# Method #4: Auto segment detection

Video: Method #4 (Auto segment detection)

**Action #1: The name of the text file “RegIcosahedron.txt” containing the vertices/coordinates for a regular icosahedron is typed into the “file name of import {x,y,z} or {i,f}” text input box (figure 1a).** This file is located in the main directory \EBiD 3D (CAD) containing the FEBID 3D CAD program. The format of this file is discussed in Method #1 (Basic CAD use), figure 16(e).

**Action #2: Press “Load {x,y,z}” button (figure 1b).**

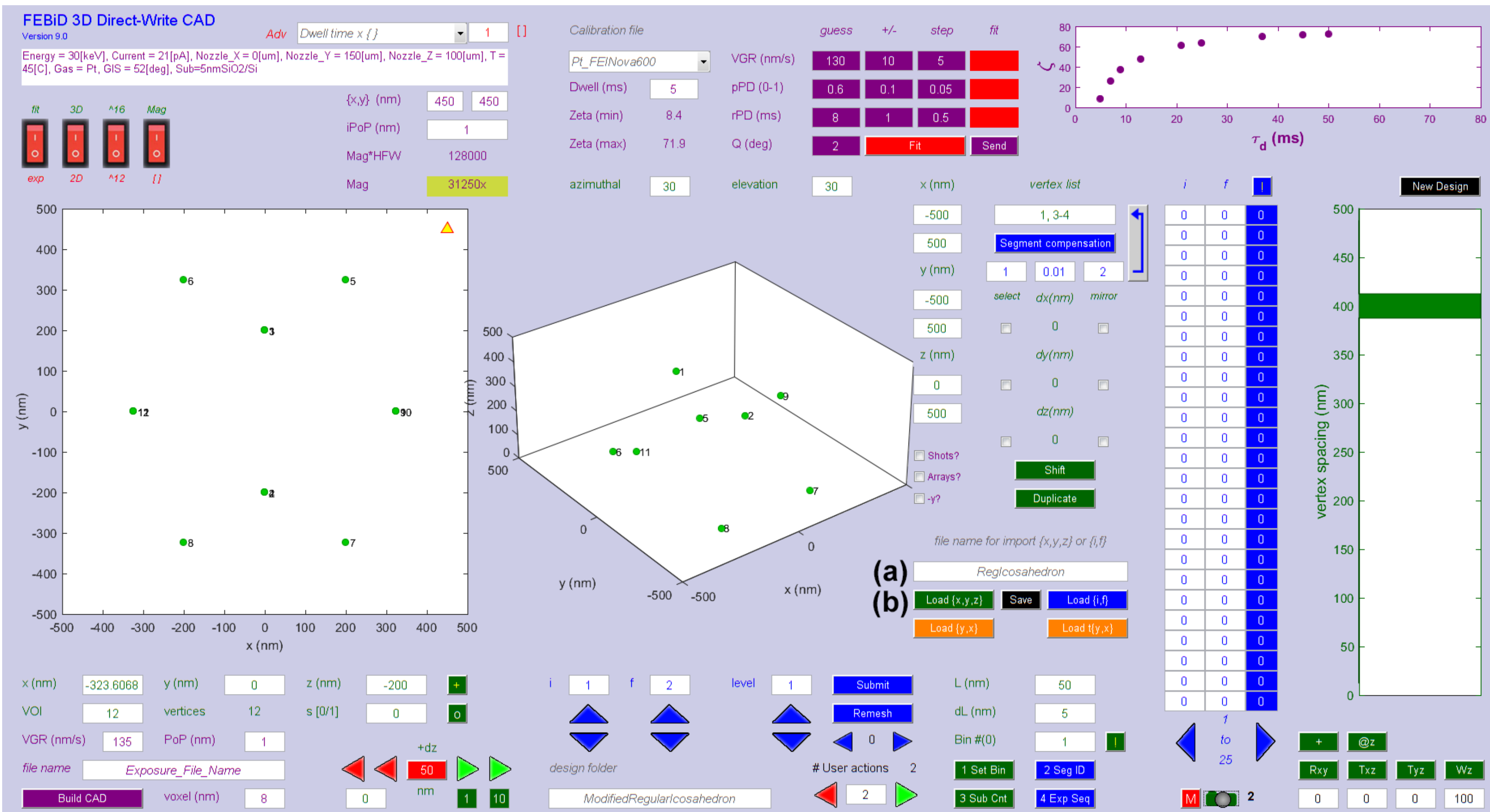



Figure 1

**Action #3: Enter “-500” into the “z (nm)” text input box (figure 2a).**

The z-axis limits need to be changed to see all the coordinates in the 3D plot.  The axis limits buttons “x (nm)”, “y (nm)” and “z (nm)” are organized in a column fashion, for each, the minimum limit is stacked above the maximum limit.  These limits apply to both the (x,y) and (x,y,z) vertex plots.

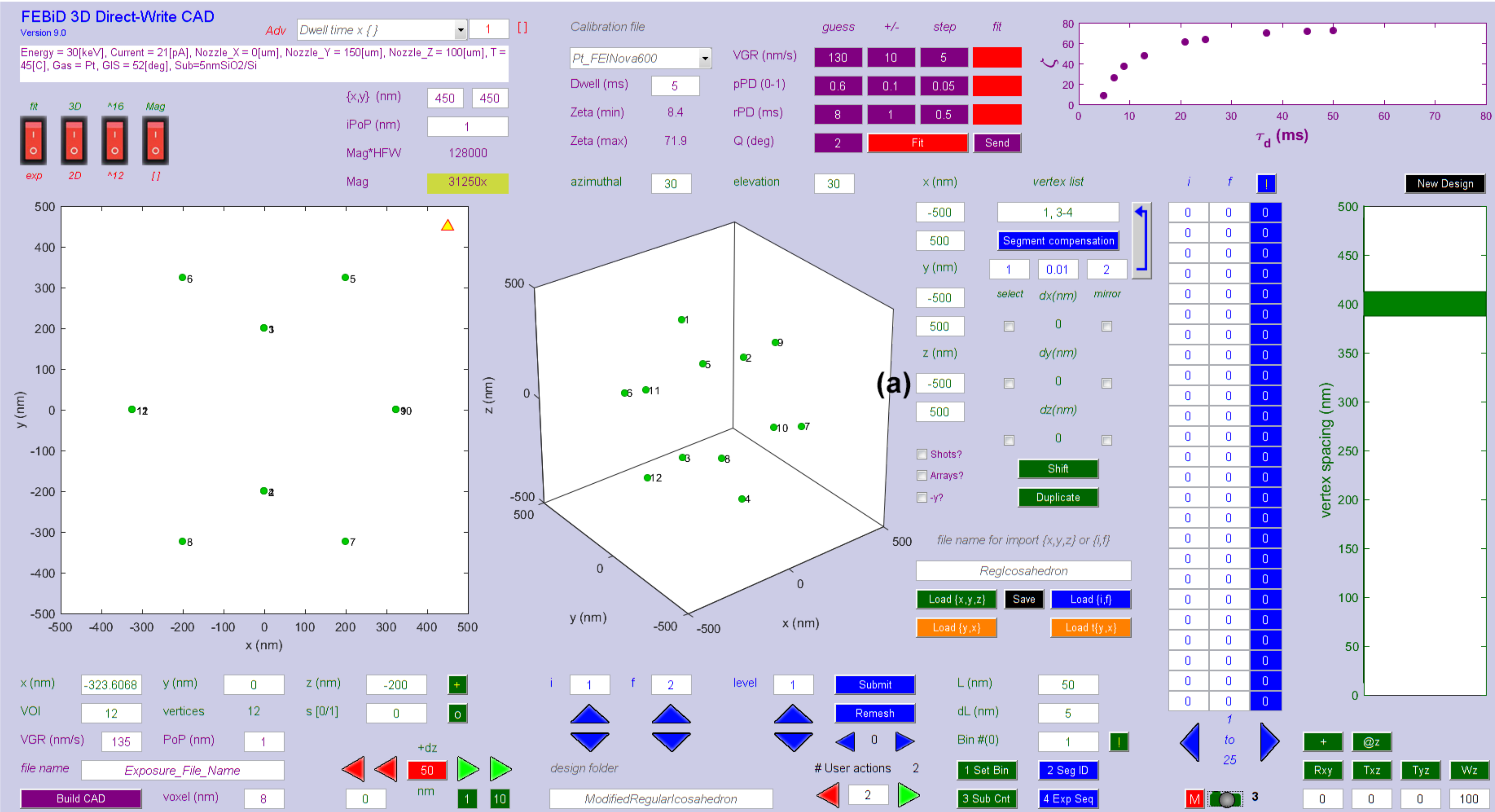



Figure 2

A tilt in x-z plane is next executed to align the vertices 9 and 12 both (x=0,y=0).  A tilt of 58.2825$^{o}$ in the x-z plane is required.

