Remez Equi-Ripple Design

```python
# Load an equi-ripple design code module from GNU Radio
#(in notebook ZIP package)
import optfir
```

```python
d_pass = 0.2
d_stop = 60.0
fs = 48000
f_pass = 3500
f_stop = 5000
n, ff, aa, wts=optfir.remezord([f_pass,f_stop], [1,0],
                          [1-10**(-d_pass/20),10**(-d_stop/20)],
                          fsamp=48000)
# Bump up the order by 5 to bring down the final d_pass & d_stop
n_bump = n + 5
b1 = signal.remez(n_bump, ff, aa[0::2], wts,Hz=2)
```

Note: The original amplitude response requirements have been changed. The passband ripple is now 0.2 db and the passband critical frequency is reduced from 4000 to 3500 Hz. This reduces the filter order.

Fixed-Point Coefficients (`int16_t`)

```python
b1_fix = int16(rint(b1*2**15))
b1_fix
```

```
array([    14,    -13,    -33,    -63,    -97,   -124,   -134,   -118,    -72,
           -2,     80,    151,    191,    179,    109,     -9,   -148,   -269,
         -332,   -304,   -176,     32,    274,    481,    581,    520,    281,
         -103,   -549,   -936,  -1128,  -1009,   -512,    360,   1519,   2810,
         4036,   4993,   5518,   5518,   4993,   4036,   2810,   1519,    360,
         -512,  -1009,  -1128,   -936,   -549,   -103,    281,    520,    581,
          481,    274,     32,   -176,   -304,   -332,   -269,   -148,     -9,
          109,    179,    191,    151,     80,     -2,    -72,   -118,   -134,
         -124,    -97,    -63,    -33,    -13,     14], dtype=int16)
```

```python
FIR_fix_header('s4_p1_remez.h',b1)
```