performed in accordance with ANSI standard S3.2-1989 and will be scored with eight listeners. The combined talker score determined by the test lab must meet or exceed the corresponding threshold score for each condition and for the weighted combination of all conditions. Table 3 shows the weight and threshold score for each condition. The implementation's combined score is calculated by multiplying its individual score for each condition by the corresponding weight given in Table 3, and then summing the results. This process is used for each of the three coder configurations evaluated for intelligibility. The threshold score for each condition is based on a one-tail 99.5% confidence interval; the combined threshold score is based on a one-tail 99% confidence interval.

Intelligibility Condition (microphone)	Weight	Threshold Score
Quiet (Dynamic)	0.100	90.80
Quiet (H250)	0.100	88.77
Office (STU-III)	0.0667	89.34
E3A AWACS (R215)	0.0667	85.10
HMMWV (H250)	0.0667	61.88
MCE Field Shelter (M87)	0.0667	87.28
Car (STU-III cellular)	0.0667	82.48
M2 Bradley (M138)	0.0667	61.33
CH47 (M87)	0.0667	63.27
F-15 (M101)	0.0667	75.06
P3C Orion (EV985)	0.0667	82.98
0.5% Block Error Rate in Quiet (Dynamic)	0.050	89.58
1.0% Bit Error Rate in Quiet (Dynamic)	0.050	87.47
CVSD → Coder in Quiet (Dynamic)	0.050	84.07
CVSD → Coder → CVSD in Quiet (Dynamic)	0.050	81.33
Combined (weighted)	1.000	82.634

Table 3: Weights and Thresholds for Intelligibility Conditions

4.4.1.2 **Quality Tests.** The quality of the three coder configurations involving the new implementation (see Table 1) is compared with the quality of the MELP reference coder using a direct paired forced choice comparison (an A/B) test. Quality is measured in each condition by the percent preference taken over all talkers and listeners. For each coder configuration involving the new implementation, the percent preferred must meet or exceed the threshold for each condition and for the weighted combination of all conditions. Reference [7] gives an overview of the A/B test methodology, and reference [8] contains a recommended test plan.

The threshold for individual conditions is 45.84%, i.e., each coder configuration involving the new implementation must have a preference percentage of 45.84% or more in each condition. This threshold is equivalent to saying that the new implementation must be selected at least 440 times out of the 960 comparisons between the new implementation and the reference in each condition. The threshold is based on a one-tail 99.5% confidence interval.

The combined percent preferred is calculated for each of the three coder configurations by summing the weighted individual condition percentages using the weights listed in Table 4. The combined threshold percentage is 48.71%, i.e., each of the three coder configurations involving the new implementation must have a combined preference percentage of 48.71% or more. The combined threshold is based on a one-tail 99% confidence interval.

Quality Condition (microphone)	Weight
Quiet (Dynamic)	0.245
Quiet (H250)	0.105
Office (STU-III)	0.070
E3A AWACS (R215)	0.070
HMMWV (H250)	0.070
MCE Field Shelter (M87)	0.070
Car (STU-III cellular)	0.070
0.5% Block Error Rate in Quiet (Dynamic)	0.075
1.0% Bit Error Rate in Quiet (Dynamic)	0.075
CVSD → Coder in Quiet (Dynamic)	0.075
CVSD → Coder → CVSD in Quiet (Dynamic)	0.075

Table 4: Weights for Quality Conditions

4.4.2 **Bit Equivalence.** A lower cost alternative for implementation verification is by demonstrating that the new implementation is bit equivalent to a verified MELP implementation. Bit equivalent means that given the same input, the new implementation's encoder produces the same bitstream as produced by the encoder of the verified MELP implementation and that given the same bitstream, the new implementation's decoder produces the same 16 bit PCM samples as produced by the decoder of the verified MELP implementation. This equivalence must be shown for all intelligibility material and quality material used in the formal evaluation.

5. GOVERNMENT CONTACT POINT

Electronic versions of the coding tables, digital recordings of the test input speech and processed reference speech, software for the MELP reference coder, software for error introduction, interim software for a KY57 equivalent CVSD coder, and reference [8], "MELP Performance Verification Method," can be obtained through the government at: (to be determined)

ORGANIZATION

ADDRESS

STATE and ZIP

6. REFERENCES

[1] A. V. McCree and T. P. Barnwell III, "A Mixed Excitation LPC Vocoder Model for Low Bit Rate Speech Coding," *IEEE Transactions on Speech and Audio Processing*, Vol. 3, No. 4, July 1995, pp. 242-250.

[2] A. McCree, K. Truong, E. B. George, T. P. Barnwell, and V. Viswanathan, "A 2.4 kbits/s MELP Coder Candidate for the New U.S. Federal Standard," *Proceedings of IEEE ICASSP 1996*, pp. 200-203.

[3] Y. Medan, E. Yair, and D. Chazan, "Super Resolution Pitch Determination of Speech Signals," *IEEE Transactions on Signal Processing*, Vol. 39, No. 1, January 1991, pp. 40-48.

[4] P. Kabal and R. P. Ramachandran, "The Computation of Line Spectral Frequencies Using Chebyshev Polynomials," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-34, No. 6, December 1986, pp. 1419-1426.

[5] W. P. LeBlanc, B. Bhattacharya, S. A. Mahmoud, and V. Cuperman, "Efficient Search and Design Procedures for Robust Multi-Stage VQ of LPC Parameters for 4 kb/s Speech Coding," *IEEE Trans-*

actions on Speech and Audio Processing, Vol. 1, No. 4, October 1993, pp. 373-385.

[6] L. Arslan, A. McCree, and V. Viswanathan, “New Methods for Adaptive Noise Suppression,” *Proceedings of IEEE ICASSP 1995*, pp. 812-815.

[7] IEEE Subcommittee on Subjective Measurements, “IEEE Recommended Practice for Speech Quality Measurements,” *IEEE Transactions on Audio and Electroacoustics*, Vol. 17, 1969, pp. 227-246.

[8] “MELP Performance Verification Method”, U.S. Department of Defense Digital Voice Processing Consortium, 1997.

Appendix A

This appendix lists the coefficients of the five bandpass filters used in the decoder.

0 - 500 Hz	500 - 1000 Hz	1000 - 2000 Hz	2000 - 3000 Hz	3000 - 4000 Hz
-0.00302890	-0.00249420	-0.00231491	0.00231491	0.00554149
-0.00701117	-0.00282091	0.00990113	0.00990113	-0.00981750
-0.01130619	0.00257679	0.02086129	-0.02086129	0.00856805
-0.01494082	0.01451271	-0.00000000	0.00000000	-0.00000000
-0.01672586	0.02868120	-0.03086123	0.03086123	-0.01267517
-0.01544189	0.03621179	-0.02180695	-0.02180695	0.02162277
-0.01006619	0.02784469	0.00769333	-0.00769333	-0.01841647
0.00000000	0.00000000	-0.00000000	-0.00000000	0.00000000
0.01474923	-0.04079870	-0.01127245	0.01127245	0.02698425
0.03347158	-0.07849207	0.04726837	0.04726837	-0.04686914
0.05477206	-0.09392213	0.10106105	-0.10106105	0.04150730
0.07670890	-0.07451087	-0.00000000	0.00000000	-0.00000000
0.09703726	-0.02211575	-0.17904543	0.17904543	-0.07353666
0.11352143	0.04567473	-0.16031428	-0.16031428	0.15896026
0.12426379	0.10232715	0.09497157	-0.09497157	-0.22734513
0.12799355	0.12432599	0.25562154	0.25562154	0.25346255
0.12426379	0.10232715	0.09497157	-0.09497157	-0.22734513
0.11352143	0.04567473	-0.16031428	-0.16031428	0.15896026
0.09703726	-0.02211575	-0.17904543	0.17904543	-0.07353666
0.07670890	-0.07451087	-0.00000000	0.00000000	-0.00000000
0.05477206	-0.09392213	0.10106105	-0.10106105	0.04150730
0.03347158	-0.07849207	0.04726837	0.04726837	-0.04686914
0.01474923	-0.04079870	-0.01127245	0.01127245	0.02698425
0.00000000	-0.00000000	-0.00000000	-0.00000000	0.00000000
-0.01006619	0.02784469	0.00769333	-0.00769333	-0.01841647
-0.01544189	0.03621179	-0.02180695	-0.02180695	0.02162277
-0.01672586	0.02868120	-0.03086123	0.03086123	-0.01267517
-0.01494082	0.01451271	-0.00000000	0.00000000	-0.00000000
-0.01130619	0.00257679	0.02086129	-0.02086129	0.00856805
-0.00701117	-0.00282091	0.00990113	0.00990113	-0.00981750
-0.00302890	-0.00249420	-0.00231491	0.00231491	0.00554149

Appendix B

This appendix lists the impulse response of the pulse dispersion filter.

Samples 1-13	Samples 14-26	Samples 27-39	Samples 40-52	Samples 53-65
-0.17304259	0.24325127	0.07343483	0.02968464	0.00019707
-0.01405709	-0.01767043	-0.00518645	-0.01247640	-0.02825247
0.01224406	-0.00018612	0.01298488	0.01854666	0.01720989
0.11364226	0.05869485	0.02928440	0.00076184	-0.06004292
0.00198199	-0.00327456	-0.01989405	-0.07749640	-0.07076744
0.00000658	0.00607395	0.01216758	0.01244697	0.00914347
0.04529633	0.02753924	0.01180979	-0.02721777	0.06082730
-0.00092027	-0.03351673	-0.38924775	0.07266098	0.01805528
-0.00103078	0.00602189	0.00720325	0.00472008	-0.00318634
0.02552787	0.01436539	-0.01154561	0.03526439	0.03444110
-0.06339257	0.82854582	0.08426287	0.02674603	0.00026302
-0.00122031	0.00033165	-0.00355720	-0.00744038	-0.01053809
0.01412525	-0.00360180	0.02151233	0.02582623	0.02165922

Appendix C

This appendix lists the codebooks used by the LSF multi-stage vector quantizer. The four stages have 128, 64, 64, and 64 levels, respectively, and are listed in order. The component values are in Hertz.

Stage 1:

Index	Vector									
0x0	355.243052,	492.660028,	635.385928,	980.347948,	1837.261588,	2144.996488,	2413.630272,	2740.714360,	3093.750680,	3368.605604,
0x1	484.729576,	640.392668,	823.554944,	1338.889972,	1880.237972,	2096.149748,	2435.878020,	2765.302236,	3111.357628,	3418.139924,
0x2	436.939032,	581.990372,	733.911476,	1083.060744,	1722.101828,	1952.692520,	2262.632928,	2665.653132,	2988.244500,	3298.130992,
0x3	350.087600,	694.610176,	944.914844,	1378.402864,	1707.342116,	2041.440136,	2412.922496,	2762.516244,	3162.275016,	3440.821584,
0x4	269.325932,	388.049792,	556.748868,	785.832892,	1397.348004,	1900.273176,	2320.491456,	2646.296664,	3000.054600,	3327.772772,
0x5	419.087356,	620.798580,	850.207244,	1116.317652,	1781.215188,	2232.204928,	2499.593164,	2780.066296,	3175.151244,	3383.999004,
0x6	358.942452,	489.979380,	640.111288,	1017.602192,	1632.583372,	1860.474172,	2231.257076,	2530.919888,	2914.448960,	3401.586484,
0x7	342.968080,	472.778640,	726.418032,	1259.413024,	1498.007664,	1891.490164,	2466.558164,	2714.287832,	3160.189944,	3419.703608,
0x8	335.018920,	454.914768,	599.867328,	961.710264,	1759.619356,	2105.336600,	2379.626816,	2706.222688,	3035.591784,	3319.356816,
0x9	442.061924,	553.856784,	797.270016,	1305.514352,	1865.663680,	2073.685500,	2435.956032,	2730.036752,	3100.078344,	3403.177804,
0xa	374.716320,	513.719596,	672.660540,	1007.318856,	1678.355252,	1896.143836,	2248.388928,	2580.640296,	3008.824168,	3367.792428,
0xb	342.836124,	602.011044,	806.724744,	1354.434868,	1736.211376,	1954.730524,	2467.389528,	2745.001960,	3164.511124,	3452.871652,
0xc	257.766340,	414.160796,	594.258332,	830.760316,	1054.375184,	1916.648492,	2340.302436,	2672.872980,	3199.556716,	3428.415672,
0xd	292.366648,	429.559640,	614.610428,	820.622344,	1351.086544,	2055.154672,	2396.444508,	2694.685012,	3094.502340,	3359.889892,
0xe	298.311292,	453.999408,	578.280580,	905.030628,	1527.175416,	1719.933116,	2049.377252,	2579.202976,	2946.545800,	3302.782036,
0xf	336.895904,	482.206380,	640.568984,	999.175084,	1291.378696,	1532.487800,	2241.188304,	2554.378500,	2930.959524,	3361.901736,
0x10	258.053976,	440.262800,	705.940108,	988.153356,	1744.921772,	2026.777644,	2372.851292,	2688.188712,	3118.794236,	3391.818028,
0x11	471.585160,	620.421768,	768.090716,	1082.870284.	1751.583152,	1997.883516,	2293.602428,	2743.392004,	3082.614040,	3332.107592,
0x12	366.246760,	496.800000,	672.316592,	1165.549716,	1447.016836,	1687.916328,	2294.844616,	2547.156980,	3219.253312,	3477.200276,
0x13	510.944264,	676.518572,	826.792700,	1134.333084,	1402.617564,	1643.895904,	2356.263864,	2667.139968,	3011.790276,	3369.347468,
0x14	255.724280,	387.165300,	573.545624,	785.894372,	1066.265316,	1757.711896,	2263.158644,	2646.989552,	3135.291852,	3456.890708,
0x15	378.012868,	501.412328,	704.314976,	917.300392,	1163.199952,	2029.238448,	2375.580324,	2680.752024,	3173.105664,	3386.439868,
0x16	363.972960,	495.552612,	721.780700,	993.407816,	1225.042276,	1462.770692,	1739.992988,	2622.551036,	3204.830068,	3432.341068,
0x17	464.747332,	611.589924,	768.577188,	1014.013508,	1220.578772,	1596.409624,	2440.187768,	2705.814872,	3152.642576,	3428.078368,
0x18	302.335364,	477.008004,	604.607176,	926.489164,	1569.256396,	1809.041736,	2089.583652,	2607.175324,	3043.296900,	3268.448912,
0x19	430.245116,	557.748552,	772.076800,	1037.343496,	1262.090300,	1915.895200,	2312.176824,	2583.310112,	3099.906888,	3325.009868,
0x1a	349.410284,	560.269912,	751.892412,	993.639536,	1507.132780,	1755.265904,	2086.514692,	2440.080336,	2858.434696,	3309.377008,
0x1b	427.173948,	653.323548,	827.836272,	1096.531816,	1318.221064,	1677.783892,	2245.503596,	2519.509312,	3076.545064,	3360.555120,
0x1c	299.500472,	447.090812,	602.425292,	837.466012,	1040.742628,	1548.003748,	2279.710052,	2618.330652,	3105.809148,	3444.257700,
0x1d	328.461056,	462.874748,	624.498484,	865.029756,	1068.647964,	1547.774676,	2473.663648,	2745.927484,	3154.435812,	3441.285648,
0x1e	324.703880,	468.558984,	661.255136,	962.651128,	1182.703040,	1462.208432,	1741.390388,	2450.460364,	3083.241456,	3370.025988,
0x1f	348.534068,	497.361908,	665.076580,	927.130960,	1152.398736,	1384.022008,	2206.452184,	2798.957392,	3143.444020,	3404.614276,
0x20	282.540208,	399.817980,	600.125840,	1360.887924,	2103.622308,	2286.514276,	2588.754368,	2874.624788,	3225.367160,	3467.056508,
0x21	412.601996,	656.294856,	1113.328984,	1647.403296,	2055.375060,	2312.331804,	2665.064932,	2918.390864,	3223.101592,	3484.878096,
0x22	311.996136,	424.640628,	677.403332,	1577.771676,	1868.681796,	2082.825792,	2435.651552,	2698.612244,	3214.556028,	3409.172052,
0x23	419.770844,	677.164136,	1019.950432,	1497.651320,	1755.500792,	2131.790972,	2580.584380,	2904.392236,	3308.333156,	3532.643648,

