```
 1: #include <stdio.h>
 2: #include <stdlib.h>
 3: #include <math.h>
 4: #include <sf.h>
 5: #include <sfmisc.h>
 6:
 7: #include "env.h"
 8:
 9: void fm(short sound[], double dur, int sr, double dB,
10:         double carrier,
11:         double modulator,
12:         double peakDeviation)
13: {
14:     double cPhase, mPhase,      // carrier and modulator phase
15:         cInc, mInc,             // increments for the two
16:         amp ;                   // linear amplitude
17:     int t, samples ;            // counter for time, and samples
18:
19:     static int counter ;        // function counter
20:
21:     // arrays for the envelope: T = time, A = amplitude
22:     double T[] = {0.0, 0.2, 0.4, 0.6, 0.8, 1.0} ;
23:     double A[] = {0.0, 1.0, 0.75, 0.5, 1.0, 0.0} ;
24:     int nPoints = 6 ;   // number of points in the envelope
25:     double *env ;       // array for the envelope
26:
27:     // print out information to the screen
28:     fprintf(stderr,
29:         "\tFM(%d): C = %g  M = %g  pd = %g  dur = %g  dB = %g\n",
30:             counter++, carrier, modulator, peakDeviation,
31:             dur, dB) ;
32:
33:     samples = dur * sr ;            // translate seconds into samples
34:     amp = pow(10.0, dB/20.0) / 2 ;  // translate dB into linear
35:
36:     cPhase = 0 ;        // initialize the phase for both
37:     mPhase = 0 ;        // carrier and modulator
38:
39:     cInc = ( 2.0 * M_PI * carrier ) / sr ;  // increments for C & M
40:     mInc = ( 2.0 * M_PI * modulator ) / sr ;
41:
42:     env = linearEnv(dur, sr, T, A, nPoints) ;   // make envelope
43:
44:     for ( t = 0 ; t < samples ; t++ ) {     // FM loop
45:         sound[t] = amp * sin(cPhase
46:                 + (peakDeviation * env[t]) * sin(mPhase)) ;
47:         cPhase += cInc ;        // increment cPhase & mPhase
48:         mPhase += mInc ;
49:     }
50:
51:     Free(env) ;     // release memory
52:
53:     return ;
54: }
55:
```