FLEXIBLE OKO OPTICAL

Flexible Optical B.V.

Adaptive Optics • Optical Microsystems • Wavefront Sensors

96-CHANNEL DEFORMABLE MIRROR WITH EMBEDDED ELECTRONICS

technical passport

OKO Technologies is the trade name of Flexible Optical BV

1 Principle of operations

The deformable mirror, shown in Fig 1(left), consists of a aluminum-coated membrane mounted over a print circuit board (PCB). The PCB has an actuator structure (Fig 1, right) and carries control electronics. The shape of the reflective membrane is controlled by voltages applied to the control electrodes with the membrane grounded.

The device can be used for fast dynamic correction of optical aberrations such as defocus, astigmatism, coma, etc with applications in microscopy, ophthalmology, laser systems, free-space communication, education and research, small telescope astronomy.

The device is controlled and powered by single USB2.0 (High-speed) interface.

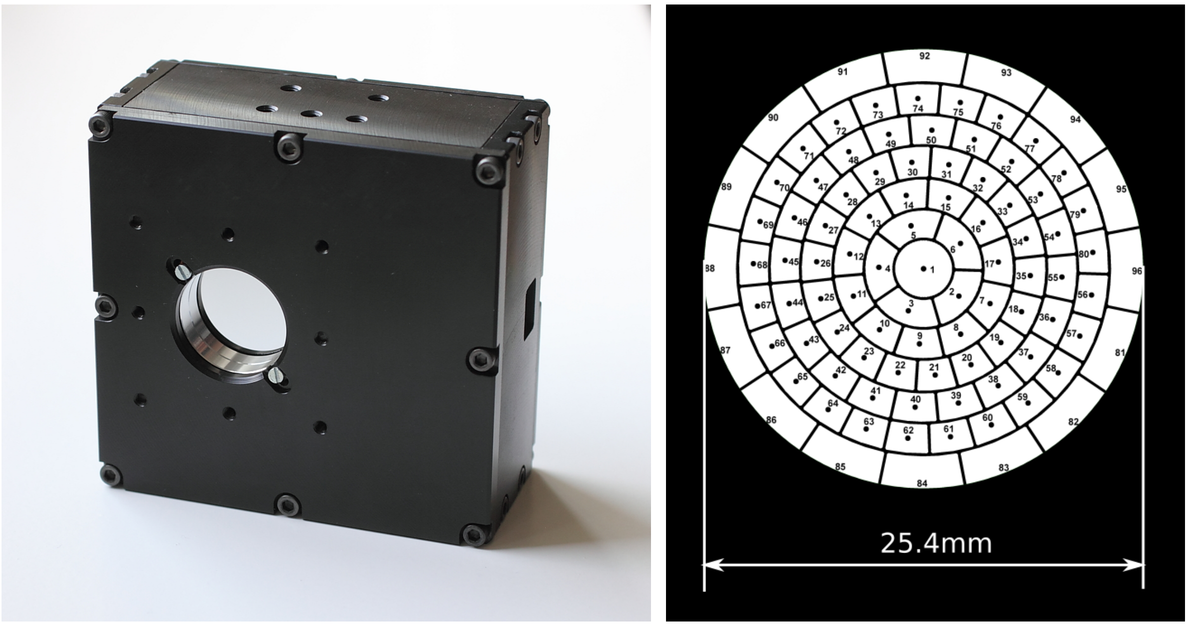

Figure 1: Typical view (left) and actuator structure layout (right) of a 96-channel deformable mirror with integrated electronics

2 Technical data

Table 1 summarizes most important technical parameters of the mirror. A series of interference patterns of the mirror before shipping is shown in Fig 2. Membrane demonstrates quadratics response to input signal, as illustrated by Fig. 4. Temporal switching ability are characterize by Fig 7 (USB interface latency and rise/fall time for electronics) and Fig 8 (optical phase modulation).

It is important to understand that since the device electronically implemented as sample-and- hold device, data should be periodically refreshed. Under normal conditions, refresh period of about 2s gives good result. In typical application such mirrors are used with some sort of dynamic feedback, thus it is naturally refreshed with a frequency of feedback loop.

Table 1: Technical parameters of the mirror.