**Action #4: Enter “1-12” in the “vertex list” text input box (figure 3a).**

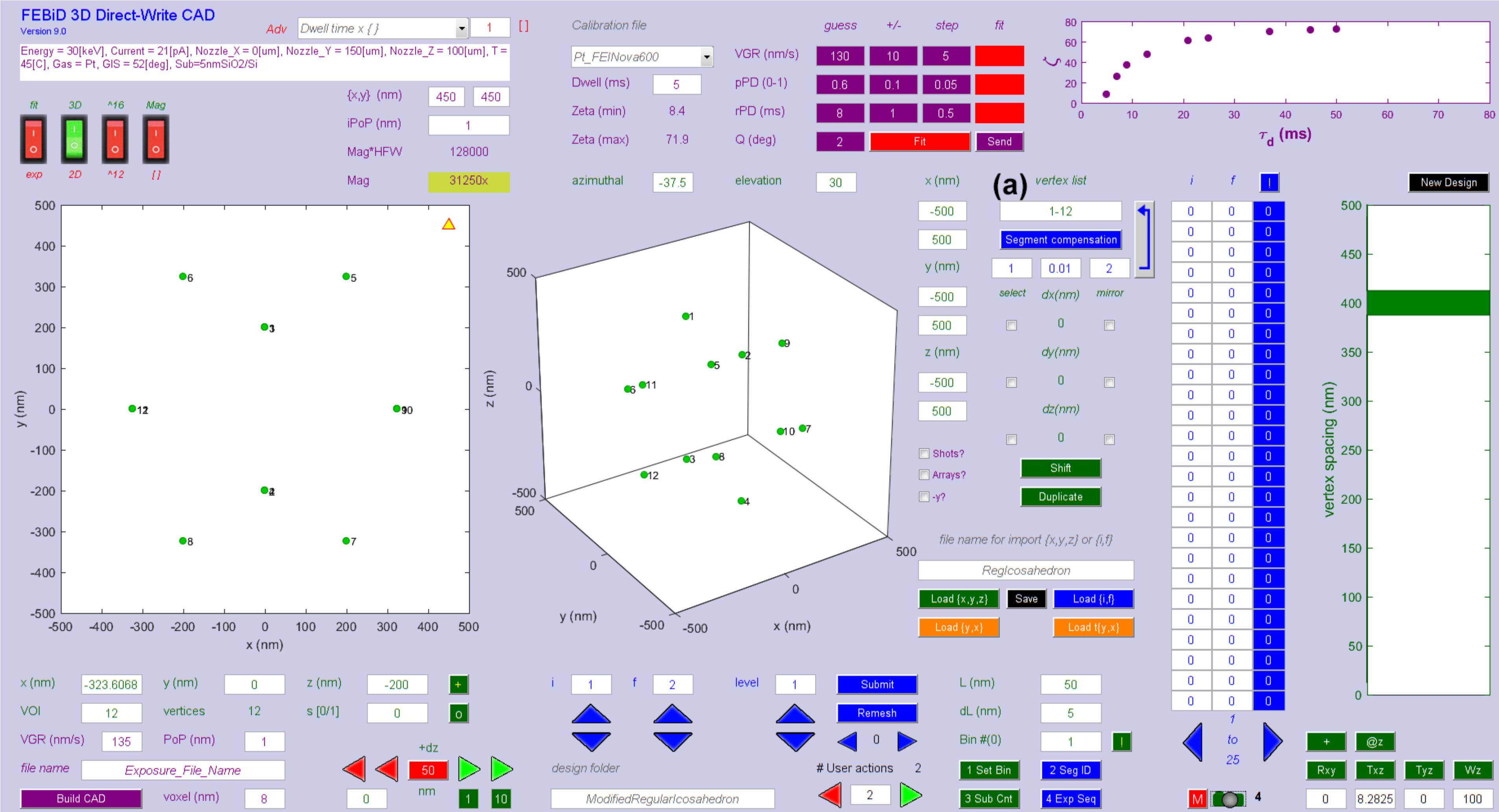



Figure 3

**Action #5: Enter "58.2825" degrees into the "Txz" (figure 4a).**

**Action #6: Press the "Txz" push button (figure 4b).**

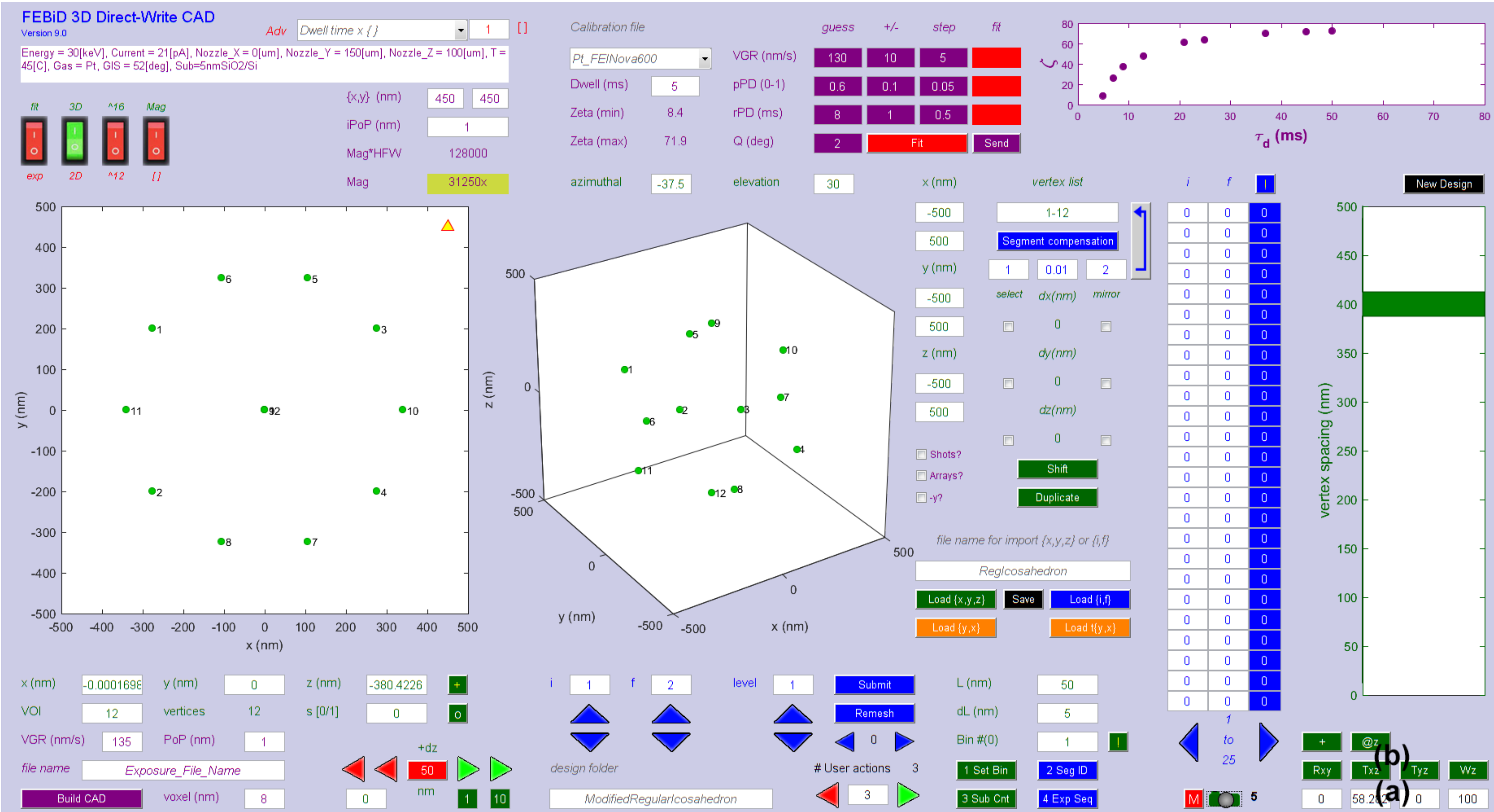
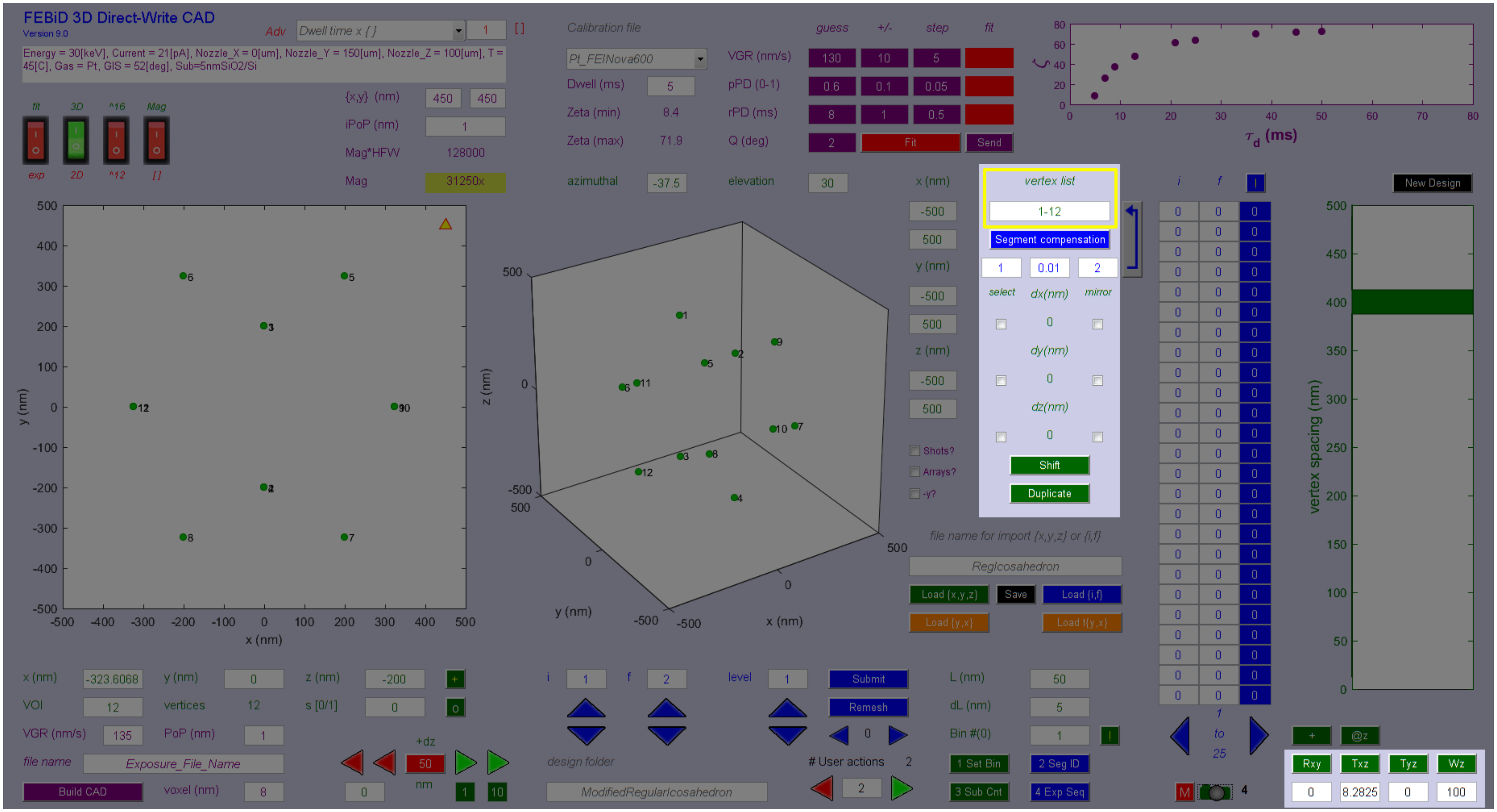


Figure 4

The 3D plot of vertices now shows that the coordinates #9 and #12 both rest at (x=0,y=0).

