

# RADDOSE-3D Command Reference

Beta version 1.3 – 19 February 2015

Please cite the following publication when using RADDOSE-3D:

## **RADDOSE-3D: time- and space-resolved modelling of dose in macromolecular crystallography**

Zeldin, O. B.; Gerstel, M. & Garman, E. F. (2013). *J. Appl. Cryst.* **46**

doi:10.1107/S0021889813011461

## Contents

|          |                                      |           |
|----------|--------------------------------------|-----------|
| <b>1</b> | <b>General syntax considerations</b> | <b>3</b>  |
| <b>2</b> | <b>Crystal block</b>                 | <b>3</b>  |
| 2.1      | TYPE . . . . .                       | 3         |
| 2.2      | WIREFRAMETYPE . . . . .              | 4         |
| 2.3      | MODELFILE . . . . .                  | 4         |
| 2.4      | DIMENSION . . . . .                  | 4         |
| 2.5      | PIXELSPERMICRON . . . . .            | 5         |
| 2.6      | ANGLEP . . . . .                     | 5         |
| 2.7      | ANGLEL . . . . .                     | 5         |
| 2.8      | CONTAINERMATERIALTYPE . . . . .      | 6         |
| 2.9      | MATERIALMIXTURE . . . . .            | 6         |
| 2.10     | MATERIALELEMENTS . . . . .           | 6         |
| 2.11     | CONTAINERTHICKNESS . . . . .         | 6         |
| 2.12     | CONTAINERDENSITY . . . . .           | 7         |
| 2.13     | ABSCOEFCALC . . . . .                | 8         |
| 2.14     | PDB . . . . .                        | 8         |
| 2.15     | UNITCELL . . . . .                   | 9         |
| 2.16     | NUMMONOMERS . . . . .                | 9         |
| 2.17     | NUMRESIDUES . . . . .                | 9         |
| 2.18     | NUMRNA . . . . .                     | 9         |
| 2.19     | NUMDNA . . . . .                     | 10        |
| 2.20     | PROTEINHEAVYATOMS . . . . .          | 10        |
| 2.21     | SOLVENTHEAVYCONC . . . . .           | 10        |
| 2.22     | SOLVENTFRACTION . . . . .            | 10        |
| 2.23     | PROTEINCONC . . . . .                | 11        |
| <b>3</b> | <b>Beam block</b>                    | <b>11</b> |
| 3.1      | TYPE . . . . .                       | 11        |
| 3.2      | FLUX . . . . .                       | 11        |
| 3.3      | FWHM . . . . .                       | 11        |
| 3.4      | ENERGY . . . . .                     | 11        |
| 3.5      | FILE . . . . .                       | 11        |
| 3.6      | PIXELSIZE . . . . .                  | 12        |
| 3.7      | COLLIMATION . . . . .                | 12        |

|          |                              |           |
|----------|------------------------------|-----------|
| <b>4</b> | <b>Wedge block</b>           | <b>12</b> |
| 4.1      | EXPOSURETIME . . . . .       | 12        |
| 4.2      | ANGULARRESOLUTION . . . . .  | 12        |
| 4.3      | STARTOFFSET . . . . .        | 12        |
| 4.4      | TRANSLATEPERDEGREE . . . . . | 13        |
| 4.5      | ROTAXBEAMOFFSET . . . . .    | 13        |

RADDOSE-3D can take input from one or more files and/or from standard input (STDIN). Any input will be processed by the `InputParser` class and the RADDOSE-3D ANTLR parser. This section describes the syntax of accepted input. Advanced users of RADDOSE-3D can create their own input method that need not rely on the `InputParser` class or the RADDOSE-3D ANTLR parser. This feature will not be covered in this reference.

The simplest use case of RADDOSE-3D will involve only one file describing the entire experiment. In some instances it may be desired to split up the input into a number of files, e.g. one file describing the crystal, one automatically updated file describing the current beam on the beamline, and one file chosen from a set of possible wedge strategies. Each file can contain an arbitrary number (including none) of `Crystal`, `Beam` and `Wedge` block (henceforth called *blocks*). However, splitting up blocks across multiple files is not allowed.