Parameter	Value
Aperture shape	circular
Aperture diameter	1" (25.4 mm)
Recommended beam diameter	18 mm
Number of actuators	96
Membrane coating	Aluminum (Fig 6)
Maximum deflection of the mirror center	18 μm
Control interface	USB 2.0 High-Speed
Maximum refresh rate	2 kHz
Digital-to-analog (DAC) converter resolution	16 bit
Current consumption (5 V)	200 mA
Initial RMS deviation from plane	less than 0.5 μm
Enclosure size	$90 \times 90 \times 30\,\mathrm{mm}^3$
Membrane recess (from package front surface)	7 mm
Weight	0.45 kg
Serial number	D96V1B02

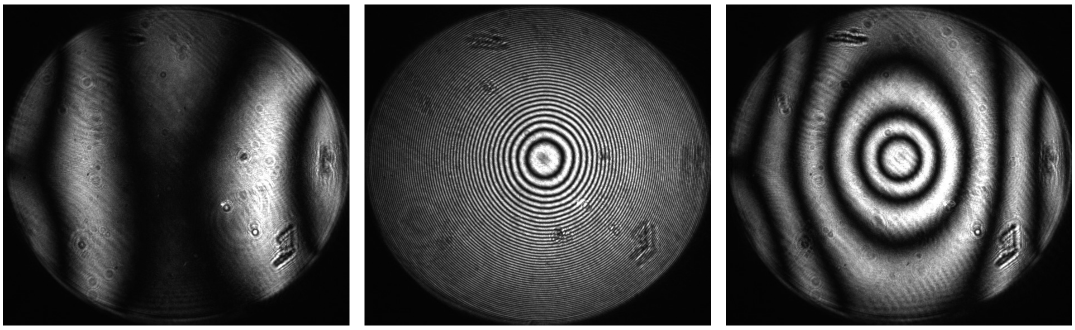

Figure 2: Test of the mirror (left to right): zero voltage applied, maximum voltage applied to all actuators, voltage applied to the central actuator. The patterns are optained with Twyman-Green Interferometer at $\lambda = 633\,\mathrm{nm}$

