generateDipoleRotationMomentsTest

```
% create full scale rotation with 0.5° resolution and no tilt,
% return moments
% and corressponding angles theta
amp = 1e6;
tilt = 0;
res = 0.5;
[MFS, tFS] = generateDipoleRotationMoments(amp, 0, tilt, res);

% create same rotation but only a subset of angles N = 7
% with equal distances to each and another, return additionally index which
% reference to full scale
[M, t, idx] = generateDipoleRotationMoments(amp, 7, tilt, res);

% create shifted subset, shift by 22 positions in full scale theta,
% so with 0.5° resolution it is phase shift by 11°
[MSH, tSH, idxSH] = generateDipoleRotationMoments(amp, 7, tilt, res, 22);
```

Test 1: output dimensions

```
assert(isequal(size(MFS), [3 720]))
assert(isequal(size(tFS), [1 720]))
assert(isequal(size(M), [3 7]))
assert(isequal(size(t), [1 7]))
assert(isequal(size(idx), [1 7]))
assert(isequal(size(MSH), [3 7]))
assert(isequal(size(tSH), [1 7]))
assert(isequal(size(idxSH), [1 7]))
```

Test 2: down sampling

```
assert(isequal(MFS(:,idx), M))
assert(isequal(tFS(idx), t))
assert(isequal(MFS(:,idxSH), MSH))
assert(isequal(tFS(idxSH), tSH))
```

Test 3: phase shift

```
assert(isequal(tSH(1), 11))
assert(isequal(idx, idxSH - 22))
assert(isequal(MFS(:,idx + 22), MSH))
assert(isequal(tFS(idx + 22), tSH))
```

Published with MATLAB® R2020b