0x24	259.509376,	385.637168,	554.562836,	937.909320,	1995.858280,	2321.667736,	2564.317268,	2836.416264,	3141.165228,	3417.669004,
0x25	259.767772,	371.811012,	613.365424,	1265.584772,	2012.314700,	2213.325596,	2549.977904,	2808.083496,	3187.912768,	3452.612372,
0x26	226.714780,	349.493580,	608.060808,	1292.857408,	1544.272080,	1868.756264,	2388.450908,	2647.670600,	3142.029760,	3437.458604,
0x27	369.138240,	511.207808,	754.415676,	1319.551916,	1697.700984,	1916.897192,	2447.294220,	2730.763144,	3177.868008,	3446.004236,
0x28	252.435392,	381.234476,	584.820988,	1159.922460,	1898.554296,	2115.347160,	2430.113732,	2702.711664,	3115.573444,	3405.707176,
0x29	319.502340,	436.349812,	863.773064,	1682.395072,	1961.092860,	2203.463320,	2567.750612,	2839.694116,	3219.440192,	3422.168484,
0x2a	331.862240,	446.244828,	660.918160,	1327.774516,	1694.060384,	1900.696384,	2279.667660,	2520.622328,	3213.492012,	3446.511648,
0x2b	400.106248,	561.949040,	934.342308,	1358.145284,	1622.604404,	2013.068024,	2389.652200,	2769.070912,	3130.311280,	3413.614884,
0x2c	242.657020,	377.248792,	522.490964,	846.946680,	1761.833916,	2273.529980,	2545.114128,	2796.080976,	3106.364252,	3351.943212,
0x2d	219.575956,	360.996896,	548.251128,	1088.434748,	1850.457128,	2070.726688,	2414.144860,	2685.428328,	3128.683452,	3450.302588,
0x2e	267.708968,	412.078272,	563.674080,	1070.274400,	1466.291016,	1693.365224,	2236.434108,	2558.971412,	2977.276580,	3294.566092,
0x2f	280.886640,	416.620632,	598.407264,	1135.358508,	1546.804680,	1744.766052,	2447.398760,	2759.398560,	3091.181864,	3426.265432,
0x30	262.027288,	382.135568,	656.530136,	1474.423620,	1840.272900,	2051.403696,	2404.401200,	2679.250592,	3166.215332,	3389.927400,
0x31	261.244876,	531.776124,	972.450092,	1439.603064,	1863.321332,	2137.972540,	2506.207492,	2793.512032,	3204.078640,	3486.612756,
0x32	339.674328,	453.648992,	751.280204,	1389.907636,	1635.730336,	1865.774876,	2173.773284,	2553.385268,	3065.628208,	3273.235436,
0x33	341.587388,	661.908516,	1017.171716,	1394.366888,	1663.345632,	1920.173796,	2203.118924,	2548.413908,	2988.712388,	3297.386536,
0x34	230.146968,	343.667980,	551.302016,	859.858328,	1580.424940,	1832.179784,	2240.921664,	2617.020472,	3037.238376,	3423.404416,
0x35	235.039356,	364.897428,	689.403960,	1029.605932,	1665.887764,	1948.938724,	2346.250084,	2648.456792,	3069.623524,	3389.224348,
0x36	308.913508,	449.624692,	663.772816,	1145.008812,	1351.964772,	1598.142492,	1852.252004,	2454.309724,	2963.855000,	3252.933440,
0x37	257.909564,	462.387712,	741.203612,	1052.100868,	1358.228760,	1622.148340,	2351.586768,	2675.735168,	3092.801340,	3413.616912,
0x38	245.750260,	376.103252,	575.477096,	922.024244,	1670.456784,	1903.208808,	2267.835800,	2608.720112,	3005.315992,	3260.555008,
0x39	247.920760,	379.176512,	646.023984,	1306.141840,	1522.275312,	2009.603136,	2373.209384,	2678.976628,	3272.521600,	3480.612912,
0x3a	252.521964,	397.542288,	729.733388,	1092.573888,	1541.289920,	1831.638300,	2172.991212,	2432.845540,	2863.515608,	3353.231332,
0x3b	293.760896,	453.119144,	952.543932,	1252.249068,	1524.138240,	1854.489388,	2240.400708,	2529.295340,	2980.645980,	3332.247948,
0x3c	194.884480,	303.136856,	524.001072,	885.984548,	1473.173076,	1919.383060,	2355.834416,	2617.938748,	3072.727588,	3484.182104,
0x3d	221.380896,	339.827248,	612.828448,	974.216724,	1296.283708,	1802.879000,	2241.073592,	2652.630088,	3095.005464,	3428.844228,
0x3e	346.441260,	477.961284,	659.909792,	1048.696260,	1321.778992,	1560.564952,	1860.835916,	2137.948484,	3028.669084,	3357.727232,
0x3f	306.865700,	452.086676,	606.287508,	958.052088,	1271.102660,	1499.649856,	2168.586976,	2510.451536,	2942.205140,	3376.739692,
0x40	367.577380,	588.407212,	947.277508,	1401.718148,	1710.042324,	2095.772072,	2530.043540,	2877.200112,	3307.829148,	3539.939676,
0x41	528.676400,	819.913676,	1280.496720,	1637.511820,	2066.765212,	2453.835716,	2824.688768,	3094.627148,	3371.254420,	3558.843104,
0x42	545.224416,	669.303732,	909.509712,	1447.146740,	1725.206608,	1977.595112,	2404.918460,	2678.227616,	3059.974176,	3374.444120,
0x43	502.175120,	755.642512,	1138.330556,	1488.741484,	1860.131964,	2268.931212,	2653.297860,	2991.633600,	3339.490956,	3557.739556,
0x44	298.315792,	427.893476,	578.377140,	1139.800860,	1526.635152,	1746.979136,	2250.458972,	2556.079216,	2919.105144,	3394.443520,
0x45	265.345712,	679.203144,	917.211964,	1262.558816,	1777.907428,	2105.336432,	2492.856668,	2845.667064,	3222.134752,	3475.157160,
0x46	459.013600,	591.152844,	773.066500,	1211.262108,	1477.162912,	1731.845324,	2117.279752,	2387.897856,	3042.563168,	3318.468632,
0x47	465.500052,	607.671140,	777.227012,	1332.425448,	1605.648308,	1856.328328,	2352.959464,	2590.842552,	3102.572804,	3381.365096,
0x48	362.053116,	490.137184,	745.960236,	1269.985928,	1492.284812,	2059.254896,	2441.930800,	2750.527340,	3227.946188,	3436.200692,
0x49	421.179164,	630.471988,	1028.686908,	1430.931296,	1863.460944,	2272.510540,	2652.559160,	2934.399884,	3308.053440,	3523.167936,
0x4a	429.689288,	559.663168,	757.935192,	1288.938980,	1578.246504,	1827.549852,	2323.808836,	2595.758796,	3042.781576,	3317.780068,
0x4b	483.431132,	665.740556,	914.168828,	1187.178920,	1730.056668,	2160.592152,	2538.107676,	2833.483468,	3214.963528,	3447.815624,
0x4c	251.069756,	400.132356,	683.248580,	1007.969788,	1282.548200,	1622.331828,	2215.831796,	2558.840008,	3112.782068,	3417.389900,
0x4d	320.443576,	444.544064,	670.428836,	1086.773516,	1537.545492,	1759.446916,	2524.721292,	2806.092276,	3163.121480,	3453.870028,