# “vertex list” details

Note: the highlighted regions on the GUI shown below indicate functions that require the specification of the “vertex list” prior to use




**“Rxy”**– counterclockwise rotation about the z–axis (enter value in units of degrees)

**“Txz** – counterclockwise rotation about the y–axis (enter value in units of degrees)

“**Tyz**”– counter clockwise rotation about the x–axis (enter value in units of degrees)

“**Wz**”– all vertices are (1) converted from (x,y,z) to (x,0,z) and then (2) wrapped onto the surface of a cylinder which points along the z-axis and is centered at (x=0,y=0). The text input box is the radius ($r_{Cyl}$) of the cylinder in nanometers. The complete transformation thus;

$$(x,y,z) = \left(r_{cyl} cos\frac{x}{r_{cyl}}, r_{cyl} sin\frac{x}{r_{cyl}}, z\right)$$

“**Shift**” – use “select” to activate one or more “dx(nm)”, “dy(nm)” or “dz(nm)” translations to be applied to the value(s) specified in “vertex list”. When “select” is checked an edit text box appears making it possible to enter the magnitude of the shift.

“**Duplicate**” – use “select” and/or “mirror” to activate one or more translations and/or mirroring (sign change) operations to the value(s) specified in “vertex list”. When “select” is checked an edit text box appears making it possible to enter the magnitude of the shift.

“**Segment Compensation**” – see Method #5 (Segment compensation)

**Action #7: Press the "@z" push button (figure 5a).**

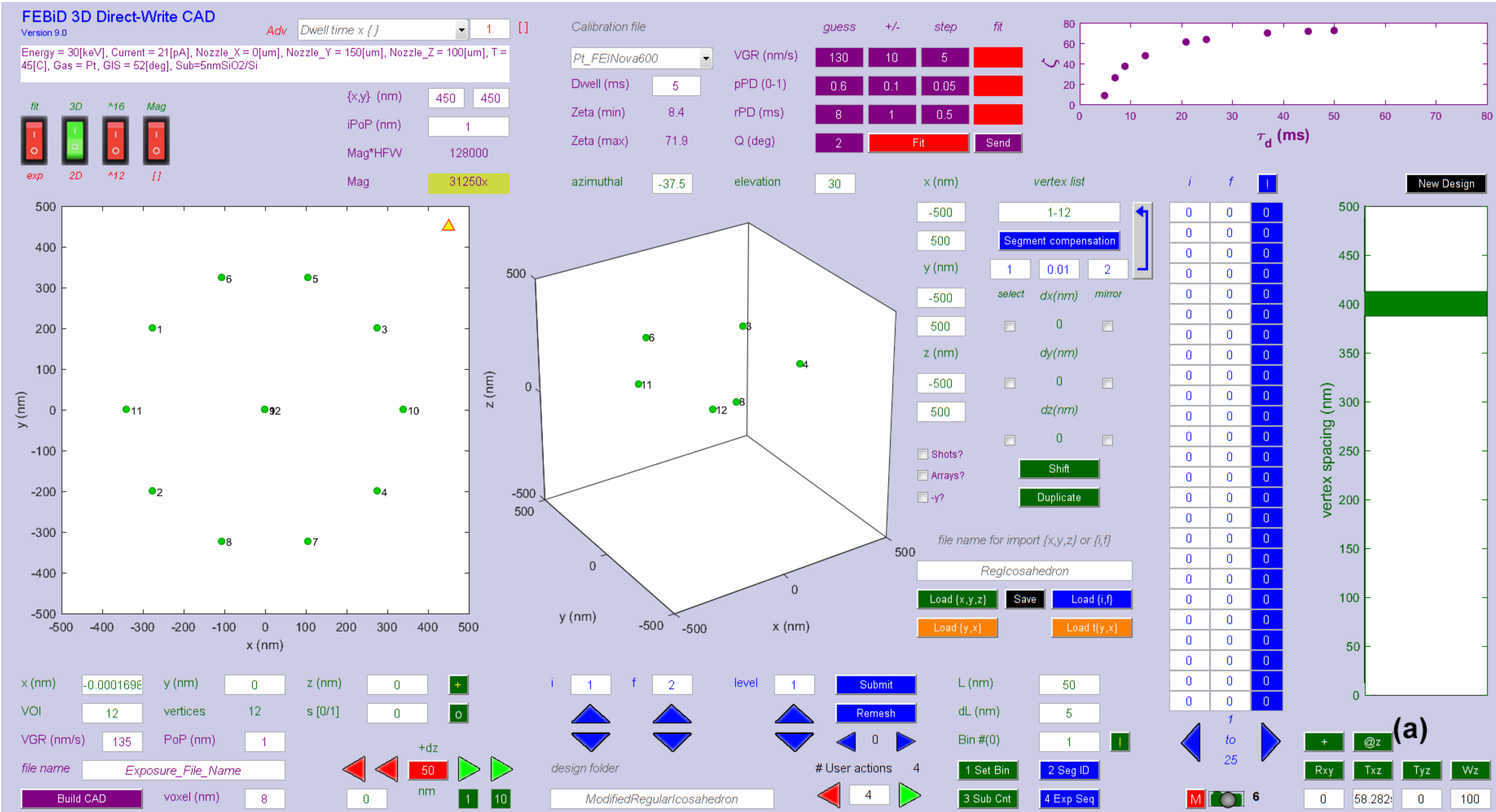



Figure 5

This action applies a z–shift to all coordinates which is equal, but opposite in sign, to the minimum z–value among all coordinates.