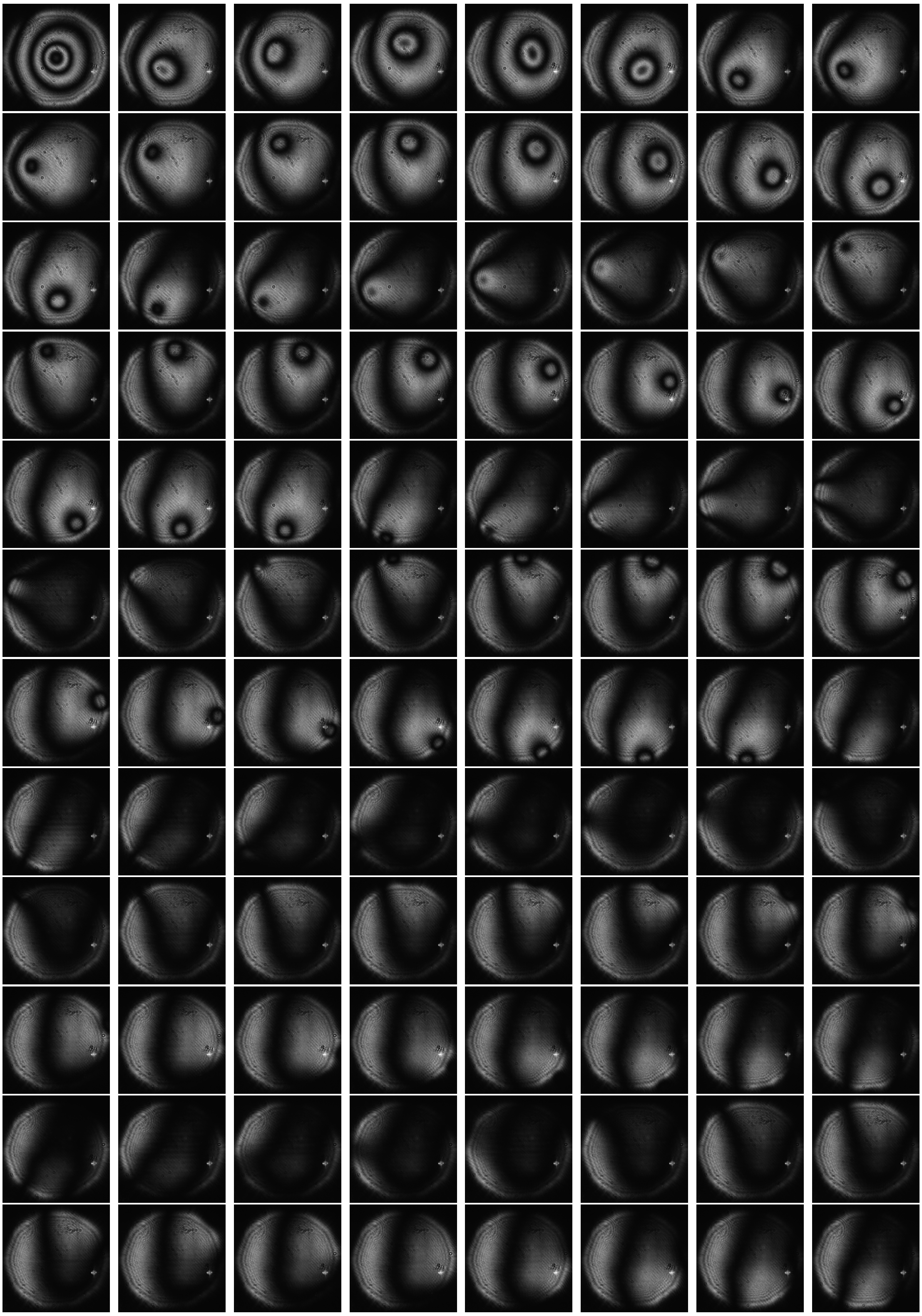

Figure 3: Individual actuator response is demonstrated in rotate test: maximum signal is applied to each actuator sequentially, while others are remained switched off. Aperture of 18 mm is shown

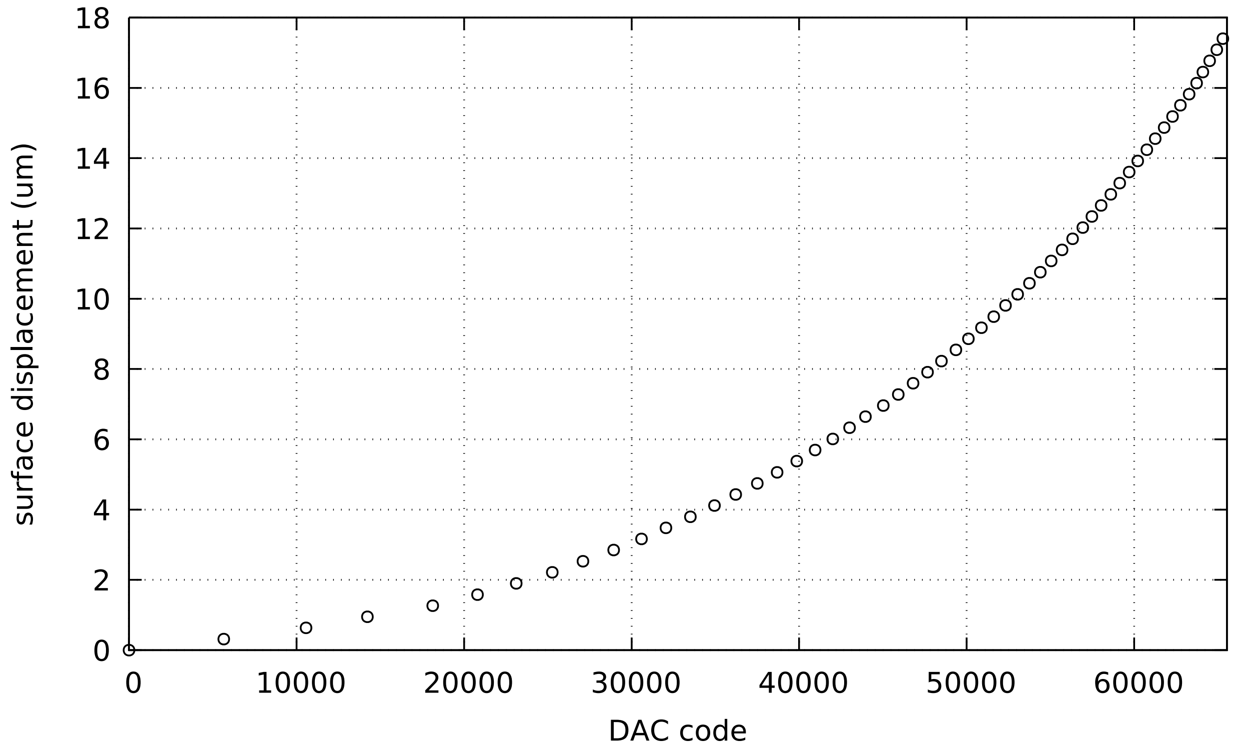

Figure 4: Response of the membrane to input signal. Measured membrane center displacement vs DAC code is shown

