```
function percussion(X, reR, imR, reL, imL, reC, imC, reA, imA) = {
  logRe     = log(magnitude(X))
  logIm     = phase(X)
  cepstrum  = IFFT(logRe, logIm)
  ceps1     = first-half-of (cepstrum)
  ceps2     = reverse(second-half-of(cepstrum))
  m1Re      = Re(ceps1) * reR + Re(ceps2) * reC
  m2Re      = Re(ceps2) * reL + Re(ceps1) * reA
  m1Im      = Im(ceps1) * imR + Im(ceps2) * imC
  m2Im      = Im(ceps2) * imL + Im(ceps1) * imA
  val mRe   = m1Re ++ reverse(m2Re)
  val mIm   = m1Im ++ reverse(m2Im)
  Y         = FFT(mRe, mIm)
  magOut    = exp(Re(Y))
  phaseOut  = Im(Y)
  polarToCartesian(magOut, phaseOut)
}
```