Inverse Haar Transform

```
function x = ihaar(w, L)
    % setup
    J = log2(length(w));
    d_arr = w(2^(J-L)+1:end);
    g0 = [sqrt(1/2) sqrt(1/2)];
    g1 = [sqrt(1/2) -sqrt(1/2)];

    % initialization
    prevs = w(1, 1:2^(J-L));
    prevd = d_arr(1:2^(J-L));
    ld = length(prevd);

    % step
    for k = J - L:J - 1
        u = upsample(prevs, 2);
        u = conv(u, g0);
        v = upsample(prevd, 2);
        v = conv(v, g1);

        prevs = u + v;
        prevs = prevs(1:end-1);
        if k < J - 1
            % prevd = d_arr(2^(k)+1+1:2^(k)+length(prevs)+1);
            prevd = d_arr(ld+1:ld+length(prevs));
            ld = ld + length(prevd);
        end

    end

    % result
    x = prevs(1:end);
end
```

Published with MATLAB® R2017a