The parser will read the input sequentially, and, when multiple sources are given, one source after the other in the specified order. While the parser may accept `Crystal`, `Beam` and `Wedge` blocks in any order, the exposure of a wedge can only take place if both the crystal and the beam have been set either in an earlier file or before the `Wedge` block within the same file.

## 1 General syntax considerations

Any keywords specified below are case-insensitive. Upper (**CRYSTAL**), lower (**crystal**) and mixed case (**CrYsTaL**) are equivalent.

The characters `#`, `!` and the character sequence `//` denote the start of a comment. Any text from that position until the end of the current line is ignored.

Tabular and newline characters are treated as white space. They can therefore be freely used to format the file for increased readability.

The order of statements within a `Crystal`, `Beam` and `Wedge` block generally is not relevant. There are two exceptions to this rule: The leading keyword (**CRYSTAL**, **BEAM**, **WEDGE**) must be the first keyword of the block. If a keyword is repeated within the same block, then the latter will always override the former.

Every block must be self-contained, e.g. the energy set for the previous `Beam` is not remembered when setting up the following `Beam`, and must be repeated.

Numeric values can be given in scientific notation ( $2.0e2 = 2e+2 = 200$ ), negative values may not have a space between the sign ('-') and the value ( $-1.9e-1 = -.19 = -0.19$ ).

## 2 Crystal block

A `Crystal` block must begin with the keyword **CRYSTAL**. At least the **TYPE** and **DIMENSION** must be specified. Depending on the chosen **TYPE** further declarations may be required.

### 2.1 TYPE

With the keyword **TYPE** the underlying crystal implementation is chosen. Currently four distinct crystal implementations exist:

**TYPE CUBOID** defines a solid crystal with a cuboid shape.

**TYPE SPHERICAL** defines a solid crystal with a spherical shape.

**TYPE CYLINDER** defines a solid crystal with a cylindrical shape.

**TYPE POLYHEDRON** defines an arbitrary crystal shape as a polyhedron. The wire frame type (**WIREFRAMETYPE** keyword - section section 2.2) and file containing the model (**MODELFILE** keyword - section section 2.3) must be specified.

## 2.2 WIREFRAMETYPE

**WIREFRAMETYPE** Specifies the type of wire frame model used to model a crystal of polyhedron type. Currently only .obj (geometry definition) files can be used:

**WIREFRAMETYPE OBJ** specifies the wire frame model to be in .obj format.

## 2.3 MODELFILE

**MODELFILE** Specifies the location of the file that contains the wire frame model of the polyhedron crystal. Currently only .obj (geometry definition) files can be read. The models and the .obj files can be generated using the free and open source 3D animation software **BLENDER**. (NOTE: if you are exporting a *Wavefront (.obj)* file in **BLENDER**, then select the option *Triangulate Faces* before you finalise the export. RADDOS-3D works with triangular faces for the polygons.):

**MODELFILE SOMEMODELFILE.OBJ** specifies the location of the .obj file to be imported.

## 2.4 DIMENSION

**DIMENSION** specifies the size of the crystal. Dimensions are given in micrometres ( $\mu\text{m}$ ). The keyword **DIMENSION** can take either one, two or three parameters:

**DIMENSION D** with a single number (see section 1) as the parameter *D* is used for specifying the crystal dimensions for spherical crystals (**TYPE SPHERICAL**). The parameter sets the crystal diameter. This syntax cannot be used for cuboid or cylindrical crystals.

**DIMENSION D H** with two numbers as parameters *D* and *H* is used for specifying the crystal dimensions for cylindrical crystals (**TYPE CYLINDER**). The parameter *D* sets the diameter of the circular cross section of the cylinder. The parameter *H* sets the height of the cylinder. This syntax cannot be used for cuboid crystals.

**DIMENSION X Y Z** with three numbers as parameters *X*, *Y* and *Z* is used to set the dimensions for cuboid crystals (**TYPE CUBOID**). *X* defines the length of the crystal orthogonal to both the beam and the goniometer at  $L=P=0$ , (see below) *Y* defines the length along the goniometer axis at  $L=P=0$  and *Z* defines the length along the beam axis.