0x4e	425.424868,	566.142816,	749.368748,	1166.345352,	1466.362360,	1699.247384,	2074.299292,	2340.129404,	2887.294576,	3185.769968,
0x4f	322.007664,	468.376272,	648.322576,	1159.563560,	1396.322072,	1703.074704,	2296.020136,	2519.346720,	3200.100636,	3455.035180,
0x50	320.644656,	596.099288,	959.540408,	1218.782764,	1581.902180,	1962.503884,	2385.921440,	2708.652388,	3210.663204,	3479.108204,
0x51	579.432324,	803.501624,	1126.062900,	1427.862412,	1802.430756,	2082.734724,	2515.367492,	2857.989008,	3233.182996,	3504.253028,
0x52	551.063564,	710.286864,	942.023808,	1248.062556,	1525.983900,	1786.259580,	2161.901320,	2562.831384,	3071.864428,	3363.444284,
0x53	609.760980,	780.872944,	980.693916,	1270.962256,	1550.882860,	1844.290896,	2341.687584,	2627.830008,	3095.457540,	3424.026632,
0x54	368.966568,	542.830728,	697.958024,	962.227744,	1185.370252,	1618.496044,	2046.085300,	2329.287928,	2995.799936,	3309.116368,
0x55	404.026008,	558.097656,	811.595992,	1010.628240,	1461.065508,	2015.035896,	2351.943240,	2684.544988,	3120.826572,	3372.178544,
0x56	467.432068,	604.514196,	811.435040,	1070.217980,	1340.694896,	1626.960300,	1978.396652,	2451.284304,	2871.931440,	3178.879984,
0x57	483.238296,	618.742476,	778.447980,	1096.796044,	1376.987232,	1594.994712,	2291.458884,	2625.138056,	2974.698368,	3369.527960,
0x58	379.418368,	521.726544,	730.197388,	1034.923576,	1254.589772,	1880.551824,	2225.600464,	2536.139916,	3027.365712,	3270.175404,
0x59	484.397464,	617.122684,	839.871000,	1142.679472,	1409.236340,	2002.964060,	2395.110708,	2680.156916,	3127.463592,	3368.707032,
0x5a	379.080976,	610.318280,	777.690424,	1032.213916,	1526.638640,	1755.544788,	2096.849712,	2563.884348,	2924.848488,	3252.445700,
0x5b	519.918808,	686.437444,	889.952740,	1122.475300,	1354.885456,	1721.585208,	2279.554208,	2577.343264,	3148.668212,	3417.217104,
0x5c	327.841264,	480.902284,	620.889844,	893.219008,	1114.708076,	1389.494440,	2100.358500,	2641.157268,	3058.513220,	3378.730548,
0x5d	355.124124,	483.658108,	677.123732,	879.520776,	1104.300568,	1866.184292,	2279.529380,	2581.821656,	3144.676444,	3369.651800,
0x5e	394.104452,	587.768788,	783.373896,	1038.643028,	1318.632312,	1564.741340,	2021.770288,	2385.037484,	2817.015404,	3289.634680,
0x5f	419.169056,	568.314168,	734.757796,	982.644060,	1183.660644,	1521.688088,	2411.250204,	2684.267700,	3102.943040,	3407.177224,
0x60	217.051472,	442.467788,	1167.844336,	1599.059960,	1943.541700,	2342.287000,	2754.109460,	3023.786416,	3368.479160,	3560.444724,
0x61	548.527208,	936.006868,	1550.171484,	1861.420968,	2197.263736,	2455.181020,	2730.234412,	2965.074536,	3243.046748,	3471.871480,
0x62	429.353204,	649.857460,	1241.313404,	1625.726460,	1852.896036,	2164.172340,	2490.051440,	2778.301472,	3199.920908,	3454.966864,
0x63	530.200920,	767.252124,	1229.137072,	1718.003588,	2078.513660,	2358.501028,	2693.879268,	2963.116860,	3269.410220,	3513.980576,
0x64	180.250056,	286.314476,	782.009868,	1256.403520,	1607.783616,	1964.381668,	2428.505580,	2763.402500,	3181.404500,	3497.090480,
0x65	363.046928,	495.176544,	932.244352,	1723.729152,	2069.873516,	2266.036024,	2598.054636,	2863.095040,	3218.926356,	3530.567056,
0x66	275.456260,	401.919616,	743.453280,	1360.345704,	1599.375664,	1867.905052,	2173.127540,	2584.451656,	3032.595160,	3309.266888,
0x67	497.729776,	703.060684,	1036.341080,	1397.418484,	1838.024488,	2064.417060,	2475.461776,	2841.367628,	3176.211216,	3468.295196,
0x68	218.701004,	364.654668,	847.953432,	1455.778028,	1822.566468,	2271.184816,	2688.881072,	3009.378188,	3344.588860,	3544.795392,
0x69	409.265696,	722.538580,	1459.908136,	1792.828940,	2125.621812,	2424.411352,	2670.935304,	2920.454980,	3237.822100,	3467.081936,
0x6a	458.000836,	574.728436,	974.289496,	1430.359792,	1689.177736,	1976.396512,	2406.885924,	2657.533792,	3115.931004,	3439.716700,
0x6b	522.480632,	818.682484,	1332.380228,	1637.908688,	1904.114776,	2177.842412,	2487.380184,	2815.092956,	3239.457384,	3495.550436,
0x6c	215.477808,	328.983084,	581.950052,	997.003380,	1496.216816,	1965.586408,	2358.623932,	2637.005088,	3023.330960,	3434.554392,
0x6d	211.424108,	336.110144,	846.609096,	1295.560356,	1629.676036,	2078.982732,	2435.617148,	2785.006628,	3211.635028,	3492.684188,
0x6e	301.433204,	429.696220,	623.869464,	1219.478592,	1673.674336,	1872.317560,	2228.603416,	2498.196592,	3095.581436,	3326.314400,
0x6f	227.796312,	619.235776,	903.737200,	1186.101972,	1486.952480,	1881.668384,	2408.921256,	2705.309088,	3149.776640,	3445.450360,
0x70	246.964648,	410.663624,	963.556232,	1336.930556,	1805.419368,	2107.567904,	2494.256680,	2767.940768,	3170.005316,	3459.535500,
0x71	311.054428,	696.503640,	1198.813572,	1667.222600,	2009.052656,	2402.897624,	2794.768432,	3103.142944,	3356.889572,	3553.501980,
0x72	376.233832,	487.344044,	957.121240,	1293.941648,	1542.461412,	1934.271352,	2279.849188,	2603.799128,	3020.524136,	3276.772280,
0x73	420.577340,	782.720780,	1157.816496,	1409.439164,	1730.193544,	2036.551884,	2542.586672,	2940.837648,	3295.060216,	3536.318668,
0x74	227.450880,	354.217032,	645.854024,	1103.949568,	1457.017632,	1735.123748,	2145.927528,	2434.094168,	2993.342528,	3398.840732,
0x75	298.009992,	505.689304,	804.054936,	1230.271608,	1782.344852,	2092.537116,	2531.424824,	2811.813824,	3208.324332,	3479.810704,
0x76	361.837248,	481.881860,	754.778976,	1157.999988,	1373.736592,	1603.401296,	1872.798780,	2597.556168,	3080.013080,	3322.141852,
0x77	363.145448,	654.603360,	851.016964,	1201.750484,	1524.731340,	1765.555460,	2194.217812,	2663.022320,	3094.972416,	3372.444548,

0x78	244.837812,	378.566316,	819.184860,	1063.961972,	1389.825920,	1926.440476,	2268.995564,	2555.088408,	3159.662448,	3442.437796,
0x79	289.878276,	678.873488,	1094.283708,	1378.083076,	1669.550396,	2126.537152,	2546.927840,	2938.947780,	3349.758448,	3564.281964,
0x7a	291.135372,	603.970104,	806.647404,	1146.202408,	1492.922632,	1738.308380,	2185.345584,	2532.566400,	3038.664848,	3362.383560,
0x7b	477.451948,	711.119268,	1120.567768,	1400.126952,	1652.650504,	1928.385468,	2281.368392,	2627.063532,	3117.426876,	3435.299932,
0x7c	197.859932,	317.615912,	561.566488,	1004.156772,	1272.079512,	1786.734928,	2269.843440,	2576.009008,	3054.133780,	3435.857468,
0x7d	209.857268,	338.088960,	742.900840,	1010.365752,	1329.130948,	1933.476212,	2274.220164,	2550.770696,	2992.557772,	3438.759304,
0x7e	371.540116,	517.541568,	699.734220,	1056.769240,	1360.004676,	1581.063852,	1896.072308,	2138.187064,	2982.447132,	3386.475496,
0x7f	421.325684,	565.415952,	733.196828,	987.054432,	1189.015404,	1594.283500,	2041.152924,	2300.535092,	3176.614820,	3416.630796,

Stage 2:

Index	Vector									
0x0	-1.083480,	-11.578352,	2.306316,	-13.996316,	-69.456300,	37.766424,	-72.555080,	-159.215592,	-339.818812,	-152.118616,
0x1	-49.649148,	-46.254812,	-78.096196,	-42.879304,	87.142444,	6.807124,	-84.402392,	-169.554840,	114.687420,	84.983800,
0x2	-64.871376,	-74.238856,	-98.423948,	73.693940,	8.272440,	-0.445588,	-69.719076,	-97.613228,	-113.804960,	-72.737376,
0x3	-76.670260,	-104.000124,	162.718804,	146.267556,	35.949460,	-54.029504,	-165.439464,	-56.673624,	-30.629936,	-20.631364,
0x4	79.209324,	69.866840,	42.538208,	-8.700224,	-41.525928,	-87.881980,	-172.058432,	-257.161664,	-205.789544,	-58.729048,
0x5	46.523288,	32.708904,	16.372320,	87.959672,	17.893436,	-31.334540,	-158.771560,	-210.683128,	50.801572,	-1.070176,
0x6	-24.884988,	89.877856,	60.112892,	-46.721488,	-88.565924,	-123.916356,	-98.926960,	-105.548976,	-49.521476,	-27.839204,
0x7	12.216416,	-10.885964,	50.445364,	113.287312,	44.675968,	7.914896,	-81.805228,	-145.891432,	-269.429460,	-201.234052,
0x8	-18.702440,	-28.966588,	-107.616672,	-54.971216,	189.132464,	132.336076,	-7.024808,	-69.746976,	-94.375328,	-6.820516,
0x9	-56.894540,	-72.024184,	-52.360740,	193.625260,	129.735560,	62.003412,	-71.477448,	-142.162752,	64.098872,	11.162704,
0xa	-60.650768,	-78.194956,	-63.935496,	64.655352,	-25.341792,	84.625644,	73.669276,	12.005084,	149.501796,	80.509048,
0xb	-15.112784,	-31.196064,	3.849784,	159.434312,	93.375988,	58.684128,	-41.109156,	87.189492,	131.801980,	56.989420,
0xc	19.254800,	29.158032,	-18.931012,	-97.893236,	63.340576,	-10.453596,	-104.093976,	-176.150796,	-211.962572,	-17.986996,
0xd	43.597972,	19.186704,	57.392544,	116.543328,	78.867444,	26.069228,	-78.842272,	-162.151528,	142.039780,	80.101388,
0xe	-80.512424,	6.911116,	10.447620,	-20.670668,	122.874212,	64.969424,	-20.095012,	-41.500620,	40.870524,	27.243612,
0xf	15.133536,	-11.533644,	91.378952,	186.566144,	151.176812,	106.171932,	-10.561384,	-69.816204,	-158.913692,	-147.982316,
0x10	-22.857064,	-45.429420,	43.667084,	21.242924,	-51.172664,	132.138704,	25.815592,	40.259908,	-43.876200,	-80.833196,
0x11	-34.458148,	-37.767448,	-50.349484,	-21.896640,	91.649508,	17.432224,	65.073496,	11.976292,	-123.733292,	-107.504528,
0x12	-22.611732,	-44.342752,	65.095116,	32.444480,	-57.425188,	-5.691424,	138.519252,	62.980136,	71.781844,	39.682944,
0x13	-54.683340,	-52.559476,	144.402736,	36.354708,	-66.256948,	-121.447840,	-47.001024,	19.677400,	55.941392,	46.382952,
0x14	76.995752,	60.130164,	57.113272,	18.707744,	-53.600588,	62.799420,	-40.517780,	-114.388064,	-26.126324,	-73.326320,
0x15	25.816108,	31.076704,	124.868068,	63.559872,	51.002984,	-27.654288,	-97.466628,	-21.956264,	-41.809916,	-45.055148,
0x16	-45.228580,	102.897240,	83.947112,	74.901616,	49.974708,	8.687204,	48.971348,	6.900972,	-8.644580,	-8.383932,
0x17	61.045120,	182.798948,	159.671368,	59.777160,	29.855824,	-31.491100,	-95.978064,	-78.175532,	-39.050008,	-13.743948,
0x18	49.646432,	28.451656,	67.128192,	0.886752,	-49.083656,	188.228948,	106.267192,	13.855044,	-160.204028,	-171.642896,
0x19	-10.121400,	-10.494132,	-51.702072,	-102.624176,	212.475096,	189.695604,	82.285124,	56.342688,	-4.878576,	-11.235844,
0x1a	36.358408,	21.779392,	42.357136,	2.492964,	-42.223212,	230.770844,	160.695752,	154.000972,	136.475152,	53.130348,
0x1b	-29.676572,	-39.223244,	-35.661060,	94.983248,	59.318312,	2.700204,	207.235004,	137.851652,	158.257256,	108.921592,
0x1c	122.140104,	120.913744,	108.201320,	67.247396,	59.132704,	40.026908,	43.540560,	8.987920,	-51.137704,	-38.635328,
0x1d	72.657632,	68.934704,	16.956396,	25.957480,	190.086812,	142.948388,	53.217716,	2.736948,	-59.085276,	-22.405936,
0x1e	25.878136,	32.075092,	-18.051520,	129.615436,	41.005184,	141.842100,	80.837536,	-5.850756,	113.953980,	43.969864,