3 Mechanical interface

Figure 9 shows unit dimensions relevant to integrating of the device into optical setup. One side face ("up") has a slot for mini-USB connector (type B), opposite face has a grid of 12 mounting holes tapped with M3×0.5 metric thread on grid with period 10×15 mm. Two other side faces are symmetric and carry arrangements of 5 holes each tapped with M4×0.75 thread. Holes on the front surface are not recommended for any mounting. Three of them are used for attachment of internal parts and two others are intended to hold the lid.

4 Software support

The unit is tested under Windows operating systems and compatible with our application software available to that system. At the same time kernel drivers are available for most modern OSes (Linux, Mac OS X) and sample programs supplied in the source form should work under those environments.

4.1 Driver installation

Device is served by FTDI bridge device driver which should be installed before connecting of the unit to a computer. Installation package can be fount on the suppled CD in the directory

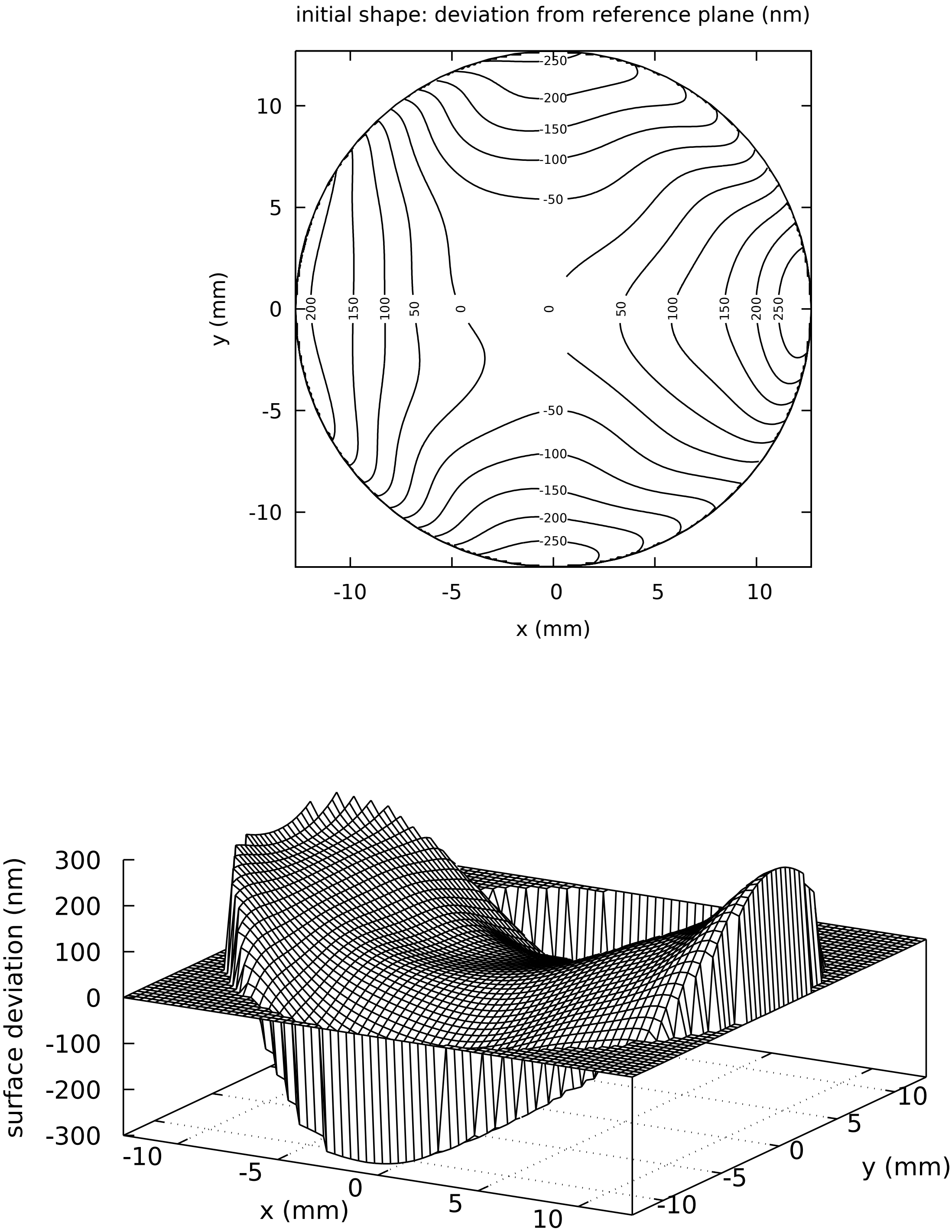

Figure 5: Initial membrane shape reconstructed from interference pattern: isolines (top) and 3d-surface (bottom) representation

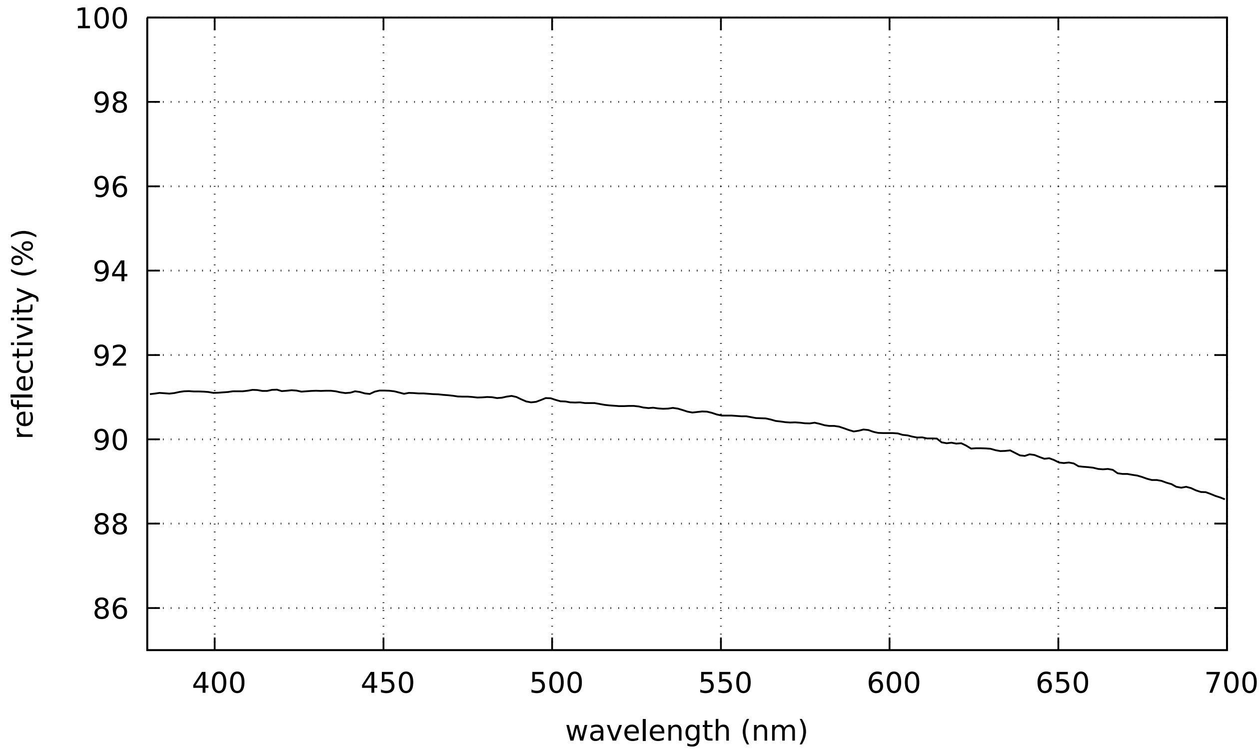

Figure 6: Reflectivity curve for the membrane coated with bare Al

ftdi. The most recent version is always available from the FTDI IC vendors web site http://www.ftdichip.com/Drivers/D2XX.htm .

4.2 Setting to use with BeamTuner application

The device has built-in support for BeamTuner application software. To use the unit with that software please select "Embedded HV DAC" for "DAC Type", "MMDM96ch, embedded control" for "Mirror type" and check "Auto IDs" check-box in main application window as shown in Figure 11. It is possible to have several MDM96 devices connected to the same compute, in this case you might un-check "Auto IDs" check-box and explicitly specify unit ID in "DAC Unit 1" box. BeamTuner software stores its settings in Windows registry between sessions, so it is required to set the parameters only once.