If three parameters are given for a spherical crystal (**TYPE SPHERICAL**) the value for  $X$  sets the diameter of the crystal while the values of  $Y$  and  $Z$  are ignored. If three parameters are given for a cylindrical crystal (**TYPE CYLINDER**) the value for  $X$  sets the diameter of the circular cross section, the value for  $Y$  sets the height of the cylindrical crystal while the value  $Z$  is ignored.

## 2.5 PIXELSPERMICRON

**PIXELSPERMICRON  $F$**  specifies the resolution of the voxel grid used to represent the crystal in voxels/ $\mu\text{m}$ . Defaults to 0.5 voxels/ $\mu\text{m}$ .

Note: When running RADDOS-3D with large dimension values (with SAXS the capillary dimensions typically range in the millimetre range) RADDOS-3D can terminate with the following error:

*Error during invocation of se.raddo.raddose3D.CrystalCylinder: Java heap space.* This is because there is too much memory being used. reducing the pixels per micron will sort this problem and also improve the speed of the program. The cost for the speed improvement is the resolution of the sample voxelization.

## 2.6 ANGLEP

**ANGLEP  $F$**  sets the angle in the plane of the loop between the crystal  $Y$  axis and the goniometer axis. The angle is to be given in degrees, but without the degree symbol ( $^\circ$ ). The default  $P$  ('plane') angle is  $0^\circ$ .

The rotation angle to be applied to the crystal in the plane of the loop (right handed rotation about  $Z$  axis applied to all voxels, as shown in figure 1).

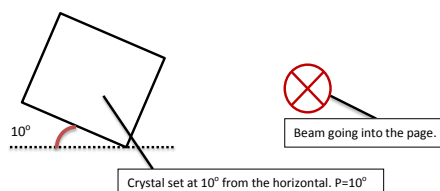


Figure 1: Schematic of **ANGLEP**. Figure courtesy of John Bremridge.

## 2.7 ANGLEL

**ANGLEL  $F$**  sets the loop angle between the plane of the crystal loop and the goniometer axis. The angle is to be given in degrees, but without the degree symbol ( $^\circ$ ). The default  $L$  ('loop') angle is  $0^\circ$ .

The rotation angle to be applied to the angle of the crystal in the loop (right handed rotation about  $X$  axis applied to all voxels, as shown in figure 2).

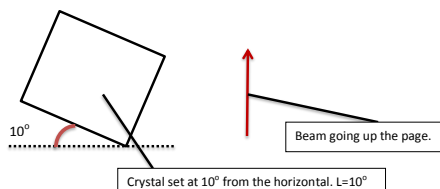


Figure 2: Schematic of **ANGLEL**. Figure courtesy of John Bremridge.

## 2.8 CONTAINERMATERIALTYPE

**CONTAINERMATERIALTYPE MATERIALTYPE** Specifies the material type of the container that encases the irradiated sample. This keyword should be used when the sample is encased within a container e.g. SAXS experiment where the sample is irradiated through a capillary. Currently three distinct container material type implementations exist:

**CONTAINERMATERIALTYPE NONE** The sample is not encased inside any type of container hence there is no attenuation of the beam prior to making contact with the sample. This is the case with a standard X-ray crystallography experiment. This option is selected as the default if the **CONTAINERMATERIALTYPE** is not defined.

**CONTAINERMATERIALTYPE MIXTURE** Defines a container encasing the irradiated sample which is a mixture of elements, determined by the name of the mixture. If this option is used then the user must specify the material mixture (see section **MATERIALMIXTURE**).

**CONTAINERMATERIALTYPE ELEMENTAL** Defines a container encasing the irradiated sample in terms of its component elements. If this option is used then the user must specify the list of the material's component elements (see section **MATERIAELEMENTS**).

## 2.9 MATERIALMIXTURE

**MATERIALMIXTURE MATERIAL** Specifies the material of the container that encases the irradiated sample by its mixture name. The exact input for **MATERIAL** is given in the URL of the corresponding material on the National Institute of Standards and Technology (NIST) Table 4 webpage (click here). For example, if the material through which the sample is irradiated is pyrex glass then click on the 'Glass, Borosilicate "Pyrex"' link and the URL is:

<http://physics.nist.gov/PhysRefData/XrayMassCoef/ComTab/pyrex.html>.