**Action #8: Enter “1000” into the maximum value text input box for “z (nm)” (figure 6a).**

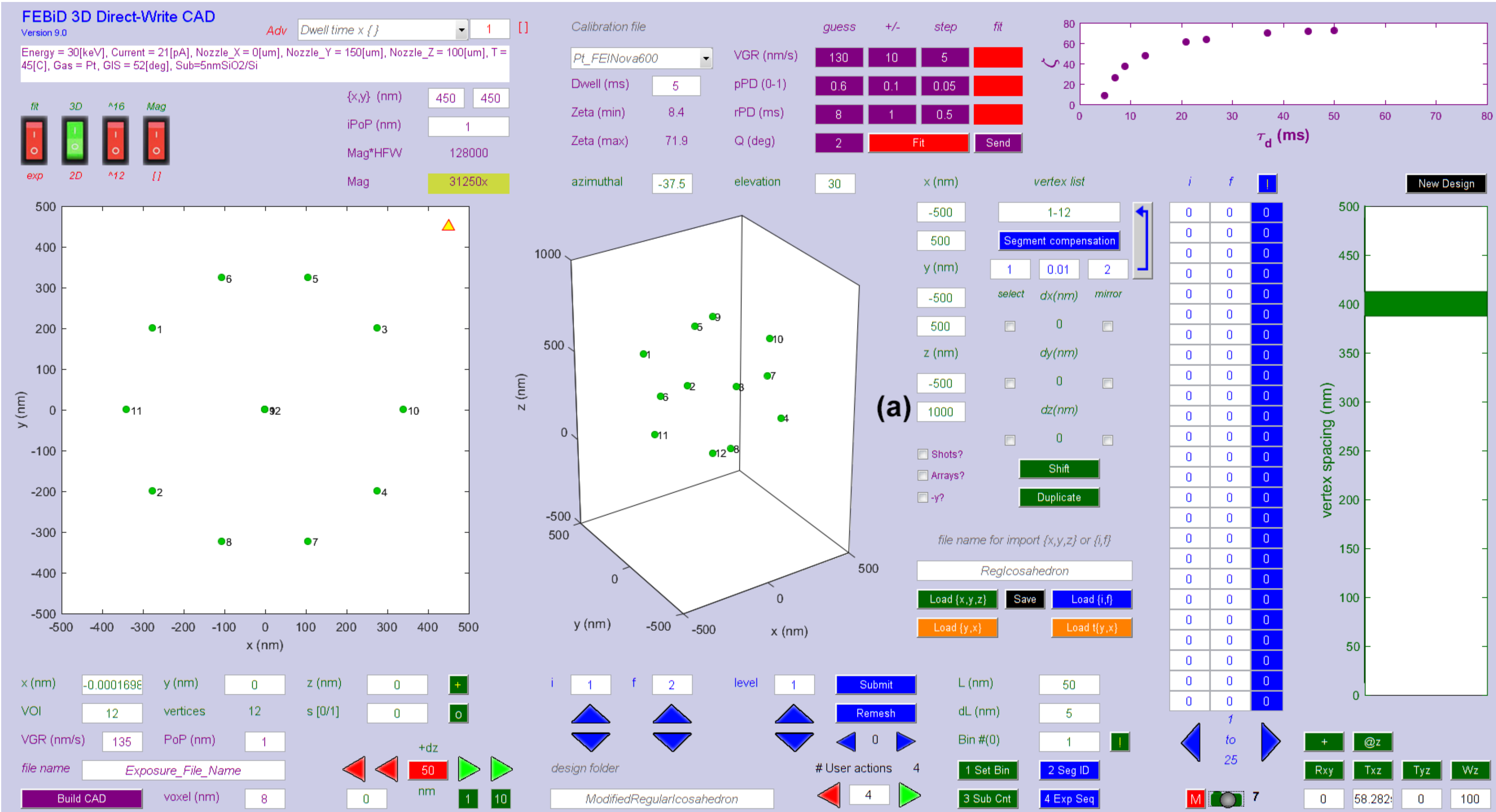



Figure 6

**Action #9: Enter “0” into the minimum value text input box for “z (nm)” (figure 7a).**

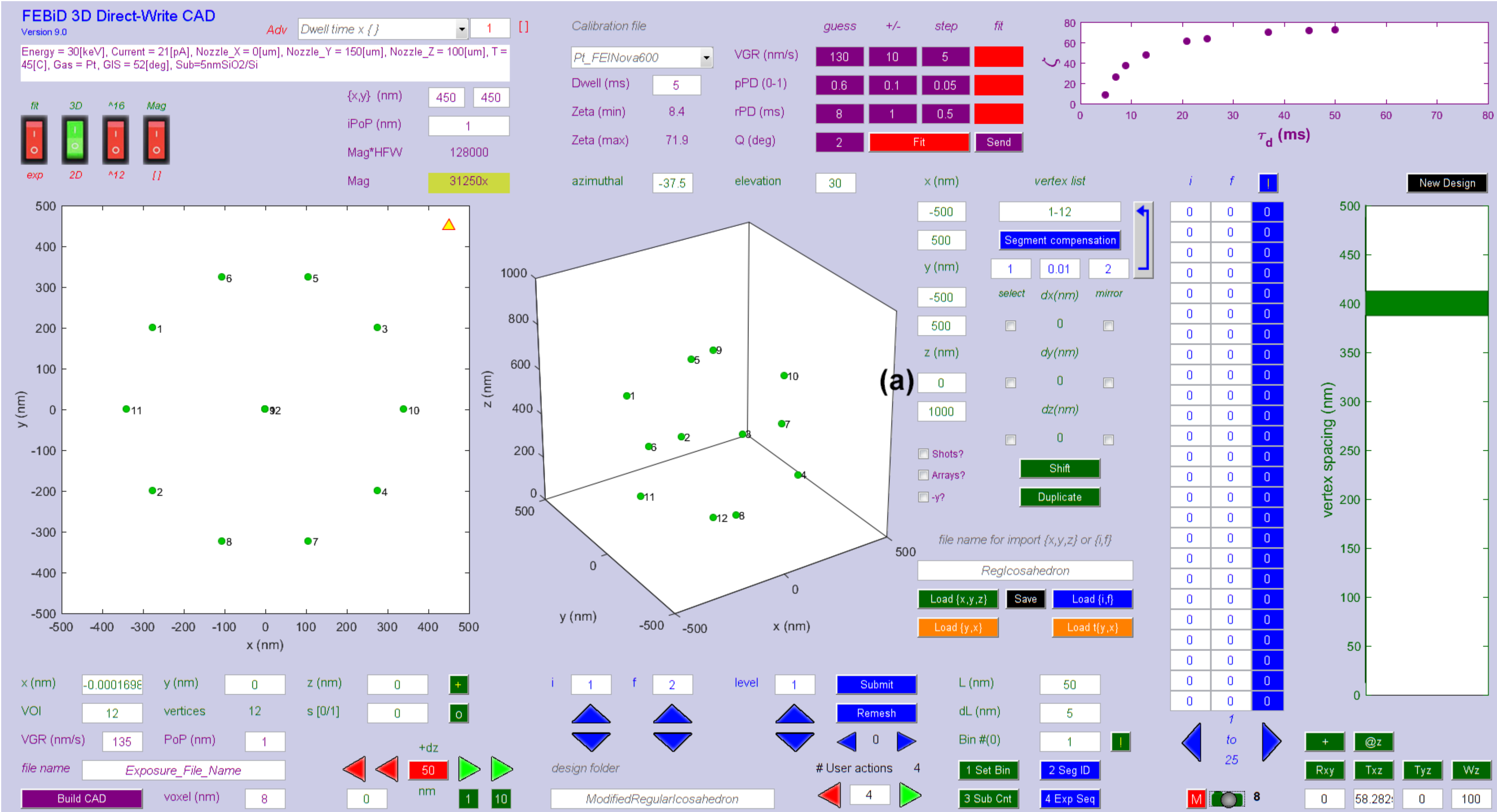



Figure 7

The regular icosahedron now rests on a “virtual” focal plane located at z = 0. *As a reminder, this position is not related to the substrate position in the actual microscope that will be used – it is only required that the User carefully focus the beam on the surface of interest. FEBID will ensue at the beam – substrate intersection regardless of the actual z – coordinate in the microscope coordinate system.*

**Action #10: Enter the segment length of interest “400” in the “L (nm)” text input box (figure 8a).**

**Action #11: Enter the segment length range “10” in the “dL (nm)” text input box (figure 8b).**

As a result, all permutations of vertex spacings will be searched to test if the spacing falling inside the range of;

$$(400 - 10) \leq L \leq (400 + 10) \ [nm]$$

The program can search for more than one segment range of interest. Therefore, the current segment range requires the definition of a unique index.

**Action #12: Enter a value of “1” for the contiguous segment search range in the “index#(0)” text input box (figure 8c).**

**Action #13: Press the “Set Bin” push button to submit the segment range (figure 8d).**

The input text box label will change from “index#(0)” to “index#(1)” which displays for the User the current number of segment ranges defined. If more than one segment range of interest has been submitted then the “index#(0)” text input box can be used to cycle between the segment ranges using the index if, e.g., “L (nm)” or “dL (nm)” need to be changed for a particular range. The current list of segment ranges can be erased by pressing the “!” push button located to the right of the “index#(0)” text input box.

Also when “Set Bin” is pressed, the vertex spacing histogram plot will refresh showing the identified segment range as a transparent yellow bin (figure 8f).

**Action #14: Press the “Seg ID” push button (figure 8f).**

This action launches as algorithm that identifies all unique segments which lengths spanning the range identified above.

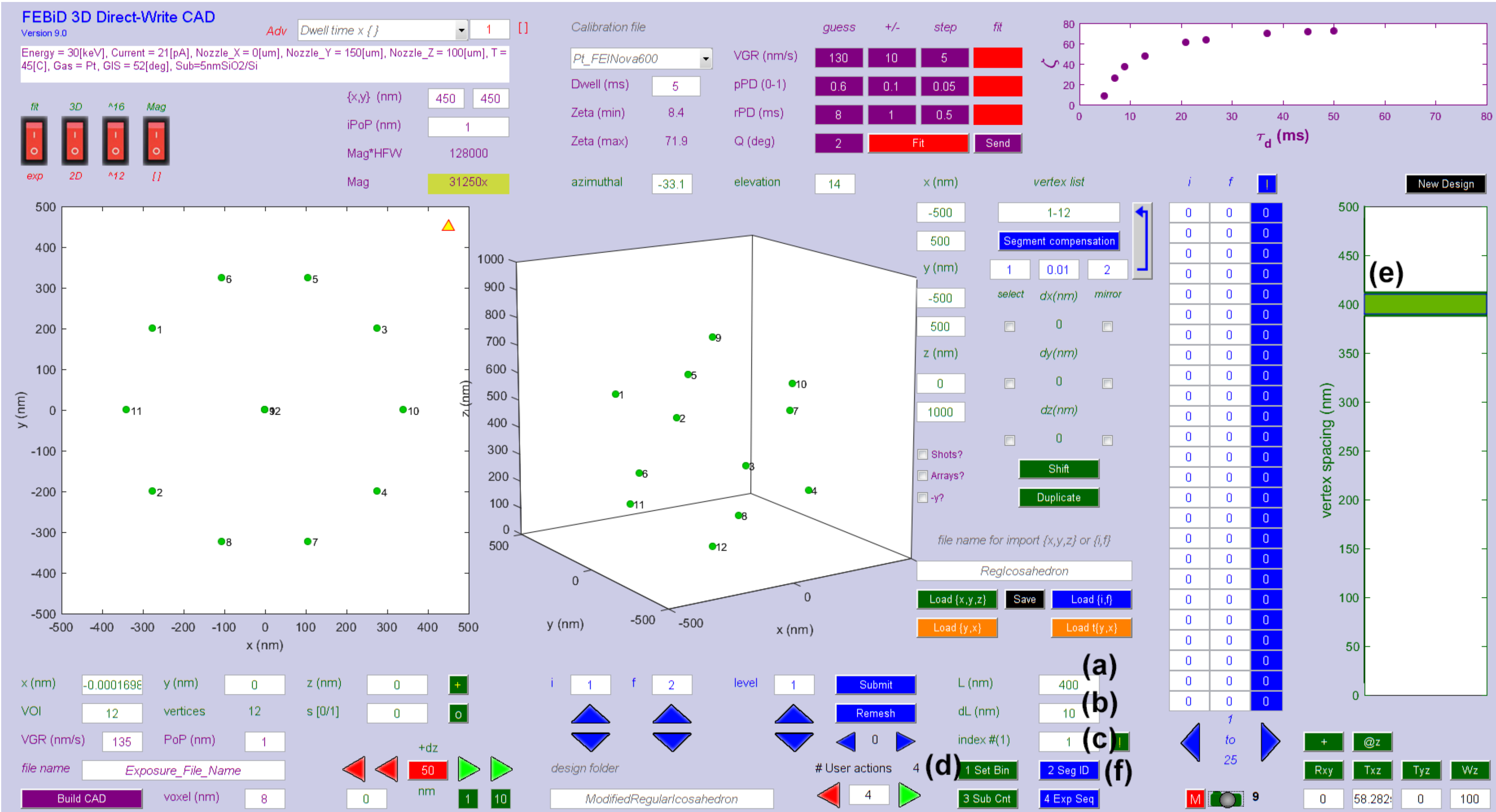

FEBiD 3D Direct-Write CAD
Version 9.0
Energy = 30[keV], Current = 21[pA], Nozzle_X = 0[um], Nozzle_Y = 150[um], Nozzle_Z = 100[um], T = 45[C], Gas = Pt, GIS = 52[deg], Sub=5nmSiO2/Si
Calibration file
Pt_FEINova600
guess
+/-
step
fit
VGR (nm/s)
pPD (0-1)
rPD (nm)
Q (deg)
Dwell (ms)
Zeta (min)
Zeta (max)
Fit
Send
{x,y} (nm)
iPoP (nm)
Mag*HFW
Mag
31250x
azimuthal
elevation
x (nm)
y (nm)
z (nm)
vertex list
Segment compensation
New Design
vertex spacing (nm)
(a)
(b)
(c)
(d)
(e)
(f)
Shift
Duplicate
file name for import {x,y,z} or {i,f}
RegIcosahedron
Load {x,y,z}
Save
Load {i,f}
Load {y,x}
Load t{y,x}
Submit
Remesh
L (nm)
dL (nm)
index #(1)
# User actions
1 Set Bin
2 Seg ID
3 Sub Cnt
4 Exp Seq
VOI
vertices
VGR (nm/s)
PoP (nm)
file name
Exposure_File_Name
Build CAD
voxel (nm)
design folder
ModifiedRegularIcosahedron
level
Rxy
Txz
Tyz
Wz

Figure 8

**Action #11: Enter "12" into the "vertex list" text input box (figure 9a).**

The vertex list should be updated to include all vertices that (1) will make contact with the substrate surface and (2) will define part of a segment in the deposit. In this example, the coordinates were rotated previously such that the 3D object would make contact with the substrate at vertex #12 only (figure 9b). A voxel with substrate contact is also indicated in the text edit input box labelled "s [0/1]" where a value of 0 indicates no substrate contact and a value of 1 indicates substrate contact. This parameter is only used with the automatic segment detection algorithm.