0x1f	56.950488,	37.665072,	132.514492,	135.379112,	75.671416,	138.633300,	118.390580,	101.879640,	101.928352,	42.930520,
0x20	22.106900,	21.059164,	-34.232504,	-93.793160,	-155.416992,	-194.590552,	20.038956,	-23.649856,	-114.881156,	11.324568,
0x21	-26.033500,	-35.961000,	-61.732092,	-162.427040,	-26.287444,	-98.269260,	-141.894620,	95.126388,	-1.505500,	-7.727060,
0x22	-31.289284,	-49.998772,	-67.781776,	-161.027220,	-175.279260,	-225.390404,	-61.840204,	0.995364,	16.801380,	48.424344,
0x23	-86.553744,	-108.489556,	-17.654176,	-72.132484,	-7.564812,	8.047932,	-86.936256,	27.160092,	12.540832,	22.373696,
0x24	17.779300,	15.339312,	-36.725424,	-8.881500,	-102.852904,	-144.867340,	-94.816980,	-196.287232,	76.219908,	45.250160,
0x25	-19.692492,	-30.220916,	-29.782548,	17.903896,	-49.210684,	-116.668528,	-256.575612,	-267.358620,	139.673908,	78.124496,
0x26	-19.207108,	4.375076,	21.642788,	-48.371256,	-77.052304,	-161.254660,	-158.684868,	250.771480,	151.906788,	53.175120,
0x27	-16.784044,	-37.622168,	5.782012,	-19.570792,	-88.589344,	-117.076008,	-236.655112,	-78.102088,	-107.415260,	-79.445552,
0x28	-33.759476,	-44.175904,	-74.702140,	-135.044380,	-188.927648,	7.016552,	82.302568,	25.474344,	76.734852,	27.800992,
0x29	-86.590864,	-120.632148,	-148.796704,	-150.115860,	-18.258660,	-84.054596,	2.635176,	37.246548,	34.580064,	56.689732,
0x2a	-70.539400,	-83.361280,	-55.232632,	-27.094692,	-100.538976,	-172.089768,	53.281532,	-13.033652,	9.663312,	46.571280,
0x2b	-51.370904,	-79.509096,	-141.551088,	1.691600,	57.508000,	-53.104592,	20.961464,	69.160500,	48.868372,	37.564824,
0x2c	-26.397896,	-22.278824,	-58.177544,	-99.990728,	-151.064636,	-22.789692,	-120.173196,	-151.092736,	-107.237000,	-84.906792,
0x2d	20.239624,	4.006208,	-28.664352,	-11.823076,	-26.946228,	-66.325064,	-59.575592,	-61.081748,	-37.028000,	-19.087332,
0x2e	-32.526320,	6.534672,	-84.214076,	-41.573904,	-42.894712,	-81.510608,	52.840208,	-43.398352,	72.024408,	68.427864,
0x2f	5.449508,	-14.963136,	-41.741028,	103.234668,	14.390108,	-64.423760,	100.949988,	29.811640,	-101.778792,	-32.993828,
0x30	32.295632,	18.407260,	-5.894076,	-74.483888,	-152.963476,	-204.044336,	205.952264,	219.522920,	81.562784,	61.202332,
0x31	18.857140,	38.485396,	-0.989396,	-55.047384,	-48.776448,	-130.013268,	107.102480,	146.292968,	-43.136920,	12.888356,
0x32	-0.859316,	-10.029480,	-5.970772,	-37.158360,	-112.881804,	25.395820,	308.705512,	231.753228,	148.199332,	77.118376,
0x33	-74.718268,	-7.140804,	45.716792,	5.980372,	66.215188,	4.203068,	147.785360,	193.993040,	80.089172,	48.570476,
0x34	66.596548,	54.383604,	27.843772,	2.162152,	-78.289776,	-136.087776,	83.808688,	-2.700500,	-13.633996,	6.292336,
0x35	44.048796,	18.914496,	31.356536,	6.002560,	-53.040524,	-33.569576,	-141.531972,	117.338852,	58.633508,	3.317584,
0x36	76.666832,	98.370332,	72.228980,	-23.754456,	-51.177288,	-139.149128,	-102.191076,	116.960600,	-15.327760,	-19.772720,
0x37	-24.797384,	74.266648,	59.426828,	-45.806184,	45.466244,	-18.249196,	-27.625136,	91.983212,	-22.390700,	-14.842720,
0x38	3.038084,	-17.986644,	-39.902620,	-148.611816,	-165.180916,	177.990356,	106.139356,	34.196816,	-54.350296,	-73.378024,
0x39	-44.622868,	-69.180132,	-94.306064,	-155.434984,	79.510788,	55.263412,	99.390112,	160.956712,	93.086912,	48.449336,
0x3a	8.528100,	-16.482720,	-20.536552,	-119.745020,	-107.365876,	343.542148,	310.162272,	223.345188,	122.204876,	31.527824,
0x3b	-74.730600,	-98.813932,	-83.739676,	156.816160,	172.775428,	133.228172,	117.447612,	109.584436,	74.203304,	39.213180,
0x3c	101.110828,	100.613404,	64.945400,	25.583900,	-51.148460,	26.378476,	-3.647860,	-4.167064,	141.475064,	72.420388,
0x3d	27.756364,	14.300488,	-39.955176,	-107.269680,	92.431928,	47.216472,	-69.001244,	85.735160,	-57.945236,	-63.784492,
0x3e	71.977736,	63.828088,	-0.871592,	-13.855768,	26.229036,	-35.582384,	133.902408,	73.158532,	103.646080,	78.834900,
0x3f	21.330804,	14.734456,	35.994576,	35.047512,	175.516392,	109.080608,	55.660600,	48.472556,	81.337388,	54.264844,

Stage 3:

Index					Vectors					
0x0	32.980820,	12.950916,	-77.521608,	3.457884,	21.115608,	-18.522344,	8.594232,	-32.455536,	79.444108,	57.096016,
0x1	20.015652,	21.871904,	-34.912312,	79.791640,	42.794480,	37.292348,	53.149448,	-12.294248,	-122.971792,	-61.566184,
0x2	14.547848,	-5.648276,	8.130888,	-50.548700,	-72.212136,	99.232328,	-8.558232,	-91.476160,	75.606244,	29.898352,
0x3	46.408076,	31.719192,	13.758460,	3.196728,	-60.667004,	96.666596,	30.653072,	-35.450592,	-54.536580,	-83.284228,
0x4	-6.354040,	-25.223476,	-47.876560,	-60.932144,	82.686748,	36.111092,	64.983628,	82.850232,	72.591960,	48.383168,
0x5	-29.084848,	25.116264,	-32.724520,	36.106892,	14.532280,	-62.664768,	98.917540,	40.993596,	64.825552,	65.144396,

0x6	-0.834372,	-21.045296,	5.294624,	-4.912748,	-69.596788,	125.862880,	43.944528,	64.578124,	72.109888,	20.414812,
0x7	25.715576,	39.154404,	-28.547164,	-23.826392,	-71.987108,	80.416488,	61.762952,	28.096916,	80.217544,	25.150576,
0x8	-21.216668,	0.919060,	-54.469596,	0.970748,	47.454988,	79.039928,	41.946800,	23.543096,	-61.798552,	-52.983464,
0x9	-19.268628,	-34.590936,	16.035816,	29.048544,	85.936380,	66.712128,	53.866020,	-9.536316,	46.423792,	40.015764,
0xa	-14.294308,	-28.466636,	24.792780,	-69.392596,	1.677280,	112.654604,	35.839992,	-4.488360,	-87.320780,	-64.399692,
0xb	32.723716,	17.351404,	63.382408,	13.164984,	7.863884,	32.270756,	32.808984,	-40.712860,	-159.488912,	-86.625816,
0xc	27.919388,	2.081884,	-40.560004,	-85.742272,	45.749008,	26.365512,	-37.336432,	91.871792,	-15.125316,	-27.295192,
0xd	-103.942848,	30.868692,	22.905368,	3.245652,	-10.864848,	25.740888,	32.632356,	14.937776,	62.672404,	36.229100,
0xe	-13.644212,	-33.497264,	7.802936,	-32.739608,	-68.895376,	-14.978252,	-13.743132,	-67.160192,	-154.790440,	-35.284996,
0xf	-61.220672,	-60.013976,	103.693816,	39.152964,	37.872412,	55.039632,	39.596404,	-12.290740,	-19.392204,	-4.557552,
0x10	32.505808,	15.583912,	1.784144,	53.673772,	73.051196,	41.469356,	-64.896556,	-7.048480,	18.025100,	-10.024028,
0x11	11.483176,	5.944724,	17.110496,	14.790684,	6.431540,	-20.200572,	-56.308788,	-134.147400,	-214.746576,	-59.407628,
0x12	43.566172,	18.989652,	55.144036,	-3.929840,	-37.379720,	61.812216,	-49.526292,	114.098232,	69.485776,	21.475288,
0x13	37.469540,	28.346816,	15.723080,	-28.085444,	-6.623116,	-19.773952,	-66.349236,	-31.002968,	-146.990804,	-120.377684,
0x14	1.120132,	-4.159908,	-14.437336,	-127.887976,	77.027304,	45.315748,	29.269936,	-15.638068,	8.620996,	14.473280,
0x15	44.768968,	30.467760,	-6.543172,	-95.704460,	26.850796,	-15.394560,	-25.588856,	-47.367504,	-2.881176,	39.420112,
0x16	-10.057328,	2.399412,	1.061956,	-65.695228,	20.687476,	-14.959904,	-117.018512,	92.386240,	85.463796,	24.175728,
0x17	49.644532,	65.386288,	9.684648,	18.181104,	-14.139504,	-5.888800,	-38.585284,	-27.183620,	58.865948,	16.431968,
0x18	-16.380640,	-34.278140,	0.022544,	56.067304,	46.892096,	30.245204,	-28.026068,	93.529428,	-56.789800,	-54.481380,
0x19	-4.060036,	-29.094112,	-52.545100,	79.031108,	69.252092,	14.267812,	9.463936,	-65.250772,	20.751184,	37.524948,
0x1a	4.583084,	-24.872664,	107.840788,	43.419736,	-1.250280,	-4.026280,	-63.961776,	21.494468,	-36.229604,	-32.554624,
0x1b	2.165696,	-29.151980,	54.527780,	94.168364,	66.493356,	36.958832,	1.594416,	-62.430724,	-87.198932,	-28.883704,
0x1c	-11.601520,	-21.077228,	-18.656652,	-63.311928,	81.936576,	29.605552,	-50.989424,	-103.844388,	-74.116364,	-25.022808,
0x1d	-20.789816,	23.424964,	27.131436,	-24.919604,	70.569748,	26.853608,	-31.525276,	-87.182472,	84.545612,	70.302900,
0x1e	-49.936228,	-3.446516,	40.200364,	-18.647372,	21.569628,	-58.944872,	-17.371572,	-22.311184,	-101.189676,	-49.805164,
0x1f	-57.177432,	62.058144,	36.515704,	55.561188,	6.520588,	6.297884,	-13.632288,	-52.826640,	-55.443100,	-54.692008,
0x20	6.249736,	5.520744,	-30.373784,	-42.147600,	20.147836,	-39.570052,	54.020328,	-16.561300,	-203.250932,	20.377448,
0x21	-35.205168,	5.085324,	-51.336656,	22.956972,	-28.242884,	-23.552672,	32.053652,	-17.893740,	-122.073052,	-160.749456,
0x22	24.945192,	13.164164,	11.748732,	-76.478132,	-4.066148,	-22.276580,	79.811928,	50.662004,	-81.866692,	-54.017108,
0x23	40.838864,	33.272060,	3.666024,	12.769580,	-45.868448,	-68.480588,	92.137976,	47.209512,	-35.710004,	-18.718076,
0x24	-13.862480,	-29.688380,	-16.062832,	-30.689744,	-21.411228,	-15.875376,	131.557968,	101.900708,	23.649376,	27.497916,
0x25	-20.078528,	-17.911912,	-24.694752,	-20.037984,	-59.345372,	-122.612400,	71.623648,	21.364328,	43.982568,	41.915580,
0x26	13.955236,	-13.670672,	-0.176880,	-47.697124,	-55.913972,	-56.124192,	62.862572,	-2.427248,	142.154476,	104.877084,
0x27	40.271076,	23.012344,	27.510152,	-41.497056,	-55.530680,	-58.212060,	8.071216,	-61.794624,	51.734348,	33.354252,
0x28	8.497868,	-2.606700,	-30.219332,	78.619552,	30.672148,	61.511100,	30.235552,	110.460236,	100.718520,	53.177352,
0x29	-20.167520,	-14.445504,	-9.336176,	127.582768,	-4.294328,	-27.401844,	61.786412,	21.701036,	22.223068,	15.230212,
0x2a	-1.647608,	54.368956,	37.237244,	-17.980500,	42.437252,	27.778416,	60.653436,	59.946944,	14.306756,	-5.660356,
0x2b	13.352932,	-14.668488,	56.844876,	39.456184,	25.357476,	11.232768,	96.450116,	63.098048,	-28.823600,	-11.658464,
0x2c	16.214176,	14.498636,	-9.273052,	-9.045960,	-27.283232,	-48.963732,	39.335716,	193.888504,	87.750712,	40.073856,
0x2d	-47.007404,	-53.423716,	9.553404,	0.589152,	-50.564088,	-1.666356,	-7.432612,	56.997036,	53.725740,	20.053884,
0x2e	-36.631948,	37.397136,	27.393352,	-30.540160,	-93.368500,	-20.053984,	-17.314104,	-0.911252,	14.602000,	0.825528,
0x2f	-1.395788,	-13.464588,	93.489772,	-1.615688,	-30.138632,	-48.064896,	15.280056,	66.923052,	69.265380,	43.586932,

0x30	10.569172,	24.809296,	-60.304192,	-48.704888,	-38.084248,	-30.716660,	-41.121760,	-46.544012,	-33.652504,	-45.618716,
0x31	14.329408,	2.787968,	1.889624,	17.819504,	-36.893736,	-131.985160,	-0.567740,	-55.884160,	-72.492488,	-31.240228,
0x32	27.387340,	7.644616,	27.150636,	-5.574300,	-36.398996,	-76.648084,	-129.163304,	86.750668,	45.036408,	6.367308,
0x33	51.258220,	25.407856,	15.030632,	72.681664,	-0.275320,	-33.546500,	-44.555092,	67.260580,	-40.684228,	-43.996440,
0x34	-25.149456,	-55.373592,	-20.508880,	-59.104064,	39.694332,	-29.855136,	23.338884,	-29.142064,	-15.073252,	16.158980,
0x35	-24.028792,	-33.787436,	7.601280,	-28.737668,	-9.413580,	-62.113404,	-56.180560,	-121.529172,	83.351164,	60.314356,
0x36	27.417444,	7.533384,	0.415240,	-4.640872,	-21.214400,	-41.096728,	-104.321976,	-151.643732,	171.809252,	110.383744,
0x37	34.088864,	14.909740,	58.394348,	39.166456,	2.694692,	-32.094636,	-33.619456,	-101.999952,	32.469464,	32.881976,
0x38	-13.082648,	-30.073608,	-85.362912,	6.847348,	-46.908688,	36.769352,	43.168012,	8.622884,	-8.777696,	-8.067804,
0x39	-41.948776,	1.387452,	-52.071212,	73.038768,	4.876640,	-17.894024,	-51.727328,	-70.581328,	17.584816,	-10.379668,
0x3a	-12.247768,	-11.237092,	-44.331148,	30.028040,	-39.230748,	-2.865544,	-119.814076,	72.574860,	-49.888108,	-64.795236,
0x3b	8.651028,	-16.327560,	0.858176,	49.272304,	-33.946904,	45.591748,	-8.960968,	-67.881588,	-94.941176,	-110.256420,
0x3c	-12.528140,	-28.219792,	-66.117904,	19.523384,	2.767992,	-68.743624,	-86.900348,	91.572472,	64.267052,	42.045840,
0x3d	-18.846356,	-40.824100,	-12.869724,	6.287764,	-18.501512,	23.048256,	-60.021784,	-20.368256,	222.374712,	136.273740,
0x3e	-65.678876,	21.048360,	10.027512,	27.274816,	12.681548,	-38.586556,	-102.199520,	98.262288,	55.721236,	13.438876,
0x3f	30.881868,	20.481092,	5.767264,	58.983840,	24.991944,	11.221364,	42.022696,	-42.025784,	160.539288,	108.165892,