4.3 Setting to use with FrontSurfer software

Support for FrontSurfer wavefront analysis and control system is implemented through "mirror plug-in" mechanism. The required module (in fact a DLL, Dynamic Link Library) can be found in `fs_plugin` directory on CD. It is convinient to copy the module to the directory into which FrontSurfer is installed (default location is `c:\Program Files (x86)\FrontSurfer1.4.10.dm\`). Plug in selection can be reached from the main menu (Mirror→Configuration) (Figure 12)

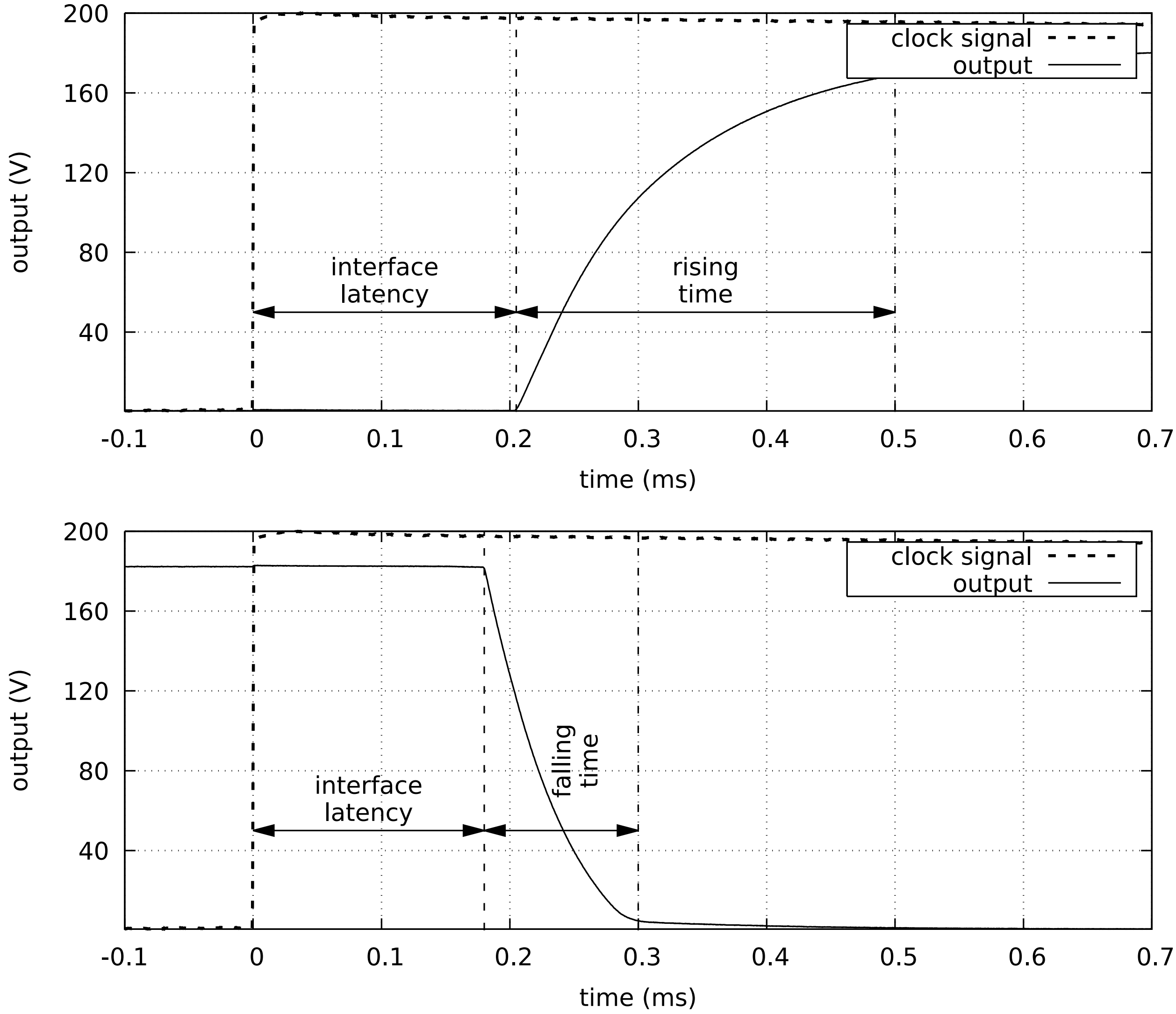

Figure 7: Temporal electric response of the device

4.4 Sample demo programs

A number of simple demo programs are provided on accompanying CD in the form of both source code and binary executables (directory `demo`). They can used for testing of the mirror as well as a basis for development of user applications. The programs are written in C programming language in intentionally straightforward manner. The following programs are included.