**Action #12: Press the "Sub Cnt" push button to submit the contact vertex/vertices lists specified in Action #11 (figure 9c).**

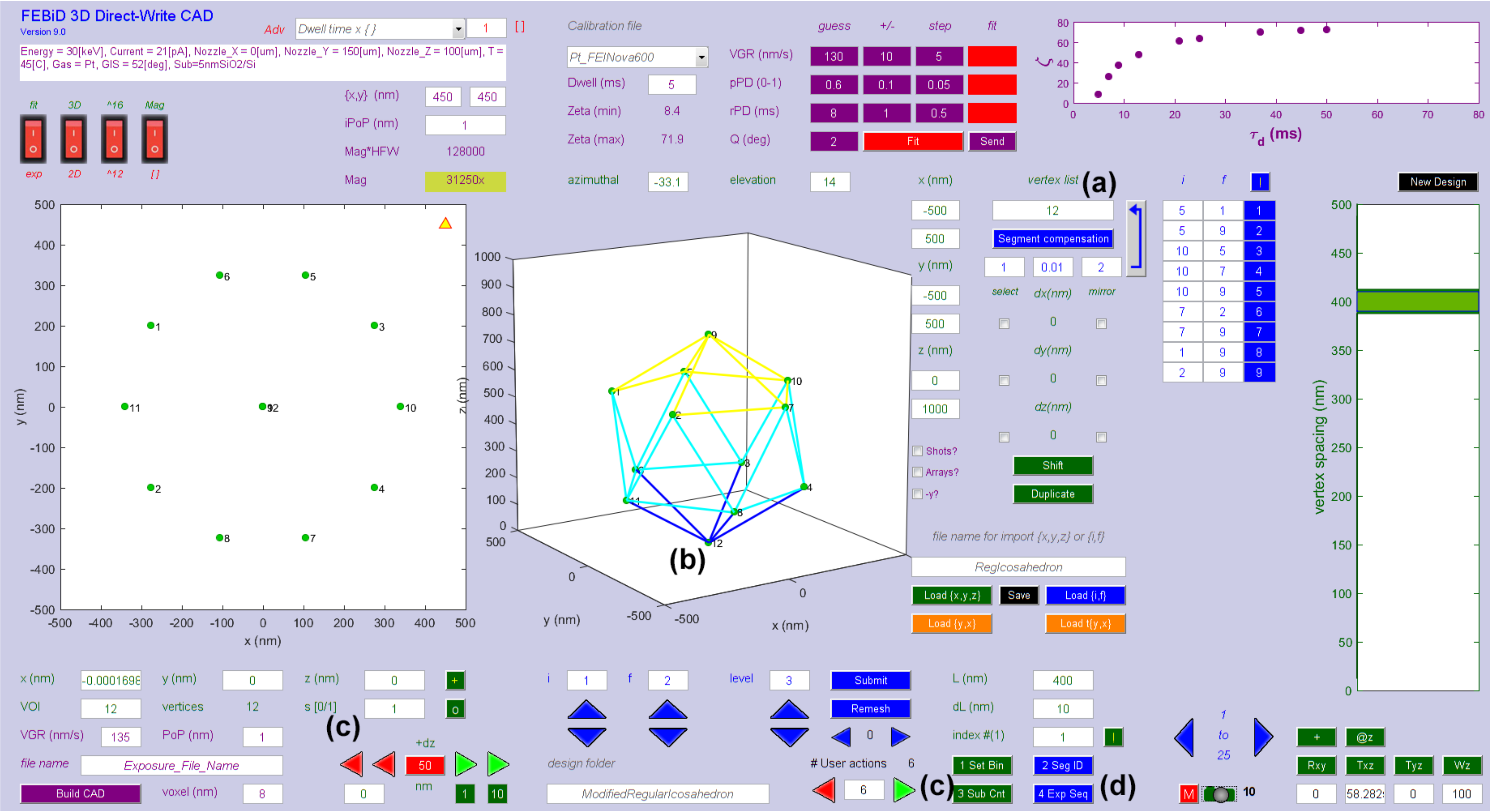



Figure 9

**Action #13: Press the "Exp Seq" push button (figure 9d).**

Starting from the substrate vertices indicated as contact points, the program estimates the exposure order by organizing the segments into exposure levels. The following constraints are applied;

(1) Primary electron beam–3D deposit contact must be preserved throughout growth
(2) Intermittent exposure is favored over continuous exposure sequences, if possible.

The 3D plot of vertices and segments updates to display the found segments and their exposure order. Exposure level is indicated by color and alternates through the following sequences of four colors;

$$Blue\ (level\ \#1), light\ blue\ (level\ \#2), yellow\ (level\ \#3), red\ (level\ \#4), blue\ (level\ \#5), light\ blue\ (level\ \#6), \ldots$$

In this example, it is further desired to remove all focal-plane parallel segments from the CAD object. This can be accomplished using the segment index list.

**Action #14: Select exposure level 2 in the "level" text input box (figure 10a).**

Exposure level #2 is shown as the light blue segments in the 3D plot of vertices and segments (figure 10b). The segment list (figure 10c) updates to reflect all current segments that are members of exposure level #2.

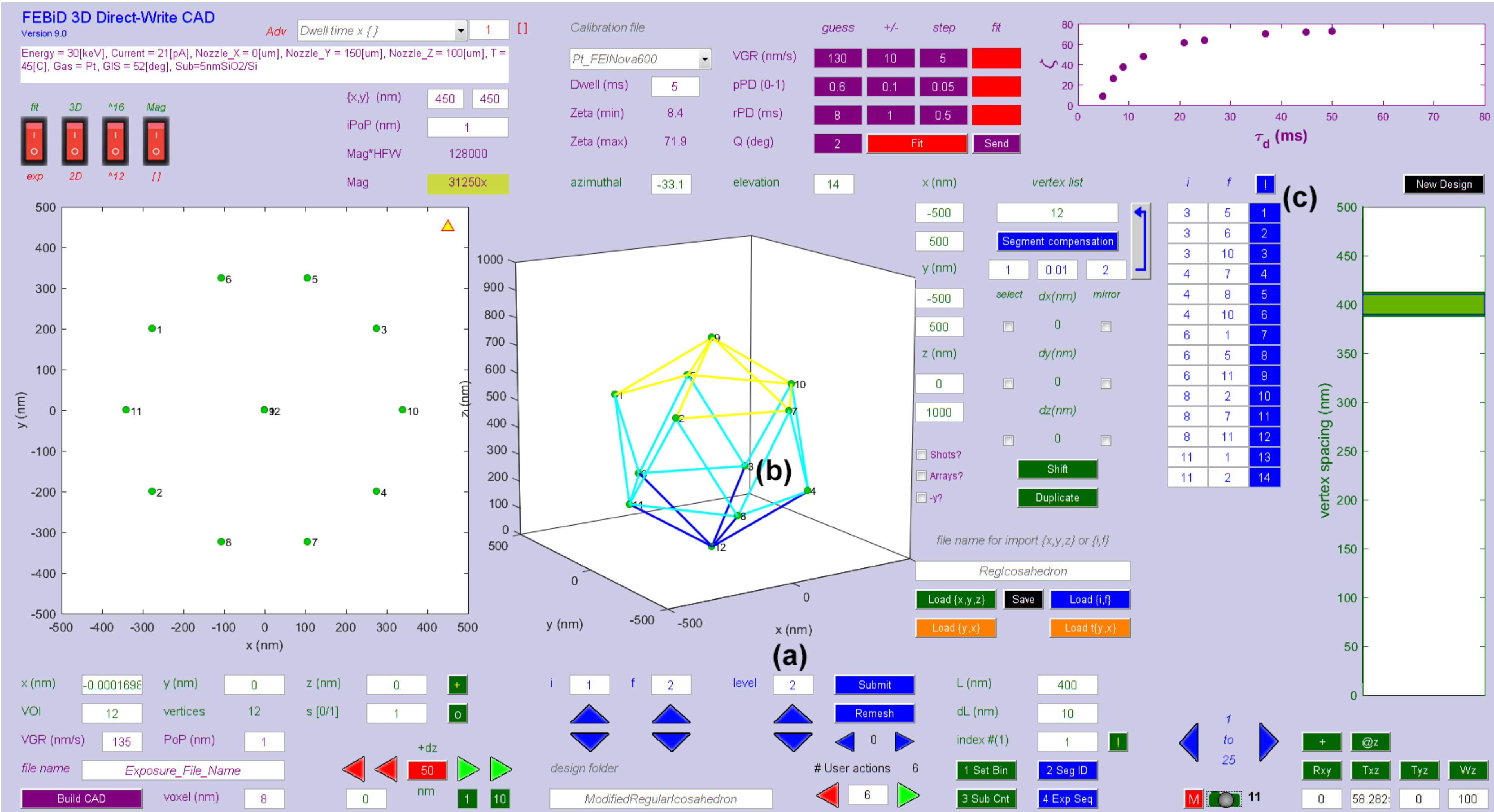



Figure 10

Each row in the segment list displays the characteristics of one segment. The segment characteristics, from left–to–right are;

$$\boldsymbol{i} = vertex\ index\ for\ initial\ position\ of\ the\ segment,\ \ \boldsymbol{f} = vertex\ index\ for\ the\ final\ position\ of\ the\ segment,\ \ \ \ unique\ segment\ index$$

**Action #15: Enter “0” for all segments to be removed from the CAD, in place of their current segment index (figure 11a, *arrows*).**

