

# Zemax<sup>®</sup> “User-Defined Surface” DLL for simulation of Micromachined Membrane Deformable Mirrors (MMDM) manufactured by Flexible Optical B.V.

User’s Guide

October 31, 2016

## 1 Introduction

Through user-defined surfaces (UDS) Zemax provides powerful mechanism for simulating of optical elements which are not covered by built-in surface types [4]. UDS is a software module (more specifically DLL, dynamic link library) which internally can simulate any phenomena (reflection, refraction, absorption, diffraction etc.) occurring at the surface of interest.

This particular UDS is designed to simulate membrane-based deformable mirrors manufactured by Flexible Optical B.V. We provide this UDS to facilitate design by customers of their systems utilizing our deformable mirrors. Thus it is possible to evaluate parameters of a particular model of deformable mirror and assess its applicability to some particular application and setup. Also it makes easier to choose between several available mirror models. It is also possible to test some custom actuator configurations by generating some new configuration files.

This software is developed by Flexible Optical B.V. in collaboration with Fraunhofer Institute for Laser Technology ILT/ Chair for Technology of Optical Systems TOS RWTH Aachen University. Part of the development was carried out within the framework of UltraSURFACE European project (H2020-ICT-2015, Project no. 687222).

Deflection  $S(x, y)$  of the membrane in response to a voltage distribution  $V(x, y)$  applied to the actuator structure positioned at a distance  $d$  below the membrane is described by Poisson equation ( $\Delta$  denotes Laplace operator) [3]:

$$\Delta S(x, y) = P/T,$$

where  $T$  is a uniform surface tension and electrostatic pressure  $P$  is given by:

$$P = \frac{\varepsilon\varepsilon_0 V(x, y)^2}{d(x, y)^2}.$$

The voltage distribution is defined by geometry of actuators and applied control signals, while boundary conditions are determined by a shape of membrane contour. In general case, the equation can be only solved numerically and this implementation utilizes pre-calculated response functions (separate set for each mirror model) for better performance. The finite-difference version of Poisson equation is solved numerically by relaxation method [2].

## 2 System requirements

Only 64-bit versions of Zemax on Windows platform are currently supported. This software is tested to work with Zemax 13 release 2, but should function properly with any recent version of the software. It exports `UserDefinedSurface4` function and would not work with some [very old] legacy versions of Zemax which expect only `UserDefinedSurface`, `UserDefinedSurface2` or `UserDefinedSurface3`. Please refer to corresponding section of Zemax manual and to their support for details. The requirements for personal computer hardware and operation system are implied by Zemax software itself, but most reasonably modern systems running Windows 7 or newer will work.

## 3 Files and installation

The library is compatible with both “classic” Zemax and “modern” Zemax OpticsStudio. Since those packages maintain their files in separate directory trees, it is necessary to choose correct directory during

Table 1: Files included in the distribution

<code>us_mmdm.dll</code>	Main executable dll file, required
<code>mmdm15-37.bin</code>	Support file for MMDM 15mm, 37 channels ([1], section 7.3)
<code>mmdm30-39.bin</code>	Support file for MMDM 30mm, 39 channels ([1], section 7.5)
<code>mmdm30-59.bin</code>	Support file for MMDM 30mm, 59 channels ([1], section 7.5)
<code>mmdm30-79.bin</code>	Support file for MMDM 30mm, 79 channels ([1], section 7.5)
<code>mmdm40-59.bin</code>	Support file for MMDM 40mm, 59 channels ([1], section 7.6)
<code>mmdm40-79.bin</code>	Support file for MMDM 40mm, 79 channels ([1], section 7.6)
<code>mmdm50-79.bin</code>	Support file for MMDM 50mm, 79 channels ([1], section 7.7)
<code>mdm96.bin</code>	Support file for membrane-based 96-channel deformable mirror with embedded electronics and USB interface (25.4mm aperture)
<code>mdm96_test_sph.zmx</code> , <code>mdm96_test_sph.ses</code>	Example model and session file demonstrating use of a deformable mirror for compensation of spherical aberration caused by a simple plane-convex lens.
<code>mmdm_set.zpl</code>	Zemax macro for setting all (or a group) of the mirror control channels to a specified value
<code>mmdm_set_var.zpl</code>	Zemax macro for setting variable status of control channels (extra data)
<code>mmdm_merit_limits.zpl</code>	Zemax macro for inserting merit function operators limiting range of control signals to desired values
<code>mmdm_load.zpl</code>	Zemax macro for loading control values from a text file
<code>mmdm_save.zpl</code>	Zemax macro for saving of control values to a text file
<code>mmdm_smile37.zpl</code>	Zemax macro for making “smile” test pattern on the surface of MMDM37
<code>us_mmdm_ug.pdf</code>	This User’s Guide

