Syntax

```
[n,Wn] = buttord(Wp,Ws,Rp,Rs)
[n,Wn] = buttord(Wp,Ws,Rp,Rs,'s')
```

Description

[n,Wn]=buttord(Wp,Ws,Rp,Rs) returns the lowest order, n, of the digital Butterworth filter with no more than Rp dB of passband ripple and at least Rs dB of attenuation in the stopband. Wp and Ws are respectively the passband and stopband edge frequencies of the filter, normalized from 0 to 1, where 1 corresponds to π rad/sample. The scalar (or vector) of corresponding cutoff frequencies, Wn, is also returned.

[n,Wn]=buttord(Wp,Ws,Rp,Rs,'s') finds the minimum order n and cutoff frequencies Wn for an analog Butterworth filter. Specify the frequencies Wp and Ws in radians per second. The passband or the stopband can be infinite.

To design a Butterworth filter, use the output arguments n and Wn as inputs to butter.

Syntax

```
[b,a] = butter(n,Wn)
[b,a] = butter(n,Wn,ftype)
[z,p,k] = butter(_)
[A,B,C,D] = butter(_)
[_] = butter(_,'s')
```

Description

[b,a] = butter(n,Wn) returns the transfer function coefficients of an nth-order lowpass digital Butterworth filter with normalized cutoff frequency Wn.

[b,a] = butter(n,Wn,ftype) designs a lowpass, highpass, bandpass, or bandstop Butterworth filter, depending on the value of ftype and the number of elements of Wn. The resulting bandpass and bandstop designs are of order 2n.

[z,p,k] = butter(_) designs a lowpass, highpass, bandpass, or bandstop digital Butterworth filter and returns its zeros, poles, and gain. This syntax can include any of the input arguments in previous syntaxes.

[A,B,C,D] = butter(_) designs a lowpass, highpass, bandpass, or bandstop digital Butterworth filter and returns the matrices that specify its state-space representation.

[_] = butter(_,'s') designs a lowpass, highpass, bandpass, or bandstop analog Butterworth filter with cutoff angular frequency Wn.

Syntax

```
[bt,at] = lp2hp(b,a,Wo)
[At,Bt,Ct,Dt] = lp2hp(A,B,C,D,Wo)
```

Description

[bt,at] = lp2hp(b,a,Wo) transforms an analog lowpass filter prototype given by polynomial coefficients (specified by row vectors b and a) into a highpass analog filter with cutoff angular frequency Wo. The input system must be an analog filter prototype.

[At,Bt,Ct,Dt] = lp2hp(A,B,C,D,Wo) converts the continuous-time state-space lowpass filter prototype (specified by matrices A, B, C, and D) to a highpass analog filter with cutoff angular frequency Wo. The input system must be an analog filter prototype.