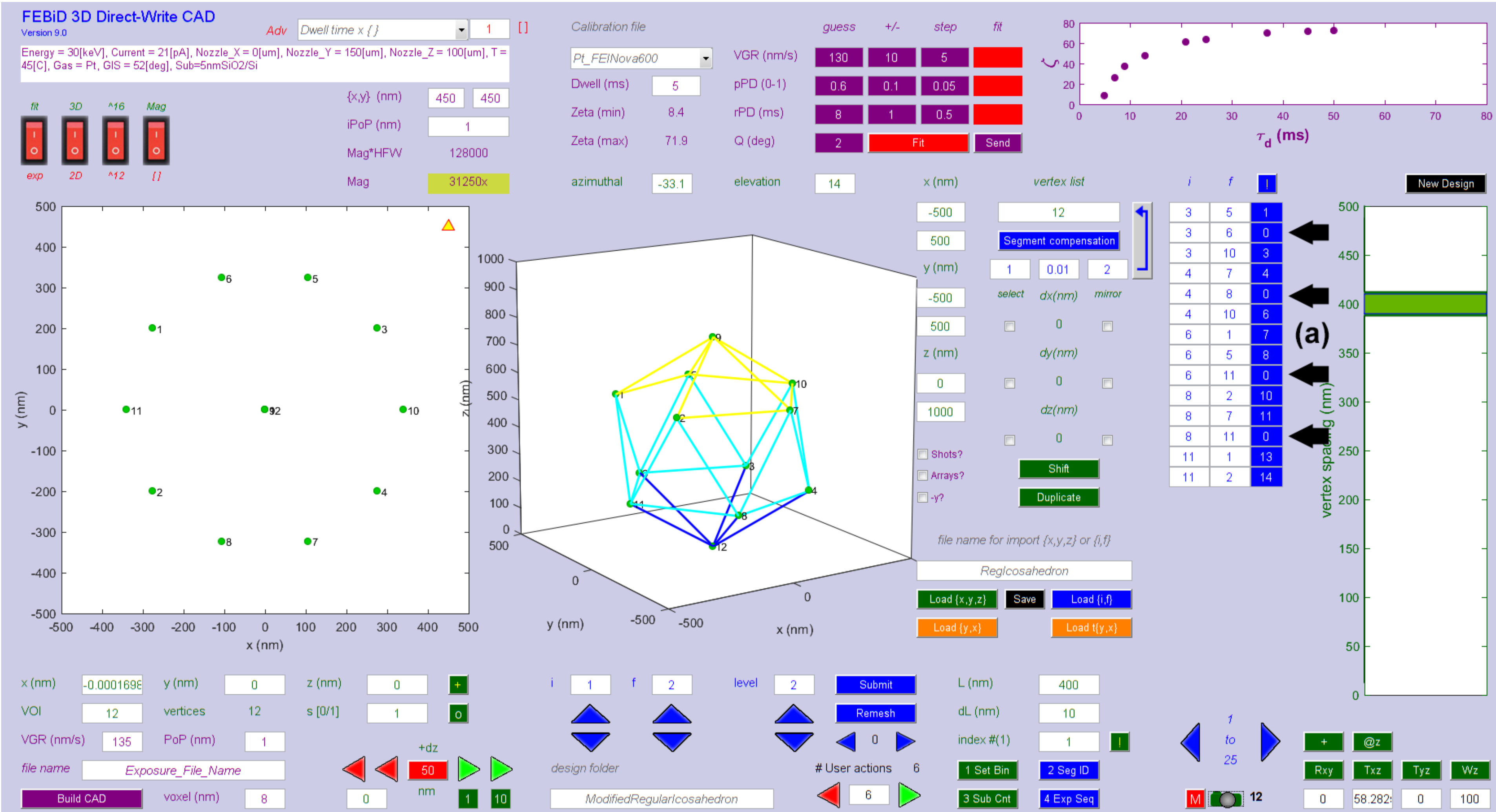



Figure 11

**Action #16: Press the "!' push button to erase the segments from the CAD (figure 12a).**

The segment indices will be reassigned due to the removal of the submitted segments. The 3D plot reflects the removal of the submitted segments from exposure level #2 (figure 12b).

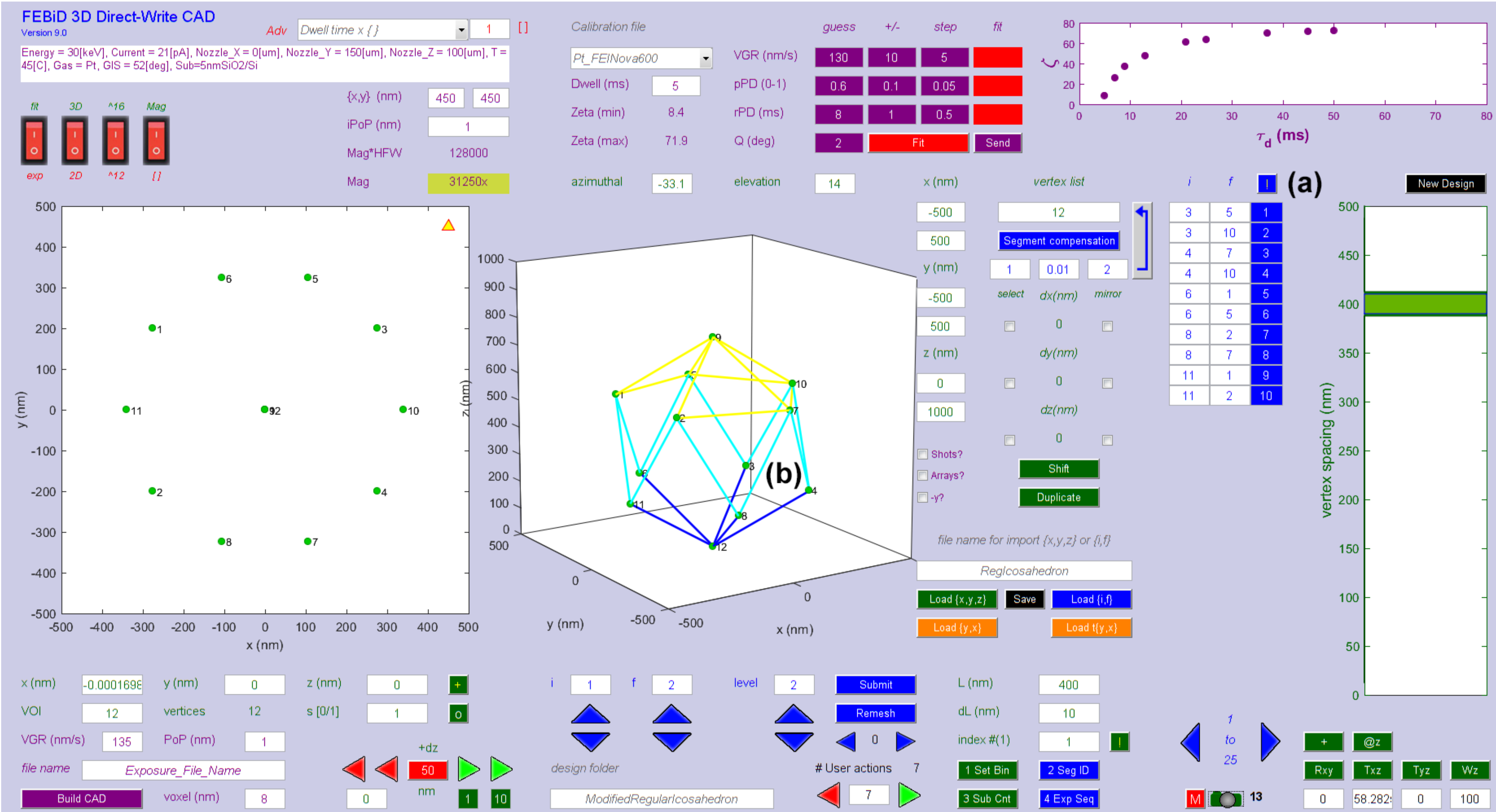



Figure 12

**Action #17: Select exposure level 3 in the “level” text input box (figure 13a).**

The segment list updates in the GUI displaying the current segment list for exposure level #3 (figure 13b).

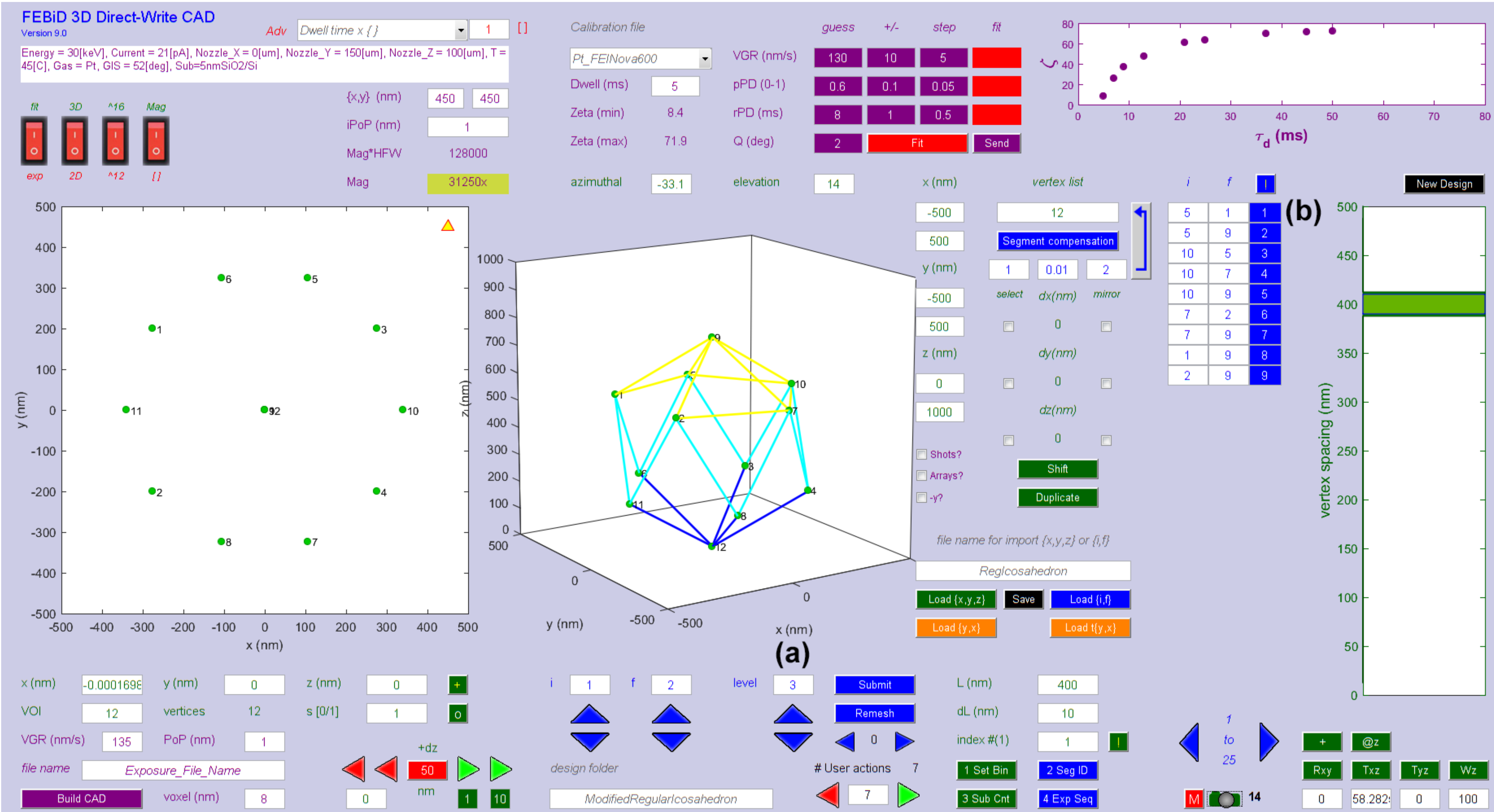


Figure 13

**Action #18: Enter “0” for all segments to be removed from the CAD, in place of their current segment index (figure 14a, *arrows*).**