installation. If it is desired to use both programs, it is required to either copy the files manually, or run the installer twice, separately for each package.

The main DLL file is named `us_mmdm.dll` and it should be placed into appropriate directory for user defined surfaces within Zemax program tree (default is `c:\Program Files\Zemax\DLL\Surfaces\`). Zemax does not allow to load UDS files from any other place. The DLL requires separate binary files for each mirror model. They named accordingly to the model, have `.bin` extension (for example `mmdm15-37.bin`, `mmdm50-79`, `mdm96` etc.), and have to be placed into the same directory as the DLL. Several examples are included along with useful macros. They are not strictly necessary for module functioning and in general may be placed in any convenient place. The summary of the files included is given in Table 1.

The software is supplied in the form of single-file executable installer which copies files automatically (Fig 1). It is also possible to manually extract files with an archiving utility (such as 7-zip) and copy them into desired place. This method might be useful for some non-standard or network installations. Please note that copying files into subfolders of `Program files` directory normally requires administrator access rights, so (depending on the system settings) it might be required to invoke the installer “as administrator”.

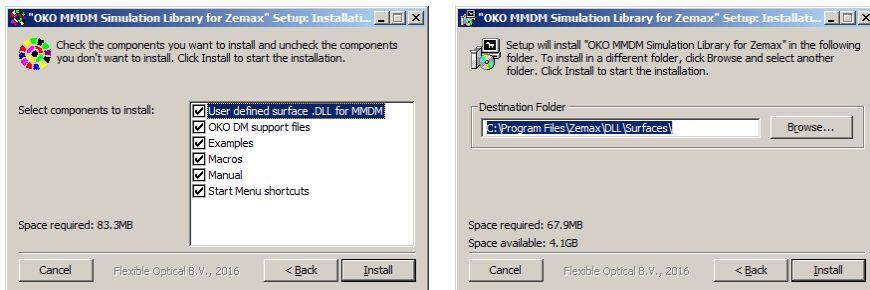


Figure 1: Installation dialog

## 4 Using the UDS with your model

After adding a new surface into your model, please perform the following steps.

1. In the lens parameter table double-click with the left mouse button on the surface type field to invoke surface type dialog. In the field “Surface type” select “User-defined” and in the field “File name” choose “us\_mmdm.dll”. Click “OK” to confirm (Fig 2).
2. In the field “Comment” of the corresponding surface enter name of the deformable mirror model (it should correspond to the filename without extension) (Fig 3).
3. Set Semi-diameter of the surface to correct value (depends on the model of the mirror).
4. Now you can use “Extra data editor” to set control values and variable status (if required) for all control channels (Fig 4). Those parameters represent normalized values and can take values from -1 to 1. If the value outside this range is entered, it is clamped internally, but that change is not reflected in Zemax editor, since there is no technical possibility to pass parameter in from DLL to the main application. Please note that Zemax OpticsStudio has slightly different (supposedly more “user-friendly”) interface compared to original Zemax. It lacks “Extra data editor” and uses lens editor to edit extra parameters. The user needs to scroll right in the window to get access to those columns.