Stage 4:

Index					Vectors					
0x0	13.864392,	-19.737336,	20.320452,	-1.373732,	-24.025380,	55.403468,	-33.427036,	-7.164116,	-76.739792,	-66.680096,
0x1	-22.288548,	-34.188684,	20.704048,	-14.170992,	28.366312,	-25.985648,	-43.561588,	15.943264,	46.563540,	34.635664,
0x2	-9.949592,	14.031280,	2.022688,	61.570748,	11.537028,	-20.475192,	-56.247088,	12.789900,	-24.688080,	-20.578288,
0x3	-64.483988,	-2.641932,	9.354576,	-1.695224,	-11.159228,	1.977040,	3.499672,	18.992872,	-28.880704,	-17.282880,
0x4	1.859316,	-18.637868,	39.190940,	19.354980,	44.978516,	5.672544,	-3.360900,	30.381176,	-52.519360,	-41.922064,
0x5	22.329840,	-8.827276,	35.918016,	-26.647964,	26.134024,	-41.064820,	4.758780,	-49.390996,	7.984100,	30.442724,
0x6	17.873364,	-3.982856,	44.497744,	39.808760,	-3.764812,	-8.364812,	-35.478012,	-11.361596,	80.956140,	54.743676,
0x7	7.958560,	-32.041208,	17.138880,	37.115176,	17.563628,	-11.659032,	33.656416,	-18.415836,	59.808248,	36.363976,
0x8	-12.145288,	-24.479824,	56.629472,	-11.682896,	-31.763652,	3.683420,	-20.414980,	-31.909604,	9.772540,	-1.279444,
0x9	-14.054776,	-35.296828,	-19.199396,	17.575908,	-28.405600,	-4.153392,	7.303092,	-29.658076,	-44.023168,	-79.546640,
0xa	11.007008,	19.022892,	28.676476,	8.685068,	41.181528,	9.850048,	-35.909068,	-57.330104,	7.240120,	18.847528,
0xb	-11.754804,	-0.200860,	-0.397848,	16.824616,	41.821960,	15.431596,	-17.873084,	-46.166628,	-113.237564,	3.930596,
0xc	4.417632,	-13.858636,	25.993272,	3.621776,	-13.282620,	-54.866568,	-25.740860,	-8.207356,	-102.337520,	-90.469400,
0xd	30.472892,	2.402004,	-8.414600,	-14.939776,	20.846288,	-42.379776,	44.159640,	-1.039776,	-86.199308,	-29.886620,
0xe	20.803664,	-5.884716,	-10.076072,	34.678792,	-1.836392,	-5.648408,	-57.621056,	-64.743808,	-35.249396,	-36.447272,
0xf	5.240548,	-7.730720,	-42.546312,	25.064936,	15.738880,	6.281712,	21.895068,	5.559424,	-104.670464,	-52.286052,
0x10	24.895620,	3.387108,	14.067512,	-35.964624,	34.713716,	56.939140,	17.696072,	-3.026896,	-64.985692,	-46.711140,
0x11	-20.307876,	32.490736,	-8.269384,	-37.426408,	15.369516,	5.518868,	-40.295176,	49.790088,	-30.418412,	-28.439284,
0x12	3.301884,	23.180704,	-24.454116,	-5.660896,	-46.692200,	72.857908,	3.808256,	-13.518736,	-27.566972,	-39.098900,
0x13	-36.148208,	33.906036,	-10.643360,	27.813232,	7.439476,	23.088892,	41.102328,	-5.758800,	-57.094056,	-38.619512,
0x14	41.485516,	33.976276,	16.557244,	11.715616,	1.168420,	11.215132,	2.419500,	-2.189148,	-20.660864,	-20.782992,
0x15	-34.868980,	44.335072,	18.371036,	-9.576508,	6.842808,	-28.376484,	-20.193584,	-36.100756,	-56.015728,	-32.668848,
0x16	22.181968,	-4.747624,	-6.587840,	19.194412,	13.324592,	50.283068,	34.604528,	-63.345460,	-24.597824,	18.117740,

0x17	-23.391532,	11.590768,	37.601468,	12.769832,	-9.503604,	-11.279292,	54.145876,	-50.561068,	11.992376,	13.849812,
0x18	-7.408196,	-30.925108,	18.743584,	-54.589100,	40.004624,	22.577360,	8.870712,	-11.437640,	1.800908,	16.894572,
0x19	2.106444,	-17.324588,	-22.842364,	-28.871344,	-21.276340,	-13.394672,	23.784676,	70.579396,	-37.411672,	-52.086276,
0x1a	15.241316,	17.603468,	-41.962316,	-4.684472,	53.083624,	16.845052,	-13.084284,	-12.831660,	-7.038208,	-2.189224,
0x1b	-5.009304,	-10.205264,	-23.680948,	-23.020348,	55.499976,	26.375932,	64.954384,	7.736592,	-5.772604,	-18.452616,
0x1c	17.559028,	9.416808,	39.605736,	23.848632,	1.871332,	-54.038420,	27.998276,	35.575168,	-29.006648,	-1.178764,
0x1d	-6.962476,	4.105532,	19.244696,	-23.883640,	-39.505700,	19.590216,	53.142492,	8.583808,	-84.255052,	-13.993440,
0x1e	29.861688,	7.994612,	-19.893828,	45.102624,	8.184900,	-39.195660,	30.159276,	-34.246888,	17.732108,	23.665304,
0x1f	14.491724,	-5.943524,	-5.773136,	28.083748,	-36.821584,	-25.838832,	80.000780,	12.158860,	-10.060840,	-7.431432,
0x20	-5.468828,	-26.673576,	-13.033748,	15.823448,	2.773560,	29.638816,	-77.755704,	29.586208,	3.212960,	-27.223712,
0x21	4.192508,	-0.595220,	-9.597436,	-29.569652,	-8.967564,	-15.488152,	-87.442240,	33.505968,	81.622340,	43.421084,
0x22	6.918280,	5.475584,	-34.010376,	55.845532,	3.372104,	41.049620,	-16.152440,	1.566632,	52.853932,	16.039552,
0x23	-38.947288,	18.104492,	-36.755712,	-14.831836,	14.532324,	-0.120888,	0.709208,	-29.737616,	63.593660,	43.486380,
0x24	16.827216,	-8.372392,	-4.047592,	14.480476,	48.383496,	8.909460,	-23.019528,	74.803304,	51.072416,	26.536640,
0x25	16.908200,	3.113504,	-0.409796,	-15.891340,	30.269736,	-61.639948,	-30.195632,	59.169420,	-8.384824,	-4.557260,
0x26	-19.531304,	-19.663028,	10.757152,	54.459716,	-5.494000,	-20.867892,	18.984844,	61.646372,	26.824196,	19.117792,
0x27	-29.128836,	-3.459292,	-9.332620,	17.339504,	25.026592,	-59.119656,	54.056544,	-12.055064,	-21.268108,	2.680656,
0x28	-15.645544,	-21.784092,	-4.418296,	-12.523804,	-16.660484,	74.213892,	34.056316,	-0.726300,	51.928168,	22.675180,
0x29	15.689516,	-11.155860,	-21.276536,	-14.047236,	-54.493236,	13.666468,	-30.506112,	-37.156156,	43.355576,	7.350436,
0x2a	-29.735320,	-13.753276,	3.548232,	31.269112,	39.668368,	44.830620,	-6.977028,	-35.663752,	30.555460,	4.388992,
0x2b	-31.082216,	11.584868,	8.487652,	-33.036500,	-13.157316,	50.502976,	-23.188224,	-51.524568,	0.382960,	-5.737796,
0x2c	17.331036,	-10.261824,	31.780508,	-23.296888,	8.928572,	52.597664,	-44.915272,	-30.676572,	65.429492,	36.915764,
0x2d	7.656704,	-12.769280,	-34.288816,	-46.295780,	7.367804,	0.185976,	-26.244420,	-33.013772,	-69.086572,	-27.984080,
0x2e	-18.013668,	-10.073172,	3.763892,	24.780428,	-21.653296,	4.704220,	-13.278056,	-57.884760,	122.655536,	78.080140,
0x2f	-7.139380,	-3.507464,	-16.812212,	-10.187820,	11.070308,	-45.039224,	-26.121504,	-78.043388,	22.171320,	17.720240,
0x30	-0.038328,	-12.205152,	2.765344,	43.550748,	4.322620,	64.041088,	37.213004,	30.822664,	-49.288732,	-32.218564,
0x31	-24.910772,	6.883424,	-12.832696,	1.845308,	-52.037336,	32.769556,	-20.558604,	73.189036,	38.056292,	9.735312,
0x32	-1.617564,	23.527380,	-13.455208,	32.733468,	38.018240,	5.814052,	46.106580,	49.198760,	6.412352,	3.359260,
0x33	-22.066160,	28.427852,	24.169128,	9.870268,	22.294164,	-14.262904,	-2.228472,	56.488580,	74.734164,	52.285380,
0x34	20.856168,	11.434760,	-17.501460,	19.356592,	-23.927048,	-10.080176,	-29.157472,	67.028020,	-72.829380,	-38.168352,
0x35	4.532676,	6.949228,	30.519480,	-1.335668,	-46.754808,	-16.408320,	-71.551928,	39.060732,	-5.316548,	-16.951812,
0x36	-6.563488,	-4.683368,	-44.646952,	11.974452,	-5.360264,	-32.144792,	27.619440,	35.933544,	80.803460,	59.884136,
0x37	-12.079848,	28.782912,	-5.610228,	32.414792,	-38.174764,	-53.727672,	-3.846380,	6.256724,	29.202688,	12.668764,
0x38	9.128584,	-15.964872,	40.529996,	-25.097428,	-9.404020,	33.634748,	25.128176,	67.999700,	15.799964,	-7.397816,
0x39	-0.584244,	-19.878520,	15.734836,	-26.676744,	-33.555660,	-67.913680,	2.189504,	28.161576,	19.481028,	6.669180,
0x3a	27.830096,	3.212196,	-18.081108,	-53.418624,	16.337124,	17.790228,	-0.935908,	18.558948,	66.264504,	36.820628,
0x3b	-8.894760,	17.362400,	13.714624,	-55.179844,	3.355216,	-25.666356,	52.092692,	9.626172,	37.176688,	36.454116,
0x3c	24.708572,	2.926604,	19.614548,	11.151496,	-50.921244,	31.058220,	38.814708,	9.470088,	51.206496,	7.702788,
0x3d	26.751716,	10.221092,	20.453016,	-38.210332,	-40.726020,	-25.178320,	-9.964692,	-19.559148,	-45.512164,	-28.789868,
0x3e	19.678856,	24.642696,	-2.825972,	-14.446076,	-19.540088,	-8.693844,	1.168576,	-59.036728,	102.917656,	79.647188,
0x3f	-7.553260,	16.710720,	-31.982072,	-28.903800,	-40.696264,	-28.379732,	36.020044,	-21.981404,	-5.020000,	-4.847084

Appendix D

This appendix lists the codebook used by the Fourier magnitude vector quantizer.