The material name is given immediately before the ".html" and after the last forward slash "/", in this case the word is "pyrex". Hence to set the container material to pyrex glass the input would be **MATERIALMIXTURE pyrex**. Note that the input is case sensitive and an internet connection is required for this option to work.

## 2.10 MATERIAELEMENTS

**MATERIAELEMENTS El I (El I (El I ...))** Specifies the material of the container that encases the irradiated sample by a list of the component elements of the material. The mass attenuation coefficients for each element are downloaded from the National Institute of Standards and Technology (NIST) Table 3 webpage (click here). For example, if the material through which the sample is irradiated is quartz, which has the formula SiO<sub>2</sub> then the input would be **MATERIALMIXTURE Si 1 O 2**. Note that an internet connection is required for this option to work.

## 2.11 CONTAINERTHICKNESS

**CONTAINERTHICKNESS F** Specifies the thickness of the container encasing the irradiated sample. The thickness should be given in  $\mu\text{m}$ .

## 2.12 CONTAINERDENSITY

**CONTAINERDENSITY *F*** Specifies the density of the container encasing the irradiated sample. The density should be given in grams/centimetre.

## 2.13 **ABSCOEFCALC**

This keyword specifies whether the program should use average absorption and attenuation coefficients, or whether it should calculate them from input crystal parameters.

### **ABSCOEFCALC AVERAGE**

### **ABSCOEFCALC DUMMY**

These two commands are equivalent. Each will cause RADDOS-3D to assume an absorption coefficient of  $0.237 \text{ mm}^{-1}$  and an attenuation coefficient of  $0.281 \text{ mm}^{-1}$ . These values are representative of an average crystal at an incident X-ray beam energy of 12.4 keV (1Å). Please see Section 3 in the main paper for more details. Crystal composition keywords will have no effect.

### **ABSCOEFCALC RD**

### **ABSCOEFCALC RDV2**

### **ABSCOEFCALC RDV3**

These three commands are equivalent. RADDOS-3D will call a previous version of RADDOS to estimate absorption and attenuation coefficients.

### **ABSCOEFCALC RD3D**

This command will use the current RADDOS-3D code to calculate the absorption and attenuation coefficients using the crystal composition specified by the user. (NOTE: If the crystal composition is specified then the **RD3D** keyword is preferred over the other **ABSCOEFCALC** keywords, particularly if the crystal contains DNA or RNA molecules).

The composition of the crystal has to be described using the keywords **UNITCELL**, **NUMMONOMERS**, **NUMRESIDUES**, **NUMRNA**, **NUMDNA**, **PROTEINHEAVYATOMS**, **SOLVENTHEAVYCONC** and **SOLVENTFRACTION**. The use of these keywords is described in the sections 2.15–2.23 below.

### **ABSCOEFCALC EXP**

This command should be used if the crystal composition from a Protein Data Bank (PDB) entry is to be used instead of a user specified crystal composition. The current RADDOS-3D code is then used to calculate the absorption and attenuation coefficients using the crystal composition from the PDB entry. The PDB entry must be specified using the (**PDB** keyword - section section 2.14).

### **ABSCOEFCALC SAXS**

This command will use the current RADDOS-3D code to calculate the absorption and attenuation coefficients using the crystal composition specified by the user similarly to the keyword **RD3D**. The major difference is that instead of supplying the number of monomers using the keyword **NUMMONOMERS**, the user supplies the protein concentration used in the SAXS experiment using the keyword **PROTEINCONC** (see section **PROTEINCONC**).

Note that RADDOS-3D treats the irradiated samples as stationary objects. Therefore if the SAXS solution is flowed through the capillary during the exposure then a suitable exposure time for the relevant volume will need to be taken into account.

## 2.14 **PDB**

### **PDB CODE**



Where **CODE** is the four letter code of the PDB entry to be downloaded. From the PDB entry RADDOS-3D will calculate **UNITCELL**, **NUMMONOMERS**, **NUMRESIDUES**, **NUMRNA**, **NUMDNA**, **PROTEINHEAVYATOMS** and **SOLVENTFRACTION**, and hence these keywords are not needed. It will NOT calculate **SOLVENTHEAVYCONC** so if there are any heavy atoms in the solvent then they will have to be specified. Note that an internet connection is required for this option to work.