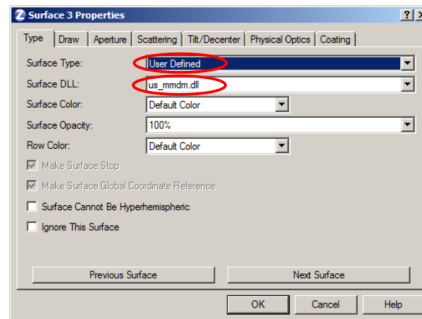


Figure 2: Settings in “Surface type” dialog

Surf:Type	Comment	Radius	Thickness	Glass	Semi-Diameter
OBJ	Standard	Infinity	Infinity		0.000
1	Standard	Infinity	100.000		12.000
2	Coordinat..		0.000	-	0.000
* OKO MDM	mmdm15-37	Infinity	0.000	MIRROR	7.500
4	Coordinat..		0.000	-	0.000
5	Standard	Infinity	-75.000		12.000
6	Standard	Infinity	-7.000	BK7	12.013
7	Standard	100.000	-192.631	V	12.014

Figure 3: Selecting of DM model

Surf:Type	Act 10	Act 11	Act 12	Act 13	Act 14	Act 15
OBJ	Standard					
1	Standard					
2	Coordinat..					
* OKO MDM	0.033	0.040	0.040	0.038	0.038	0.036
4	Coordinat..					
5	Standard					
6	Standard					
7	Standard					
TMA	Standard					

Figure 4: Setting DM control values in “Extra data editor”

## 5 Usage example

The example Zemax file is supplied along with the package demonstrating how to use `us_mmdm.dll` user defined surface in optical design. In this example, parallel beam is reflected by the mirror and focused with a thick plane-convex lens. One can see appreciable amount of the spherical aberration (Fig. 5). After optimization (all deformable mirror control values are taken as variables), focusing quality is drastically improved and has reached the diffraction limit (Fig. 6).

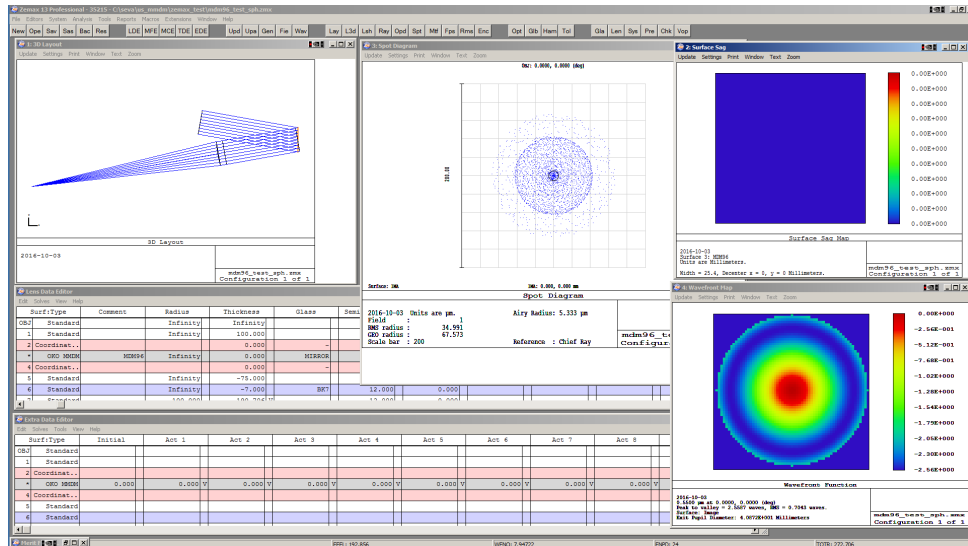


Figure 5: Example `mmdm96_test_sph`. Initial state, appreciable amount of spherical aberration is clearly visible.

