Contents


```
% create sensor array infos for size and position
% number of sensors at one edge
N = 8;

% sensor array edge length in mm
a = 2;

% relative postion of the sensor array to the center of a 3D cordinate
% system (z inverse)
p = [0; 0; 2];

% z offset, later used as sphere radius of a dipole which is placed in the
% center of the coordinate system
r = 2;

% generate coordinates grid
[X, Y, Z] = generateSensorArraySquareGrid(N, a, p, r);

% create a shift in same layer
p2 = [-2; 3; 2];
[X2, Y2, Z2] = generateSensorArraySquareGrid(N, a, p2, r);
```

Test 1: output dimensions

```
assert(isequal(size(X), [N N]))
assert(isequal(size(Y), [N N]))
assert(isequal(size(Z), [N N]))
```

Test 2: equal x and y distances

```
assert(isequal(diff(Y), diff(-X, [], 2)'))
```

Test 3: constant z distances

```
assert(all(Z == -(p(3) + r), 'all'))
```

Test 3: position shif in x and y direction

```
assert(isequal(X + p2(1), X2))
assert(isequal(Y + p2(2), Y2))
assert(isequal(Z, Z2))
```

Published with MATLAB® R2020b