## 2.15 UNITCELL

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDOS-3D (see section 2.13).

**UNITCELL A B C**

**UNITCELL A B C  $\alpha$   $\beta$   $\gamma$**

Dimensions and angles of the unit cell a, b, c,  $\alpha$ ,  $\beta$ ,  $\gamma$

The first three numbers specify the unit cell size in Angstroms. The second three numbers optionally specify the unit cell angles alpha, beta and gamma.

The (optional) angles are to be given in degrees, but without the degree symbol ( $^{\circ}$ ). If no angles are specified RADDOS-3D assumes default angles of  $90^{\circ}$ .

## 2.16 NUMMONOMERS

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDOS-3D (see section 2.13).

**NUMMONOMERS I** specifies the number of monomers in the unit cell. Only integer numbers *I* should be used. This number should not be confused with the number of monomers in the asymmetric unit.

## 2.17 NUMRESIDUES

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDOS-3D (see section 2.13).

**NUMRESIDUES I** specifies the number of amino acid residues per monomer. Only integer numbers *I* should be used. Using this keyword the number and types of atoms are calculated according to the formula

$$\text{amino acid} = 5C + 1.35N + 1.5O + 8H$$

Sulfur atoms, e.g. from CYS and MET residues, should be added explicitly with the **PROTEINHEAVYATOMS** keyword.

The default value for *I* is 0.

## 2.18 NUMRNA

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDOS-3D (see section 2.13).

**NUMRNA I** specifies the number of RNA nucleotides per monomer. Only integer numbers *I* should be used. Using this keyword the number and types of atoms are calculated assuming an average nucleotide content defined as

$$\text{mean nucleotide} = 11.25H + 9.5C + 3.75N + 7O + 1P$$

If a more accurate estimate is required, individual atoms may be entered explicitly with the **PROTEINHEAVYATOMS** keyword.

The default value for  $I$  is 0.

## 2.19 NUMDNA

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDPOSE (see section 2.13).

### NUMDNA $I$

specifies the number of DNA deoxynucleotides per monomer. Only integer numbers  $I$  should be used. Using this keyword the number and types of atoms are calculated assuming an average deoxynucleotide content defined as

$$\text{mean nucleotide} = 11.75H + 9.75C + 4N + 6O + 1P$$

If a more accurate estimate is required, individual atoms may be entered explicitly with the **PROTEINHEAVYATOMS** keyword.

The default value for  $I$  is 0.

## 2.20 PROTEINHEAVYATOMS

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDPOSE (see section 2.13).

**PROTEINHEAVYATOMS  $El\ I\ (El\ I\ (El\ I\ \dots))$**  defines a list of atoms to add to the protein part of the absorption. Each species is defined by a two character string  $El$  for the elemental symbol, and an integer number  $I$  of atoms of that species per monomer.

The command **PROTEINHEAVYATOMS S 10 Se 2** would add 10 sulfur and 2 selenium atoms per monomer.

## 2.21 SOLVENTHEAVYCONC

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDPOSE (see section 2.13).

**SOLVENTHEAVYCONC  $El\ I\ (El\ I\ (El\ I\ \dots))$**  defines the concentration of elements (not including water) in the solvent in millimoles per litre. Oxygen and lighter elements should not be specified.

The command **SOLVENTHEAVYCONC Na 1000 Cl 1000** specifies 1M sodium chloride in the solvent.

## 2.22 SOLVENTFRACTION

This keyword only has an effect when the absorption and attenuation coefficients are estimated using a legacy version of RADDPOSE (see section 2.13).

### SOLVENTFRACTION $F$

The fraction of the unit cell that is occupied by solvent. If not given explicitly, this value is estimated from **NUMRESIDUES**, **NUMRNA** and **NUMDNA** using 1.35 g/ml for protein, and 1.35 g/ml for DNA and 1.3 g/ml for RNA.

## 2.23 PROTEINCONC

This keyword only has an effect when the absorption and attenuation coefficients are estimated using the keyword **SAXS** section 2.13

### PROTEINCONC *F*

The protein concentration used in the SAXS experiment in grams per litre.