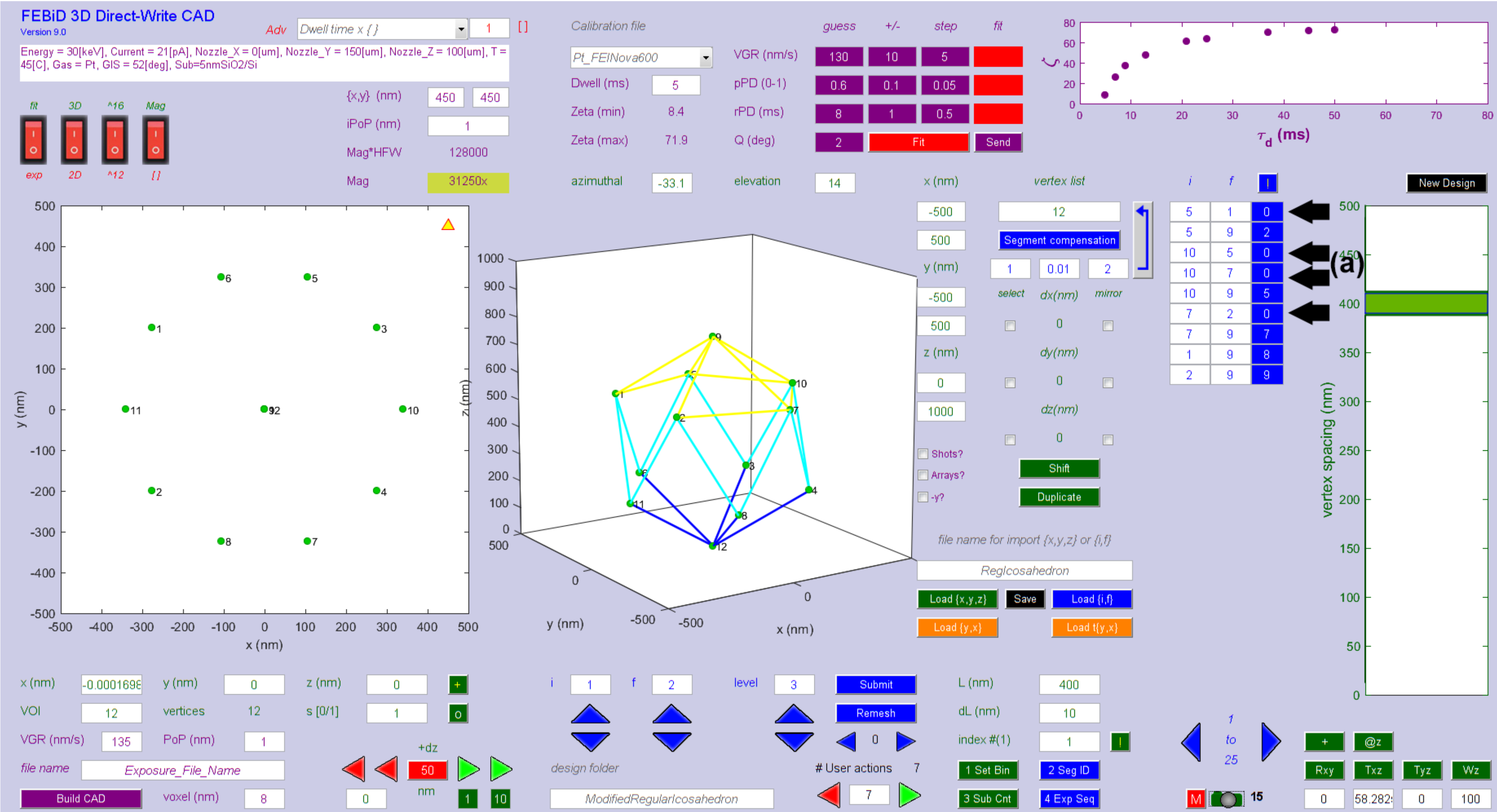



Figure 14

**Action #19: Press the "!' push button to erase the segments from the CAD (figure 15a).**

The segment indices will be reassigned due to the removal of the submitted segments. The 3D plot reflects the removal of the submitted segments from exposure level #3 (figure 15b).