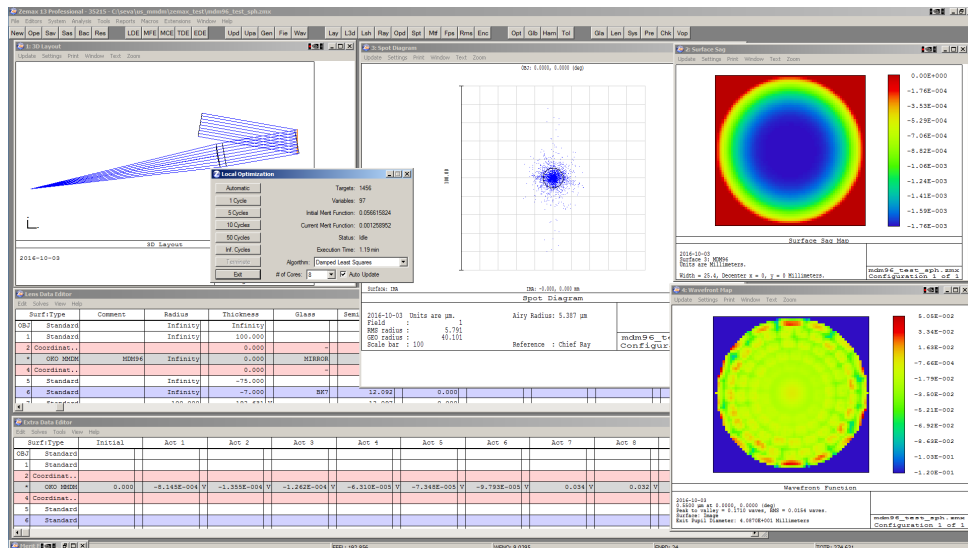


Figure 6: Example `mmdm96_test_sph`. Deformable mirror control channels were set as variables. After optimization the spot diagram and wavefront map indicate that almost diffraction quality of focusing is achieved.

## 6 Data formats

In order to add support for some custom membrane mirror, it is necessary to create binary support file containing response functions for individual actuators. File format is given in Table 2 below. All parameters are in Intel little-endian byte order. The data can be obtained either experimentally, by wavefront measurement, or by simulation.

Table 2: Mirror support file structure definition.

Offset (bytes)	Type	Length (bytes)	Parameter name	Description
0		16		Reserved
16	uint32	4	<code>act_n</code>	Number of actuators
20	uint32	4	<code>grid_x</code>	Horizontal grid dimension (pixels)
24	uint32	4	<code>grid_y</code>	Vertical grid dimension (pixels)
28	float	4	<code>aperture_r</code>	Semi-diameter of the mirror aperture (mm)
32	float	<code>grid_x×grid_y×4</code>		Flat configuration
...	...	...	...	...
<code>32+grid_x×grid_y×4×i</code>	float	<code>grid_x×grid_y×4</code>		Response function for the $i$ -th actuator, $i \in [1 \dots \text{act\_n}]$
...	...	...	...	...

Please note that current version of the library expects only files with `grid_x=grid_y`, thus second of those parameters is redundant.

## 7 Differences between Zemax versions

The software can be used with both “classic” *Zemax* and *Zemax OpticStudio*. The following potential issues should be taken into consideration.

- The packages are installed into different directories. Default is `c:\Program Files\Zemax\` for Zemax (64-bit) and `c:\Program Files\Zemax OpticStudio\` for OpticStudio (64-bit). Please choose appropriate directory during installation. If it is desired to use both versions side-by-side you can either copy files manually or install our simulation software twice, for each version separately.
- Zemax OpticStudio does not have “Extra Data Editor” as a separate window. This functionality is provided by “Lens Data Editor”, additional columns for extra data are located far to the right. In the case of `us_mmdm` user-defined surface relevant parameters are marked as *Act 1*, *Act 2* and so on.
- It seems that due to software bug (or feature?) OpticStudio lacks some merit function operators, `XDLT` and `XDGT` are among them. When optimization procedure is used during simulation and actuator control voltages are often defined as variables. Naturally, there are physical limits on those parameters and they are normally implemented with merit function operators (`mmdm_merit_limits.zpl` macro can be used to make those settings easier). This works in Zemax but not in OpticStudio. Please contact Zemax technical support to learn about possible solutions.
- Comment field is used to pass model of the deformable mirror into the DLL. In Zemax this field is initially left empty, while OpticStudio pre-fills the field with some information. It is still possible to enter correct data in OpticStudio.

## References

- [1] Flexible Optical B.V. OKO guide to adaptive optics. <http://www.okotech.com/images/pdfs/catwww4.pdf>, 2013.
- [2] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical recipes in C (2nd ed.): the art of scientific computing*. Cambridge University Press, New York, NY, USA, 1992.
- [3] S. Timoshenko and S. Woinowsky-Krieger. *Theory of plates and shells*. McGraw-Hill, 1953.
- [4] Zemax LLC. Zemax 13 optical design program. User’s manual. (supplied with the software), 2014.