am_set applies the same voltage given as argument to all actuators.

rotate sets the voltage given as argument to all channels of OKO mirror, one by one.

set_channel sets given voltage into one given channel, 0s to others.

ramp gradually increase signal on all actuators.

The programs above follow the same logic that Flexible Optical B.V. uses for other models of deformable mirrors/DAC combinations.

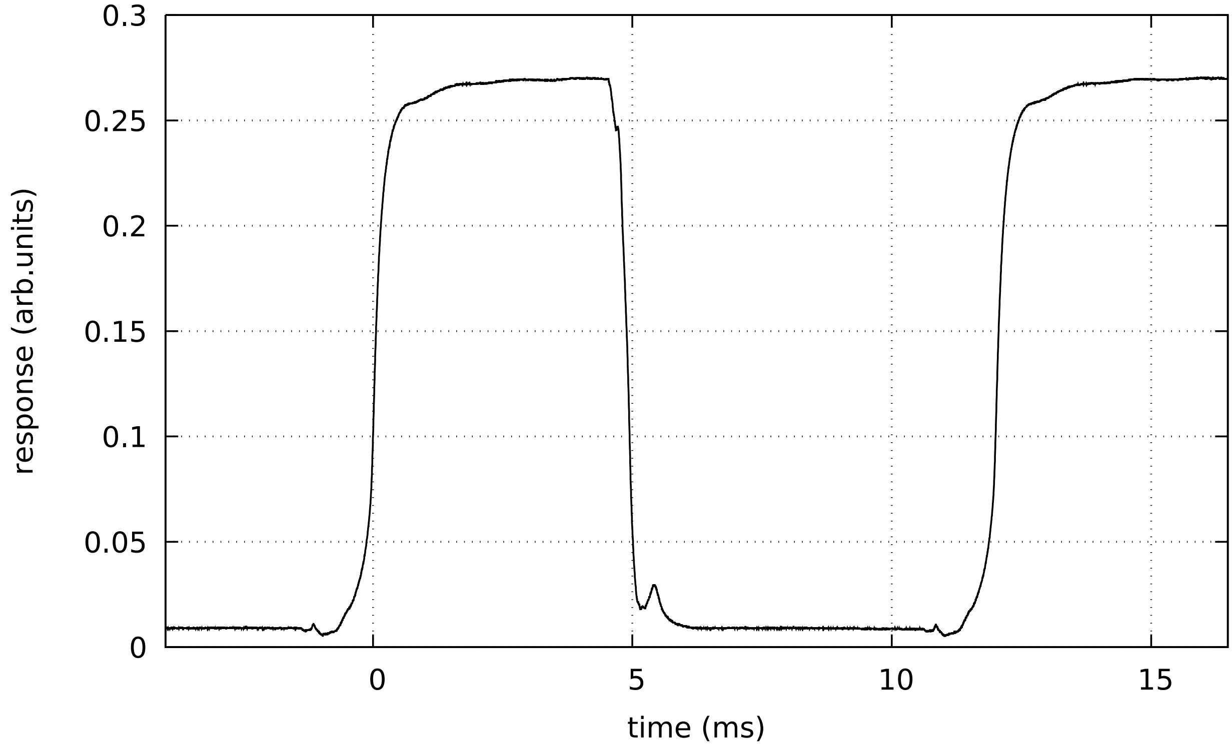

Figure 8: Switching characteristics of the deformable mirror. Square wave signal with maximum amplitude is applied to all actuators leading to switching focus between two extreme states. A pinhole is placed in the focal plane with photo diode detector recording the signal

4.5 Developing your own software

Another way to develop user application is to use our new software development kit. The core of the library is implemented as a hierarchic set of C++ classes and provided in the source form. All models of digital-to-analog converters and all standard mirrors manufactured by Flexible Optical B.V. are supported in unified way. Those classes can be used directly in user developed applications. The code is pretty much straightforward and the minimal sample program is shown in Fig. 13

Dynamic-link libraries (DLLs) compiled from C++ use “decoration” of function names which make them not compatible between different programming languages and compilers. That is why a sort of “wrapper” functions is created in plain C. Those functions are compiled together into simplified DLL which can be easily used in other programming systems. We provide a binding for C, Python and MATLAB. It is relatively easy to import the functions into other languages such as LabView. The example below (Fig. 14) shows how to access a mirror from a program in Python.