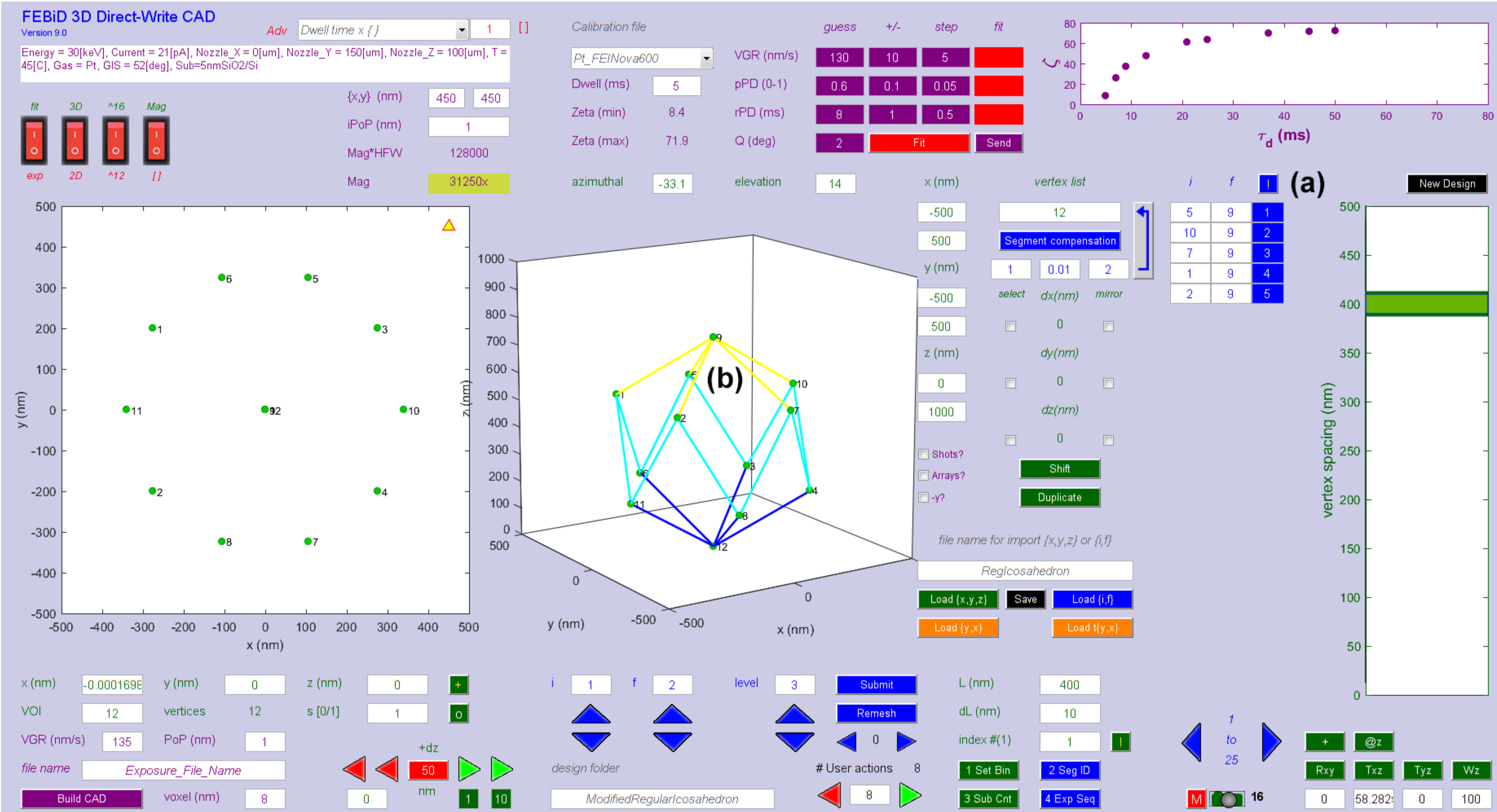



Figure 15

**Action #20: Select "s_i + {ds}" from the "adv" selection menu (figure 16a).**

**Action #21: Enter the value of "1" in the text input box for the "adv" menu (figure 16a).**

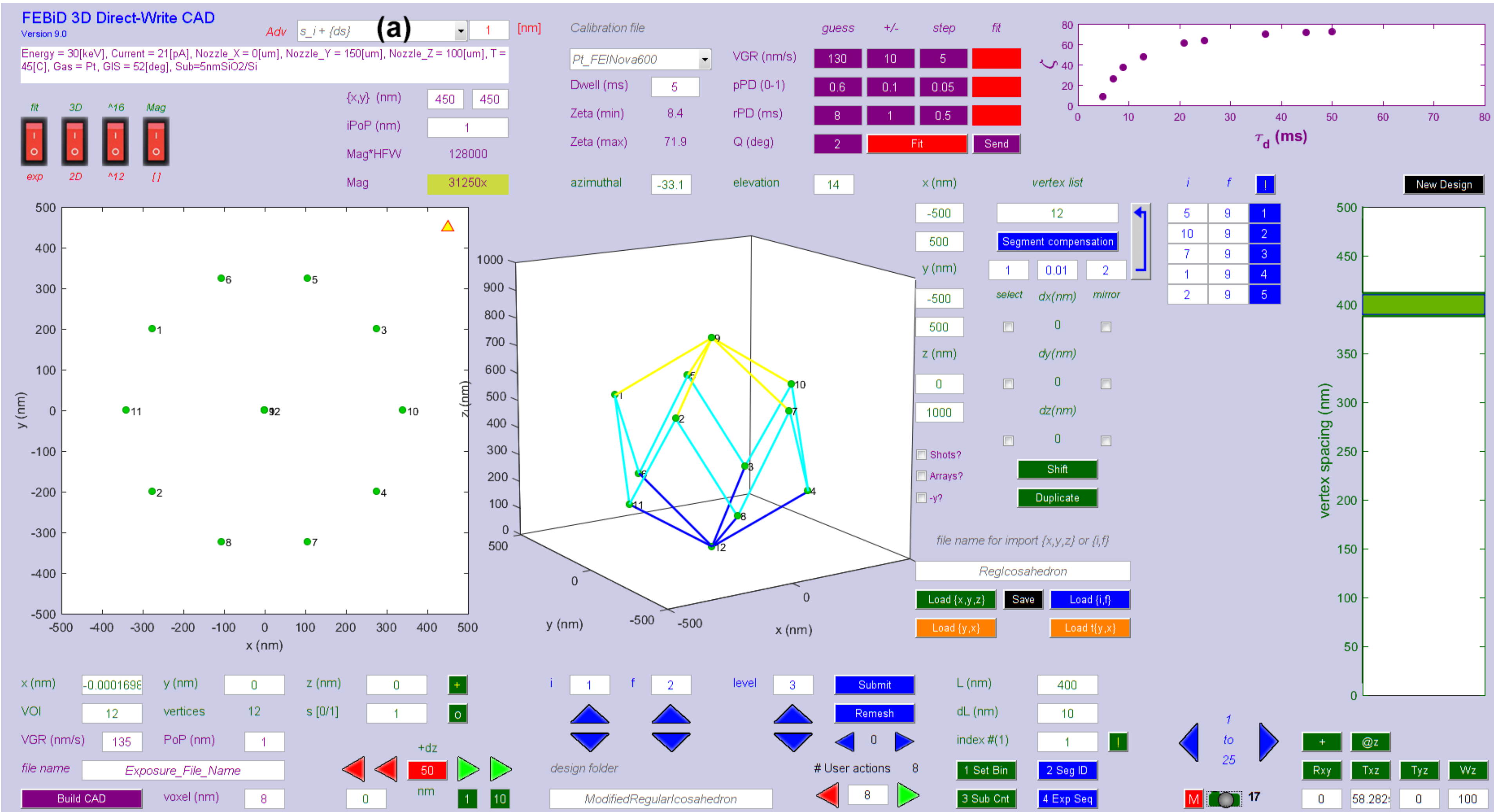



Figure 16

**Proximity correction at segment origin ➔ "s_i + {ds}"**

The parameter "s_i + {ds}" is a simple proximity correction variable. The variable is in units of nanometers. *The variable helps reduce overexposure when multiple segments diverge from a common initial vertex*. *In figure 16, segment divergence occurs at vertices 3, 4, 6, 8 & 11.* This variable will be discussed in the reference frame (x',z), defined in the main text, where (x') is oriented parallel to the scanning direction per segment;

$$x' = (x, y) \ \ (see\ Figure\ 1, main\ text)$$

$$x' = x'_i = 0 \ \ (the\ initial\ position\ of\ the\ segment\ of\ interest)$$

The final segment position is a multiple of the currently selected pixel point pitch (PoP). The pixel point pitch appears as (PoP) on the GUI and as (Λ) in the main text;

$$x'_f = X' \cdot \Lambda \ \ \ (the\ terminus\ of\ the\ segment)$$

The positive x–direction extends in the direction of segment growth. Thus, the beam scanning list along (x') for a given segment, without proximity correction, is;

$$[(\Lambda), (2 \cdot \Lambda), (3 \cdot \Lambda), (4 \cdot \Lambda), \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots, (X' \cdot \Lambda - 3 \cdot \Lambda), (X' \cdot \Lambda - 2 \cdot \Lambda), (X' \cdot \Lambda - \Lambda), (X' \cdot \Lambda)]$$

**value = 0**

By default in the program, the pixel ($x'_i$) is not exposed because this pixel may have been exposed on the previous level as ($x'_f$). This is the case when "s_i + {ds}" is set to "0".

**value > 0**

First, the value is converted into an integer multiple of the current pixel point pitch; PoP*round(*value*./PoP)). In the current example the value was set to "1" [nm]. This value removes the first exposure pixel from the exposure list;

$$[\cancel{(\Lambda),} (2 \cdot \Lambda), (3 \cdot \Lambda), (4 \cdot \Lambda), \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots, (X' \cdot \Lambda - 3 \cdot \Lambda), (X' \cdot \Lambda - 2 \cdot \Lambda), (X' \cdot \Lambda - \Lambda), (X' \cdot \Lambda)]$$

for segments with indices 2, 5, 9 and 12 (see figure 11a, arrows).

The User should expect to have to increase this value as the number of diverging segments increases. A series of calibration experiments is recommended.

**Action #22: Select “s_f - {ds}” from the “adv” selection menu (figure 17a).**

**Action #23: Enter the value of “2” in the text input box for the “adv” menu (figure 17a).**

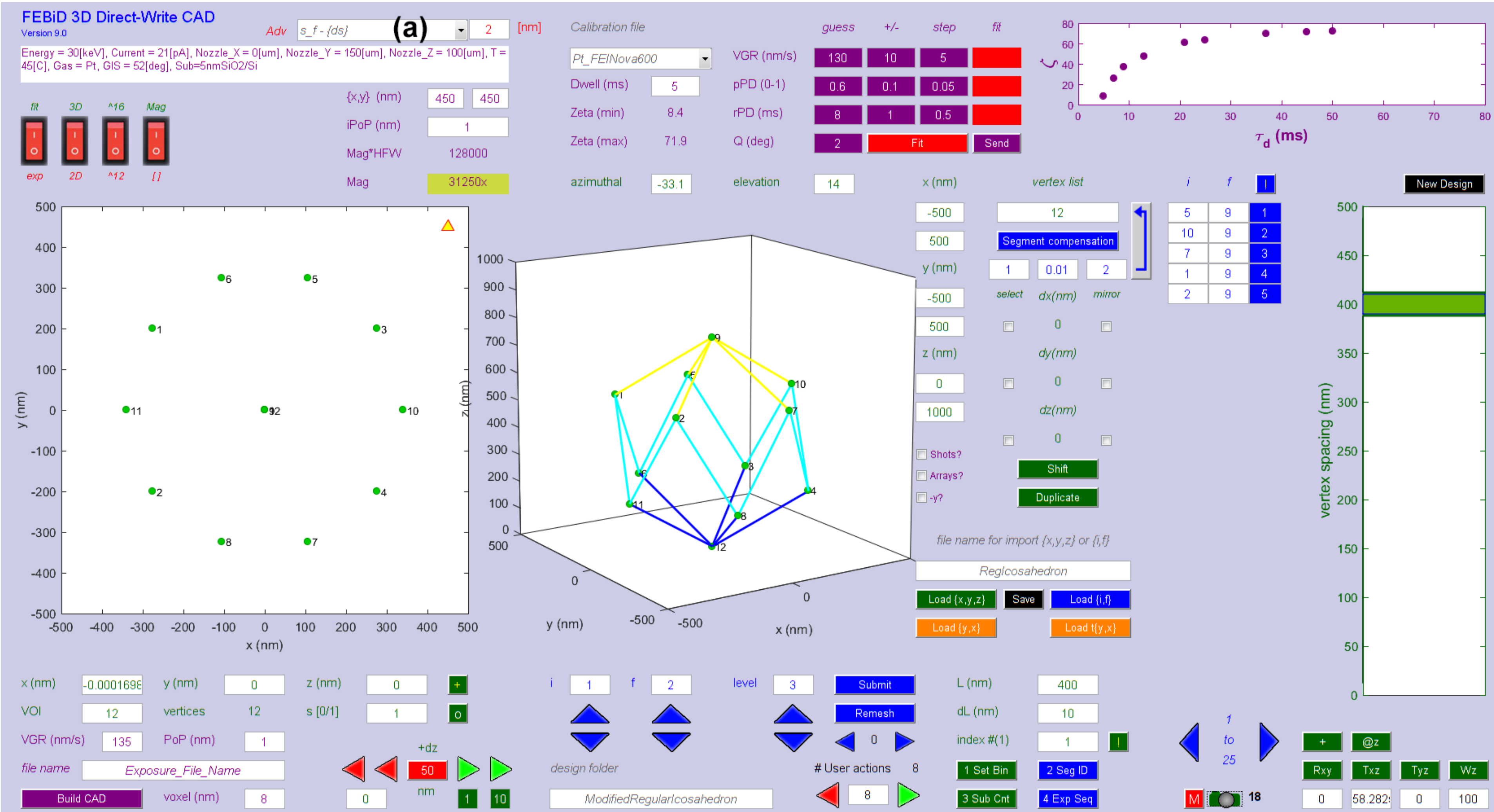



Figure 17

**Proximity correction at segment origin ➔ "s_f - {ds}"**

The parameter "s_f - {ds}" is also a proximity correction variable which works backward from the final vertex position. The variable is also in units of nanometers. *The variable helps reduce overexposure when multiple segments converge at a common vertex*. *In figure 17, segments convergence occurs at vertices 1, 2, 5, 7 & 10*. Building on the previous example we;

$$[\cancel{(\Lambda)}, (2\cdot\Lambda), (3\cdot\Lambda), (4\cdot\Lambda), \dots\dots\dots\dots\dots\dots\dots\dots, (X'\cdot\Lambda - 3\cdot\Lambda), (X'\cdot\Lambda - 2\cdot\Lambda), (X'\cdot\Lambda - \Lambda), (X'\cdot\Lambda)]$$

**value = 0**

No proximity correction is applied from the segment terminus.

**value > 0**

First, the value is converted into an integer multiple of the current pixel point pitch; PoP*round(*value*./PoP)). In the current example the value was set to "2" [nm]. This value removes the final, and second to last, exposure pixels from the exposure list;

$$[\cancel{(\Lambda)}, (2\cdot\Lambda), (3\cdot\Lambda), (4\cdot\Lambda), \dots\dots\dots\dots\dots\dots\dots\dots, (X'\cdot\Lambda - 3\cdot\Lambda), (X'\cdot\Lambda - 2\cdot\Lambda), \cancel{(X'\cdot\Lambda - \Lambda)}, \cancel{(X'\cdot\Lambda)}]$$

for segments with indices 1, 3, 4, & 6 (see figure 14a, arrows).

The User should expect to have to increase this value as the number of converging segments increases. A series of calibration experiments is recommended.