Index	Vectors									
0x0	0.822998,	1.496297,	0.584847,	1.313507,	0.846008,	0.614041,	0.615298,	0.881819,	1.227331,	0.902116,
0x1	1.248150,	1.020382,	0.517184,	1.489079,	0.650498,	0.716904,	0.650349,	1.260358,	0.871024,	0.907932,
0x2	1.167392,	1.468091,	0.743322,	0.712255,	1.564293,	0.721041,	0.528277,	0.657527,	0.763228,	0.846655,
0x3	1.043080,	1.570270,	0.444422,	0.933200,	1.211773,	0.669870,	0.932675,	0.806473,	0.913787,	0.868244,
0x4	0.759126,	1.037894,	1.126967,	1.133186,	0.789090,	0.807399,	1.054906,	0.949898,	1.007863,	1.119437,
0x5	0.977225,	1.130888,	1.044601,	1.432985,	0.819888,	0.517787,	0.932401,	0.964477,	0.886867,	0.895436,
0x6	0.666187,	1.364039,	1.088940,	0.902131,	1.110838,	0.980791,	0.926721,	0.867719,	0.925053,	0.913544,
0x7	0.983185,	1.460579,	0.732273,	0.923475,	1.088579,	0.952291,	0.789664,	1.025082,	0.904670,	0.854338,
0x8	0.697275,	1.651033,	0.506420,	0.809603,	1.113992,	1.083678,	0.629739,	0.986236,	0.871250,	0.997090,
0x9	0.749416,	1.354976,	0.731452,	1.281438,	1.080946,	1.037235,	0.748186,	0.821493,	0.879839,	0.965350,
0xa	0.976897,	2.009469,	0.741592,	0.606862,	0.644927,	0.687413,	0.760360,	0.833318,	0.869851,	0.810534,
0xb	0.576891,	1.527863,	0.657703,	1.006739,	1.226803,	0.786733,	0.991212,	0.821165,	1.014984,	0.878058,
0xc	0.517211,	1.220859,	0.886556,	0.980153,	0.884365,	1.041315,	0.949276,	0.881296,	1.373464,	0.890748,
0xd	0.651809,	1.106789,	0.885180,	1.329281,	1.124388,	0.626778,	1.261132,	0.822453,	0.966925,	0.806183,
0xe	0.614335,	1.430380,	0.899696,	0.847168,	0.624567,	1.307056,	1.360741,	0.735438,	0.854307,	0.674763,
0xf	0.830604,	1.373081,	0.785967,	0.735951,	0.968745,	1.100706,	0.927619,	0.728449,	1.292031,	0.884959,
0x10	1.084085,	1.385767,	0.448480,	1.297458,	0.770611,	0.719263,	1.080766,	0.909836,	0.838299,	0.909742,
0x11	1.244608,	1.158451,	0.615757,	1.169599,	1.032735,	0.921809,	0.916509,	1.060343,	0.836990,	0.767101,
0x12	1.313869,	1.341417,	0.481710,	0.685601,	1.086817,	1.003461,	0.762127,	0.832518,	1.081618,	0.917168,
0x13	1.427137,	0.932544,	0.595881,	0.652610,	1.073852,	0.847480,	1.264058,	0.899502,	0.899264,	0.958964,
0x14	1.027745,	1.214124,	0.896291,	0.972030,	0.841943,	0.738307,	1.048254,	1.077882,	0.952564,	1.056319,
0x15	0.952252,	1.098463,	0.690316,	1.175721,	0.911712,	0.962678,	0.859269,	1.138901,	1.074558,	0.924091,
0x16	0.902314,	1.163750,	1.143587,	0.814860,	0.923235,	0.973077,	1.098871,	0.966014,	0.968856,	0.893133,
0x17	1.133879,	1.110463,	0.687589,	0.793794,	1.026535,	1.076900,	1.185281,	0.821843,	0.845464,	1.079782,
0x18	0.623910,	1.269921,	0.722072,	0.990256,	1.183314,	1.161476,	0.677381,	1.199311,	0.902081,	0.883634,
0x19	0.585861,	1.189163,	0.882952,	1.134136,	0.900218,	1.263223,	0.995922,	1.011126,	0.906268,	0.863339,
0x1a	0.687616,	1.562003,	0.815795,	0.701934,	0.789752,	1.204232,	0.821560,	1.168157,	0.888904,	0.860778,
0x1b	0.950686,	1.329037,	0.692678,	0.754646,	1.095995,	1.446339,	0.941535,	0.710369,	0.789949,	0.826271,
0x1c	0.520026,	1.293024,	0.938993,	0.990412,	0.897114,	0.914170,	1.358696,	0.952011,	0.936183,	0.837973,
0x1d	0.669622,	1.291638,	0.749196,	1.167486,	1.042641,	0.721329,	1.005261,	1.252692,	0.883186,	0.854416,
0x1e	0.497926,	1.222508,	1.403952,	0.799058,	0.798533,	0.793545,	1.264461,	0.958949,	0.843825,	0.909886,
0x1f	0.891705,	1.342163,	0.761635,	0.758182,	0.987397,	1.008477,	1.377249,	0.845635,	0.845744,	0.822265,
0x20	1.174305,	0.454482,	0.432532,	0.691549,	1.066271,	0.890505,	0.845957,	1.125649,	1.338280,	1.204068,
0x21	1.607432,	0.688224,	0.500617,	1.288348,	0.678978,	0.674969,	1.047764,	1.012974,	0.954688,	0.803209,
0x22	1.739179,	0.919542,	0.639494,	0.775935,	1.021945,	0.991820,	0.887633,	0.876449,	0.796800,	0.771028,
0x23	1.620174,	1.211703,	0.610139,	1.372774,	1.057336,	0.618126,	0.642995,	0.667444,	0.666172,	0.676810,
0x24	1.219952,	0.844191,	1.016601,	0.867402,	1.006595,	0.940309,	0.893858,	0.963053,	0.730414,	1.286290,

0x25	1.251753,	1.004762,	0.777132,	1.203828,	0.656382,	0.657205,	1.233246,	0.874799,	1.106710,	0.855250,
0x26	1.348819,	0.791939,	1.030755,	0.765314,	1.014939,	1.044297,	0.747682,	1.309667,	0.853913,	0.723126,
0x27	1.262759,	1.109681,	0.672002,	1.109957,	0.827608,	1.272683,	0.670943,	1.068174,	0.752511,	0.878203,
0x28	1.108504,	1.250712,	0.620039,	0.752463,	1.439216,	0.890828,	1.018300,	0.736766,	0.910504,	0.849118,
0x29	1.222589,	1.083869,	0.627894,	1.150342,	1.377353,	0.785771,	0.679900,	0.897937,	0.854521,	0.920765,
0x2a	1.326349,	1.690290,	1.188837,	0.595281,	0.602145,	0.626852,	0.676343,	0.829212,	0.785923,	0.799661,
0x2b	1.499772,	1.400400,	0.655881,	0.892565,	1.089036,	0.702486,	0.835749,	0.800092,	0.731263,	0.835818,
0x2c	1.126664,	1.353291,	0.898265,	0.897870,	0.934461,	1.003119,	0.982470,	0.815886,	0.952342,	0.831774,
0x2d	1.316478,	1.060041,	0.667653,	1.065408,	0.910377,	1.002121,	1.339044,	0.710796,	0.720291,	0.813624,
0x2e	0.976194,	1.377380,	1.071262,	0.789567,	1.085928,	1.253806,	1.017727,	0.719049,	0.561583,	0.692329,
0x2f	1.202869,	1.059008,	0.962883,	0.999919,	0.524897	1.074531,	1.091141,	0.910749,	1.044517,	0.859795,
0x30	1.413976,	0.836644,	0.666408,	1.039645,	1.088087,	0.820217,	0.816371,	1.049252,	1.083270,	0.881172,
0x31	1.388093,	0.807110,	0.781561,	0.964242,	0.973557,	1.138037,	1.026733,	0.970914,	0.866461,	0.841057,
0x32	1.456077,	0.982162,	0.557993,	0.738683,	1.375522,	0.772422,	0.644426,	0.952329,	0.954631,	1.004692,
0x33	1.606208,	1.220452,	0.738208,	0.744965,	0.659231,	0.813191,	0.932141,	0.922849,	0.956161,	0.892080,
0x34	1.126839,	1.038057,	0.935449,	1.012686,	1.080640,	0.884338,	1.126468,	1.076987,	0.817224,	0.736383,
0x35	1.215351,	0.845602,	0.829299,	1.204972,	0.891515,	0.934431,	0.887728,	0.938925,	0.955741,	1.114145,
0x36	1.206387,	0.977863,	1.051199,	0.847855,	0.939759,	1.254109,	0.970361,	0.901040,	0.844070,	0.819544,
0x37	1.207602,	0.976323,	0.841307,	0.870300,	0.877422,	0.988163,	0.931714,	1.031677,	1.205409,	0.903902,
0x38	1.095130,	1.105129,	0.722843,	0.943180,	1.211443,	1.178861,	0.634252,	0.870933,	1.011727,	0.936443,
0x39	1.032014,	0.998735,	0.686262,	1.100404,	1.022352,	1.222401,	1.146689,	0.812882,	0.927129,	0.801404,
0x3a	1.131931,	1.489739,	0.677826,	0.610733,	0.759219,	0.808122,	1.042500,	1.077132,	1.035717,	0.906736,
0x3b	1.342504,	1.076171,	0.583400,	0.764890,	0.818097,	1.293405,	1.058200,	0.852259,	0.958646,	0.872888,
0x3c	1.308520,	1.219689,	0.935572,	0.714861,	1.129251,	0.856944,	0.920880,	0.974834,	0.821443,	0.847930,
0x3d	1.372592,	0.887155,	0.710732,	1.029836,	1.278897,	0.650119,	1.136552,	0.881858,	0.850047,	0.819899,
0x3e	1.019956,	1.381944,	1.089064,	0.576825,	0.594933,	0.752016,	1.265619,	1.129530,	0.867864,	0.768122,
0x3f	1.154864,	1.069591,	0.913967,	0.796105,	0.790255,	0.854206,	1.523905,	0.856969,	0.887946,	0.756665,
0x40	0.874806,	1.645517,	0.834275,	1.578196,	0.903996,	0.547038,	0.565520,	0.704114,	0.697290,	0.713786,
0x41	1.153418,	1.095160,	0.870564,	1.655735,	1.071838,	0.806210,	0.607539,	0.566859,	0.683404,	0.826413,
0x42	0.830864,	1.448871,	1.014508,	0.939034,	1.574030,	0.796490,	0.654852,	0.558685,	0.641596,	0.797132,
0x43	1.125819,	1.396448,	0.746324,	1.295697,	1.137447,	1.301677,	0.574765,	0.486441,	0.489204,	0.608138,
0x44	0.832614,	0.802739,	1.148272,	1.316531,	1.305743,	0.696667,	0.940671,	0.848198,	0.872696,	0.851311,
0x45	1.260278,	0.958467,	0.667246,	1.406435,	0.859092,	0.534003,	0.485425,	0.723814,	1.418320,	0.944772,
0x46	0.887527,	1.330678,	1.062704,	0.552689,	1.098956,	0.917567,	0.874732,	1.075172,	0.845710,	1.019528,
0x47	0.906302,	1.326878,	0.632127,	0.781817,	1.111568,	0.753606,	0.896250,	0.945621,	0.969063,	1.304544,
0x48	0.163798,	1.616260,	0.520967,	1.244674,	0.597986,	1.174228,	0.802735,	1.079115,	0.892800,	0.877604,
0x49	0.717419,	1.075445,	0.997204,	1.206762,	1.312107,	1.316050,	0.696587,	0.745413,	0.739740,	0.731447,
0x4a	0.339447,	1.827655,	0.684798,	0.716559,	0.676084,	1.081139,	0.894730,	0.954454,	0.948165,	0.910477,
0x4b	0.864667,	1.451421,	0.725543,	1.176080,	1.209637,	1.083631,	1.163663,	0.662386,	0.501862,	0.463726,
0x4c	0.563978,	1.135516,	0.835026,	1.736364,	1.000137,	0.805360,	0.828496,	0.833563,	0.776277,	0.805186,
0x4d	1.009051,	1.360317,	0.668259,	1.417717,	1.518623,	0.638014,	0.446922,	0.496800,	0.693633,	0.726920,
0x4e	0.186440,	1.625134,	0.835253,	1.175778,	0.860628,	0.885086,	1.011866,	0.789303,	1.010232,	0.802450,