## 3 Beam block

A Beam block must begin with the keyword **BEAM**. At least the **TYPE** must be specified. Depending on the chosen **TYPE**, further declarations may be required.

### 3.1 TYPE

With the keyword **TYPE**, the underlying beam implementation is chosen. Currently three distinct beam implementations exist:

**TYPE TOPHAT** defines a beam with uniform flux.

**TYPE GAUSSIAN** defines a beam with a 2-dimensional Gaussian flux profile. The full-width half-maximum must be specified with the **FWHM** keyword (see section 3.3).

**TYPE EXPERIMENTALPGM** defines an arbitrary beam profile as defined in a given portable graymap file (.pgm) specified with the **FILE** keyword (see section 3.5). The pixel size of the image file also has to be specified with the **PIXELSIZE** keyword (see section 3.6).

### 3.2 FLUX

**FLUX *F*** specifies the total beam flux in photons per second. The flux parameter *F* can be specified in scientific notation (e.g. **1.3e12**).

### 3.3 FWHM

#### FWHM *X Y*

The FWHM of the beam (vertical), (horizontal). Not needed if a Top-Hat beam is used. This defines the *X* and *Y* FWHM of the beam respectively in the RADDPOSE coordinate system.

### 3.4 ENERGY

#### ENERGY *F*

#### ENERGY *F KEV*

specifies the incident photon energy in keV. The optional keyword **KEV** can be appended for human readability of the input file.

### 3.5 FILE

#### FILE *SOMEPGMFILE.PGM*

specifies the location of the .pgm file that describes the beam profile. NOTE: the .pgm file should be in ASCII format. The pixel size of the image file also

has to be specified with the **PIXELSIZE** keyword (see section 3.6).

### 3.6 PIXELSIZE

**PIXELSIZE X Y.**

**X** and **Y** specify the horizontal and vertical size of the pixels in micrometers in the .pgm file respectively.

### 3.7 COLLIMATION

**COLLIMATION RECTANGULAR X Y**

specifies the horizontal and vertical collimation of the beam. Delimits where the beam has non-zero intensity. This is defined by the slits. For an uncollimated Gaussian beam, set to  $\approx 3 \times \text{FWHM}$ .

## 4 Wedge block

A Wedge block must begin with the keyword **WEDGE**.

**WEDGE A B**

**A** and **B** define the start and end angle of the rotation in degrees ( $^{\circ}$ ). At  $0^{\circ}$  the front face of the crystal (**X-Y** plane) is normal to the beam. Rotation is right handed about the **Y** axis, as shown in figure 3).

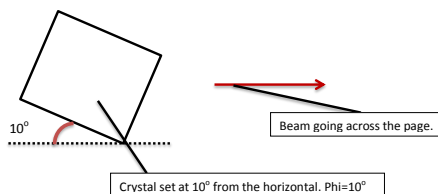


Figure 3: Schematic of angles for **WEDGE**. Figure courtesy of John Bremridge.

### 4.1 EXPOSURETIME

**EXPOSURETIME F**

specifies the total exposure time for this wedge in seconds.

### 4.2 ANGULARRESOLUTION

**ANGULARRESOLUTION F**

specifies the angular step size used for wedge iterations in degrees ( $^{\circ}$ ). Defaults to  $2^{\circ}$ .

Note: If very small wedges are being used e.g.  $< 5^{\circ}$  then the angular resolution should be decreased.

### 4.3 STARTOFFSET

**STARTOFFSET X Y Z**

offset translation in  $\mu\text{m}$  applied to the crystal relative to the origin (defined as the intersection of the beam and the aligned goniometer axis) for the starting position of the wedge. Defaults to 0 0 0.

#### 4.4 TRANSLATEPERDEGREE

##### **TRANSLATEPERDEGREE *X Y Z***

translation of the goniometer during exposure in  $\mu\text{m}/^\circ$  for helical scanning, leading to improvements in dose distribution. Defaults to 0 0 0.

#### 4.5 ROTAXBEAMOFFSET

##### **ROTAXBEAMOFFSET *F***

the offset in  $\mu\text{m}$  along *X* (vertical in most set-ups) between the beam axis and the rotation axis. Used to create 'offset' scanning for improvements in dose distribution. Defaults to 0  $\mu\text{m}$ .