0x4f	0.652311,	1.599127,	0.599516,	1.058215,	0.801753,	0.992815,	1.146092,	0.829615,	0.939228,	0.837653,
0x50	0.759780,	1.399480,	0.987329,	1.024824,	0.974458,	0.551246,	1.150852,	0.844956,	1.016771,	0.925568,
0x51	1.069814,	1.160678,	0.803250,	1.193140,	1.112463,	0.744748,	1.070724,	0.739996,	0.924675,	0.942765,
0x52	1.168517,	1.633441,	0.774778,	0.736824,	1.089258,	1.057042,	0.783527,	0.691403,	0.779568,	0.728487,
0x53	1.301817,	1.572909,	0.526128,	0.558027,	0.660410,	1.078262,	1.089774,	0.718543,	0.753433,	0.965337,
0x54	0.891795,	1.262297,	0.911545,	1.214103,	0.776010,	0.944986,	1.138773,	0.772798,	0.957464,	0.885334,
0x55	1.219015,	1.048683,	0.943178,	1.150567,	0.837824,	0.566225,	0.660764,	1.219023,	1.019092,	0.988723,
0x56	0.925533,	1.350162,	1.164350,	0.542695,	1.227640,	0.600203,	1.205779,	0.701896,	0.932956,	0.815466,
0x57	1.019130,	1.292473,	0.796847,	0.819714,	1.254208,	0.522062,	1.293349,	0.823616,	0.832862,	0.890015,
0x58	0.686388,	1.534513,	1.023273,	1.065238,	0.769952,	0.914897,	0.958226,	0.965983,	0.847693,	0.876086,
0x59	0.560425,	1.321635,	0.837563,	1.105726,	0.800833,	0.952432,	0.673960,	1.327502,	0.806885,	1.135634,
0x5a	0.722275,	1.482196,	1.269134,	0.695424,	0.734410,	1.240968,	0.873750,	0.825209,	0.794290,	0.840055,
0x5b	1.022015,	1.408327,	0.702067,	0.966478,	0.742264,	1.204704,	0.633885,	0.888498,	0.713166,	1.231153,
0x5c	0.512817,	1.187577,	0.939995,	1.273059,	1.261853,	0.875396,	0.884657,	0.928591,	0.900721,	0.880112,
0x5d	0.781537,	1.223652,	0.854824,	0.972599,	1.347144,	0.807227,	0.911694,	0.956919,	0.990262,	0.893345,
0x5e	0.358832,	1.601542,	1.125781,	0.730841,	0.795475,	0.856229,	1.070817,	0.869601,	1.043598,	0.896567,
0x5f	0.821456,	1.676333,	0.931807,	0.740210,	0.946762,	0.917425,	0.936539,	0.884692,	0.872158,	0.827619,
0x60	1.133434,	1.069057,	0.628708,	1.348112,	1.024242,	0.726333,	0.485013,	0.517711,	0.812878,	1.483409,
0x61	1.439495,	0.401963,	0.704352,	1.185582,	1.196889,	0.894400,	0.762308,	0.871978,	1.004571,	0.965074,
0x62	1.355857,	1.217795,	0.623188,	0.626000,	1.080819,	1.443039,	0.690217,	0.676470,	0.752167,	0.890824,
0x63	1.384203,	0.893545,	0.571820,	1.422045,	1.054457,	0.896980,	0.929532,	0.779836,	0.816629,	0.774798,
0x64	1.272930,	0.807536,	0.943865,	1.161822,	1.109685,	0.883353,	0.756987,	1.183318,	0.792499,	0.818768,
0x65	1.368426,	0.939970,	0.927314,	1.372501,	0.966845,	0.661820,	0.817570,	0.813094,	0.874638,	0.905751,
0x66	1.234512,	0.812394,	1.092298,	0.956999,	0.926831,	1.159350,	0.560790,	0.914404,	1.104409,	0.951357,
0x67	1.462063,	1.107453,	0.736919,	1.182041,	0.753060,	0.817381,	0.845724,	0.960321,	0.930034,	0.870101,
0x68	1.154263,	1.149050,	0.969239,	1.003195,	0.709246,	1.334899,	0.539388,	0.648414,	1.170228,	0.841587,
0x69	1.363427,	0.856319,	0.848329,	1.330223,	1.580245,	0.593904,	0.534564,	0.577523,	0.647285,	0.799871,
0x6a	1.430748,	1.277622,	0.535910,	1.316419,	0.704018,	1.139968,	0.844730,	0.708186,	0.689915,	0.673166,
0x6b	1.456831,	0.852769,	0.836842,	1.661979,	0.555156,	0.659789,	0.754965,	0.765223,	0.801905,	0.871901,
0x6c	1.219340,	0.941999,	1.021315,	1.258016,	0.869648,	1.086416,	1.006363,	0.806097,	0.807889,	0.739562,
0x6d	1.269435,	1.217910,	1.091920,	1.130622,	1.241678,	0.548636,	0.659441,	0.769661,	0.738142,	0.844145,
0x6e	1.109102,	1.109600,	0.921873,	1.080103,	0.644065,	1.551581,	0.917328,	0.782032,	0.649409,	0.682361,
0x6f	1.148823,	1.115256,	0.784950,	1.519787,	0.632186,	0.880947,	0.965093,	0.738516,	0.923041,	0.843738,
0x70	1.099162,	1.072237,	1.059239,	1.012868,	1.091543,	0.820111,	0.632757,	0.796845,	1.140959,	1.032531,
0x71	1.382078,	1.014650,	0.855020,	1.088445,	1.147617,	1.054602,	0.759230,	0.758359,	0.828497,	0.830315,
0x72	1.331108,	0.911897,	0.551680,	0.739304,	1.080511,	1.173246,	0.671436,	0.925538,	1.082333,	1.110124,
0x73	1.380841,	1.130295,	0.743747,	0.910074,	0.957719,	0.986217,	0.885270,	0.809240,	0.914088,	1.051811,
0x74	1.310270,	0.998818,	0.989857,	1.025202,	0.906801,	0.876459,	0.895856,	0.981348,	0.982188,	0.901957,
0x75	1.442392,	0.947518,	1.052543,	0.888925,	1.092888,	0.674498,	0.920650,	0.987177,	0.890263,	0.816537,
0x76	1.173251,	0.879037,	1.128982,	0.905439,	1.302826,	1.072854,	0.709855,	0.689286,	1.000834,	0.819403,
0x77	1.162547,	0.939193,	1.038151,	0.930689,	0.914292,	0.877584,	1.144403,	0.667857,	1.257437,	0.810249,
0x78	0.957634,	1.145630,	1.008914,	1.045813,	1.030746,	1.071616,	0.801657,	0.909400,	0.880216,	1.014015,

0x79	1.250553,	1.148173,	0.606277,	0.991795,	0.997859,	0.688890,	0.942568,	0.922857,	1.233180,	0.891771,
0x7a	1.223899,	1.343365,	0.813470,	0.600631,	0.514261,	1.292102,	0.994158,	1.012705,	0.774822,	0.889761,
0x7b	1.434258,	1.190874,	0.478282,	0.454840,	0.731792,	1.170314,	0.776934,	1.201793,	0.865726,	0.993823,
0x7c	1.075483,	0.950940,	1.056806,	0.939311,	0.767863,	1.050672,	0.923138,	1.208091,	0.957328,	0.914622,
0x7d	1.152850,	0.985329,	0.758860,	0.902903,	1.268339,	0.822622,	0.951593,	0.972154,	0.988798,	1.010038,
0x7e	0.934316,	1.213602,	1.137776,	0.530840,	0.583216,	1.327735,	0.648220,	1.122533,	0.983140,	0.947688,
0x7f	1.096461,	1.329950,	1.056696,	1.096312,	0.518753,	0.667957,	1.377914,	0.802439,	0.713627,	0.728387,
0x80	0.438479,	0.986558,	0.795180,	1.048475,	1.212747,	1.336801,	0.859633,	1.040980,	0.943136,	0.921149,
0x81	1.106514,	0.482598,	0.612108,	1.642264,	0.798236,	0.780622,	0.825869,	0.883680,	1.122055,	0.975218,
0x82	0.977340,	0.929082,	0.748423,	0.688517,	1.866792,	0.978102,	0.704420,	0.691328,	0.814419,	0.732273,
0x83	2.061007,	1.317233,	0.531080,	0.663650,	0.466228,	0.673965,	0.663756,	0.680624,	0.722060,	0.735546,
0x84	0.324483,	0.566419,	1.626532,	0.909957,	0.767043,	1.218518,	0.923857,	0.781191,	1.034489,	0.929690,
0x85	1.132776,	0.664024,	0.658276,	1.115888,	1.058611,	0.773592,	1.143017,	1.345343,	0.885859,	0.793138,
0x86	0.693193,	0.957946,	1.074530,	0.991271,	1.178869,	1.050262,	1.231718,	0.919607,	0.838514,	0.820971,
0x87	0.981206,	1.282808,	0.628439,	0.831905,	0.871888,	1.000360,	1.072068,	1.359583,	0.865264,	0.714229,
0x88	0.444708,	1.320901,	0.863009,	0.823633,	1.282252,	1.175766,	1.016159,	0.800445,	0.885734,	0.918392,
0x89	0.524278,	0.903183,	0.842282,	0.999358,	1.663203,	0.862494,	0.990065,	0.863742,	0.861246,	0.885159,
0x8a	0.647517,	1.630489,	0.674425,	0.552789,	1.367019,	0.802523,	1.078554,	0.810538,	0.843750,	0.824923,
0x8b	1.157635,	1.168651,	0.661005,	0.556302,	1.240422,	0.970912,	0.829631,	1.237884,	0.917322,	0.831318,
0x8c	0.271398,	0.978008,	0.699287,	1.501347,	0.903792,	1.105029,	0.960131,	1.036695,	1.001109,	0.890359,
0x8d	0.275143,	0.628587,	1.143603,	1.112293,	1.315251,	1.071179,	0.889841,	0.954181,	1.018312,	0.964080,
0x8e	0.659209,	1.447333,	0.894087,	1.001992,	0.839399,	1.449942,	0.796803,	0.843993,	0.756016,	0.817636,
0x8f	0.853138,	1.064738,	0.855133,	0.742170,	0.868512,	1.370756,	0.989681,	1.038327,	0.937196,	0.993200,
0x90	0.973539,	1.188932,	1.023116,	0.931037,	1.059522,	0.699512,	0.709015,	1.360161,	0.889684,	0.811761,
0x91	1.226588,	1.022695,	0.719215,	1.364611,	0.676914,	0.710196,	1.255418,	1.207082,	0.643748,	0.569250,
0x92	0.911359,	1.344114,	1.042414,	0.518823,	1.545711,	0.394875,	0.583463,	1.042424,	0.973970,	0.781663,
0x93	1.728382,	0.630457,	0.384633,	0.410263,	0.623340,	0.756137,	0.963416,	1.240436,	1.179740,	0.957703,
0x94	0.738959,	0.590931,	1.110580,	1.101743,	0.983178,	0.969963,	1.084357,	1.136055,	1.021410,	0.944931,
0x95	1.244427,	0.569958,	1.057313,	1.369461,	0.723367,	0.805159,	1.027868,	1.091877,	0.873967,	0.776658,
0x96	0.787743,	0.905228,	1.203273,	0.843008,	1.182612,	1.029414,	0.763719,	1.097128,	0.900016,	1.016990,
0x97	1.314427,	1.012982,	0.695385,	0.878737,	0.749022,	0.792771,	1.048592,	1.421864,	0.816344,	0.837093,
0x98	0.284264,	1.037450,	0.783668,	1.045137,	1.085681,	0.925955,	1.010468,	0.963587,	1.285465,	1.083612,
0x99	0.182413,	0.501944,	0.613194,	0.864945,	1.046530,	1.201125,	1.222115,	1.201816,	1.144754,	1.100226,
0x9a	0.399720,	1.302627,	1.339572,	0.544091,	1.113712,	0.704227,	0.959569,	1.086265,	0.919326,	0.996120,
0x9b	1.312087,	1.234779,	1.155127,	0.573204,	0.603170,	1.675500,	0.578462,	0.549438,	0.586763,	0.627105,
0x9c	0.347498,	1.019206,	0.902255,	0.918363,	1.078096,	0.895444,	1.295650,	1.236476,	0.896594,	0.955950,
0x9d	0.228297,	0.547356,	1.316811,	1.334579,	0.850478,	0.739122,	1.162684,	1.153175,	0.961239,	0.847730,
0x9e	0.450951,	1.164991,	1.356497,	0.748838,	1.098313,	1.180630,	0.960353,	0.882357,	0.871493,	0.801907,
0x9f	1.101005,	1.177952,	1.577841,	0.463778,	0.535020,	0.573959,	0.857461,	1.214102,	0.824723,	0.827632,
0xa0	1.008192,	0.512187,	0.511630,	0.803656,	1.363784,	1.397129,	1.041883,	0.945399,	0.880500,	0.836761,
0xa1	1.333032,	0.522169,	0.512363,	0.626419,	0.780427,	0.951713,	1.495366,	1.235918,	0.888195,	0.836141,
0xa2	1.680621,	0.528954,	0.445504,	0.623836,	1.316133,	1.053039,	0.890863,	0.877480,	0.775161,	0.883213,

0xa3	2.235511,	0.589895,	0.429925,	0.577054,	0.680209,	0.667395,	0.784315,	0.803936,	0.861809,	0.801209,
0xa4	1.092796,	0.555938,	1.481165,	0.731586,	1.165349,	0.972450,	0.847847,	0.832928,	0.893096,	0.925555,
0xa5	1.276456,	0.719658,	0.981702,	1.037082,	1.176864,	0.684247,	0.782879,	0.891723,	1.117659,	1.034450,
0xa6	1.228245,	0.771626,	1.121788,	0.749934,	0.884593,	0.958108,	1.233677,	1.045060,	0.933820,	0.833295,
0xa7	1.469187,	0.712424,	0.977325,	0.707326,	1.237597,	0.953937,	0.835168,	0.794138,	0.991406,	0.934207,
0xa8	1.079164,	1.065143,	1.263724,	0.575489,	1.628970,	0.516204,	0.546497,	0.527023,	0.774807,	1.079119,
0xa9	1.337945,	0.994695,	0.675169,	0.865744,	1.373516,	1.324911,	0.866779,	0.633895,	0.599425,	0.727921,
0xaa	1.511212,	0.641169,	1.710560,	0.731522,	0.590315,	0.546117,	0.624898,	0.778970,	0.908906,	0.903649,
0xab	1.760845,	1.236091,	1.067203,	0.487204,	0.655720,	0.713362,	0.697764,	0.749475,	0.847174,	0.921298,
0xac	1.242002,	1.065400,	0.986706,	0.626575,	1.306310,	0.681929,	0.717473,	0.615010,	1.350818,	0.832838,
0xad	1.114715,	0.651334,	1.082160,	1.095767,	1.064034,	1.122643,	1.004475,	0.903535,	0.874168,	0.865362,
0xae	1.244497,	1.005714,	1.193496,	0.734583,	0.598761,	1.274922,	1.289539,	0.614727,	0.688696,	0.778442,
0xaf	1.408250,	0.839767,	1.067828,	0.572007,	0.672582,	1.239141,	0.865001,	0.858314,	1.080318,	0.941161,
0xb0	1.250488,	0.918235,	1.101326,	0.668242,	1.365644,	0.555295,	0.552784,	1.259466,	0.907460,	0.811412,
0xb1	1.631045,	0.643427,	0.448596,	0.462433,	0.721966,	1.376064,	1.149863,	0.918544,	0.813496,	0.874574,
0xb2	1.535075,	1.109493,	1.002904,	0.867410,	1.515072,	0.754291,	0.569348,	0.512821,	0.594932,	0.664862,
0xb3	1.968050,	0.717136,	1.151858,	0.741070,	0.764881,	0.686026,	0.715694,	0.699206,	0.765000,	0.778737
0xb4	1.413788,	0.795490,	1.377022,	0.733397,	1.230761,	0.728025,	0.815985,	0.866154,	0.713555,	0.821627,
0xb5	1.339077,	0.710095,	1.014431,	1.092420,	0.988468,	0.688934,	1.213969,	0.865531,	0.928889,	0.852068,
0xb6	1.172569,	0.971105,	1.147345,	0.503528,	1.077878,	1.006880,	0.936191,	1.024685,	0.978125,	0.897520,
0xb7	1.346133,	0.825702,	1.278602,	0.712776,	0.990801,	0.824307,	0.741063,	0.777009,	1.172757,	0.958583,
0xb8	0.838360,	1.061588,	1.184451,	0.806865,	1.445930,	0.716610,	1.122729,	0.763437,	0.824729,	0.830950,
0xb9	1.276108,	0.881797,	1.013445,	0.821855,	1.264900,	1.025945,	1.227647,	0.703569,	0.659312,	0.734361,
0xba	1.043635,	1.011762,	1.513310,	0.739962,	0.660585,	0.587323,	0.680578,	0.841721,	1.167268,	1.132996,
0xbb	1.473979,	0.627258,	1.381177,	0.659284,	0.759202,	1.049849,	1.020305,	0.870139,	0.779335,	0.834162,
0xbc	1.308547,	0.926336,	1.119570,	0.550398,	1.232051,	0.605843,	1.181402,	0.819554,	0.865953,	0.920083,
0xbd	1.337645,	0.769449,	1.229177,	0.813618,	0.735630,	0.883470,	1.569295,	0.634285,	0.606052,	0.688148,
0xbe	1.290794,	1.093894,	1.145434,	0.552063,	0.804350,	0.769436,	0.840595,	1.089482,	1.009731,	1.051359,
0xbf	1.364932,	0.913519,	1.316632,	0.718362,	0.529225,	0.723872,	1.181281,	0.876150,	0.962565,	0.918262,
0xc0	0.803741,	1.388152,	0.980932,	1.386743,	0.402886,	1.229764,	0.855613,	0.743859,	0.738745,	0.812385,
0xc1	1.187083,	1.328233,	0.750736,	1.808456,	0.498683,	0.428506,	0.498878,	0.815022,	0.756569,	0.712230,
0xc2	1.008780,	1.634667,	0.998899,	1.114864,	1.106568,	0.814867,	0.677242,	0.626767,	0.727848,	0.701162,
0xc3	1.442123,	1.578969,	1.235055,	1.279799,	0.604508,	0.459123,	0.486663,	0.548079,	0.537477,	0.639041,
0xc4	0.824699,	0.832512,	1.286809,	0.946005,	0.788899,	1.581450,	0.819352,	0.804348,	0.757910,	0.797158,
0xc5	1.173084,	0.852089,	0.837159,	1.215371,	0.614107,	0.664921,	0.798941,	1.185092,	1.404650,	0.752108,
0xc6	0.837980,	1.195548,	1.251800,	1.166862,	1.139151,	0.630036,	0.966812,	0.823920,	0.819578,	0.833867,
0xc7	1.284059,	1.360185,	0.903226,	1.260066,	0.869280,	0.794052,	0.767674,	0.784999,	0.770146,	0.814413,
0xc8	0.434909,	1.062061,	0.820721,	0.980006,	0.915903,	1.352711,	1.363232,	0.818512,	0.933755,	0.804752,
0xc9	0.140705,	0.369673,	0.761460,	1.721146,	1.281940,	0.814390,	0.770430,	0.912685,	1.029093,	0.894100,
0xca	0.381833,	1.386068,	0.895360,	0.756433,	0.795777,	0.665474,	1.098763,	1.438104,	1.037485,	0.828884,
0xcb	0.848084,	1.088554,	0.824684,	0.776900,	0.721938,	0.676660,	0.800316,	1.135752,	1.511374,	1.057070,
0xcc	0.675218,	1.083363,	1.015844,	1.396638,	0.903676,	0.983177,	0.888414,	1.062584,	0.890620,	0.821218,

0xcd	0.754970,	0.952752,	1.127383,	1.539338,	0.700672,	0.751028,	0.640102,	0.797817,	1.188132,	0.971816,
0xce	0.701999,	1.321737,	1.118923,	1.317673,	0.798997,	1.146361,	0.653062,	0.923909,	0.716226,	0.881342,
0xcf	0.916108,	1.387334,	0.769658,	1.200061,	0.617306,	1.096865,	0.828575,	1.077143,	0.950269,	0.767253,
0xd0	0.662335,	1.292372,	1.294051,	1.134329,	0.611097,	0.878492,	0.979876,	0.932184,	0.972586,	0.870668,
0xd1	1.081800,	1.056505,	1.431825,	1.568379,	0.608668,	0.601313,	0.533137,	0.646402,	0.768233,	0.839110,
0xd2	1.062379,	1.439992,	1.195124,	1.101291,	0.690349,	0.686199,	0.744970,	0.782839,	0.947571,	0.913290,
0xd3	1.302173,	1.079498,	1.059178,	0.961715,	0.721412,	0.548559,	0.490001,	0.649919,	1.527619,	0.917298,
0xd4	0.680923,	1.193324,	1.324460,	1.076160,	0.940757,	1.158478,	1.016504,	0.799201,	0.766968,	0.708056,
0xd5	1.013158,	0.953016,	1.286560,	1.218908,	0.634730,	1.131540,	0.788277,	0.928104,	0.871793,	0.842399,
0xd6	0.801060,	0.948640,	1.501946,	0.737009,	0.751623,	1.024697,	1.091762,	0.879438,	0.970670,	0.874395,
0xd7	1.322228,	1.272781,	1.083638,	0.879445,	0.649796,	0.924154,	0.851784,	0.949572,	0.890756,	0.876190,
0xd8	0.308295,	1.165975,	1.273902,	1.153456,	0.938787,	0.761951,	0.988352,	0.937937,	0.984994,	0.988803,
0xd9	0.307028,	0.833986,	1.891761,	1.166166,	0.526468,	0.545358,	0.772629,	0.838092,	0.849941,	1.006412,
0xda	0.724400,	1.661022,	1.407543,	0.659395,	0.621709,	0.664584,	0.781703,	0.915312,	0.912252,	0.868291,
0xdb	0.899730,	0.520003,	2.050203,	0.570193,	0.525624,	0.618870,	0.773975,	0.953262,	0.860057,	0.852288,
0xdc	0.564036,	1.194072,	1.054038,	1.309234,	0.682175,	1.184083,	1.261020,	0.720925,	0.837966,	0.682236,
0xdd	0.705321,	0.906931,	1.440192,	1.246568,	0.945792,	0.911725,	0.914322,	0.903469,	0.819844,	0.853160,
0xde	0.639259,	1.118893,	1.183435,	0.912825,	0.722814,	1.107206,	0.903505,	1.289695,	0.937470,	0.852877,
0xdf	0.833210,	1.220093,	1.668781,	0.922503,	0.998916,	0.696036,	0.765626,	0.779931,	0.724635,	0.754248,
0xe0	1.214346,	0.962383,	1.132890,	0.827035,	0.876470,	1.255731,	0.611676,	0.550693,	0.663031,	1.355367,
0xe1	1.790710,	0.452581,	0.576174,	0.955171,	0.930091,	0.858695,	1.025674,	0.874218,	0.875633,	0.842868,
0xe2	1.436801,	0.888196,	1.189311,	0.679367,	1.104152,	1.351368,	0.585875,	0.677641,	0.703740,	0.775152,
0xe3	1.614636,	0.783579,	0.883180,	1.156453,	0.973477,	0.848164,	0.830311,	0.892666,	0.763745,	0.831350,
0xe4	1.372060,	0.499597,	1.000811,	0.900767,	0.931695,	0.898965,	1.028259,	1.067040,	0.960141,	0.995706,
0xe5	1.464549,	0.744441,	0.906385,	0.936059,	0.731646,	0.716360,	0.829522,	1.117880,	1.156979,	1.018394,
0xe6	1.367898,	0.866499,	1.263666,	1.112268,	1.167850,	0.937974,	0.788936,	0.718079,	0.635187,	0.707080,
0xe7	1.439445,	0.795863,	0.796972,	1.211896,	0.538767,	1.255454,	0.958684,	0.857981,	0.848545,	0.824913,
0xe8	1.376381,	0.568678,	0.926291,	0.641770,	0.834535,	1.808678,	0.804428,	0.607970,	0.669675,	0.720965,
0xe9	1.355563,	0.460449,	0.596175,	1.214770,	0.946022,	1.352957,	0.860313,	0.864696,	0.785836,	0.918103,
0xea	1.435024,	0.909544,	1.041289,	1.123733,	0.794294,	1.439099,	0.648301,	0.574374,	0.602775,	0.780902,
0xeb	1.516270,	0.918985,	0.927275,	1.189389,	0.590144,	0.748748,	1.248918,	0.683386,	0.755543,	0.882486,
0xec	1.476744,	0.514611,	1.218581,	1.065365,	0.862836,	0.922786,	0.764142,	0.839729,	0.935193,	0.933221,
0xed	1.315547,	0.722841,	1.311024,	1.315785,	0.913893,	0.706439,	0.682783,	0.788285,	0.877441,	0.888931,
0xee	1.348712,	0.858988,	1.055635,	1.052220,	0.611431,	1.171928,	0.730139,	1.115700,	0.744873,	0.946277,
0xef	1.180305,	1.053248,	0.878721,	1.288189,	0.507336,	1.103290,	0.705925,	1.007651,	0.841461,	1.044969,
0xf0	1.196638,	1.080029,	1.301461,	0.997280,	0.885975,	0.846955,	0.887870,	0.844710,	0.873575,	0.870310,
0xf1	1.373492,	1.068929,	1.077698,	1.096515,	0.998252,	0.717247,	1.234505,	0.694028,	0.663647,	0.627234,
0xf2	1.437332,	1.207382,	1.145016,	0.902884,	0.999814,	1.049959,	0.751617,	0.679969,	0.707880,	0.681629,
0xf3	1.540923,	0.966825,	1.074880,	0.967647,	0.780575,	0.824665,	0.721806,	0.771586,	0.917974,	1.059911,
0xf4	1.075058,	0.874560,	1.223286,	0.875247,	1.114946,	0.785841,	1.030228,	1.006401,	0.939058,	0.894774,
0xf5	1.230599,	0.763269,	1.223421,	1.021346,	0.751078,	0.863709,	0.928759,	0.989248,	1.037576,	0.975537,
0xf6	1.200394,	0.826452,	1.374917,	0.728714,	0.749645,	1.169742,	0.809933,	1.098679,	0.840553,	0.806698,

0xf7	1.458186,	0.835033,	1.120633,	0.933190,	0.831612,	1.053974,	1.060913,	0.785122,	0.845646,	0.769115,
0xf8	1.087334,	1.237373,	1.295267,	0.657919,	1.094758,	1.035074,	0.693941,	0.869327,	0.868297,	0.787626,
0xf9	1.183917,	0.792976,	1.143049,	0.749982,	0.626687,	0.658678,	0.983599,	1.416025,	1.044253,	0.931234,
0xfa	1.694116,	1.108393,	1.504700,	0.828677,	0.586589,	0.658925,	0.648047,	0.676080,	0.628926,	0.681370,
0xfb	1.363659,	0.859205,	1.134321,	1.227826,	0.519545,	0.570781,	0.847817,	1.035284,	0.967904,	0.993060,
0xfc	1.202407,	0.995001,	1.034142,	0.944991,	0.778536,	0.657893,	1.164819,	0.964661,	0.913810,	1.114029,
0xfd	1.325400,	0.953743,	1.087893,	1.003095,	0.754906,	0.862034,	1.119859,	1.201467,	0.736865,	0.616466,
0xfe	1.321790,	0.767809,	1.313064,	0.954328,	0.816200,	0.607967,	0.663671,	1.355803,	0.816644,	0.850276,
0xff	1.493994,	1.117090,	1.211209,	1.311853,	0.619853,	0.708056,	0.764515,	0.708914,	0.718302,	0.743372,