Python bindings and examples tested to work with both 2.7 and 3.X language flavours with both 32- and 64-bit implementations. MATLAB examples required compatible C compiler (works with freely available Microsoft Visual Studio Express Edition) and tested with MATLAB 2015b version. Separate DLLs are provided for 32-bit and 64-bit versions. C/C++ code tested to compile with GCC compiler (MinGW package) and Microsoft Visual Studio. Batch files with compilation instruction are included where appropriate. The software is tested under Windows

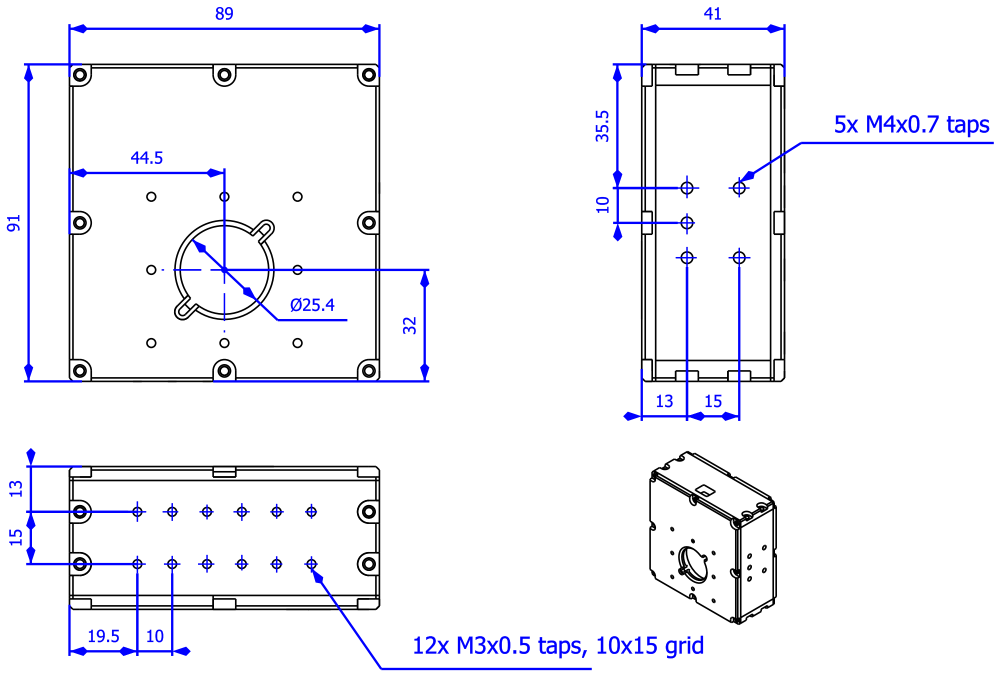

Figure 9: MDM96 enclosure dimensions

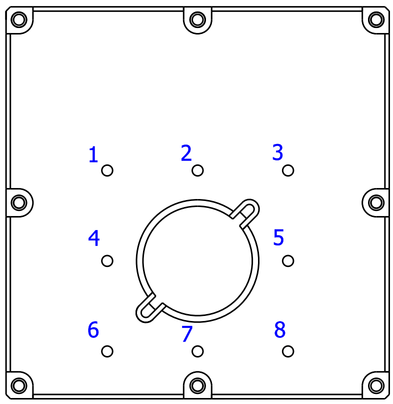

Figure 10: MDM96 front panel sketch showing eight holes with M3x0.5 thread located at 21mm grid. The holes are not intended (and not recommended) to be used for mounting. Three of the holes (## 2,6,8) have some internal parts attached to them (**absolutely not suitable** for mounting!). #4 and #5 are normally used for the lid. Usually unused holes are filled with plastic stubs.

environment but could be deployed under Linux with minimal effort.

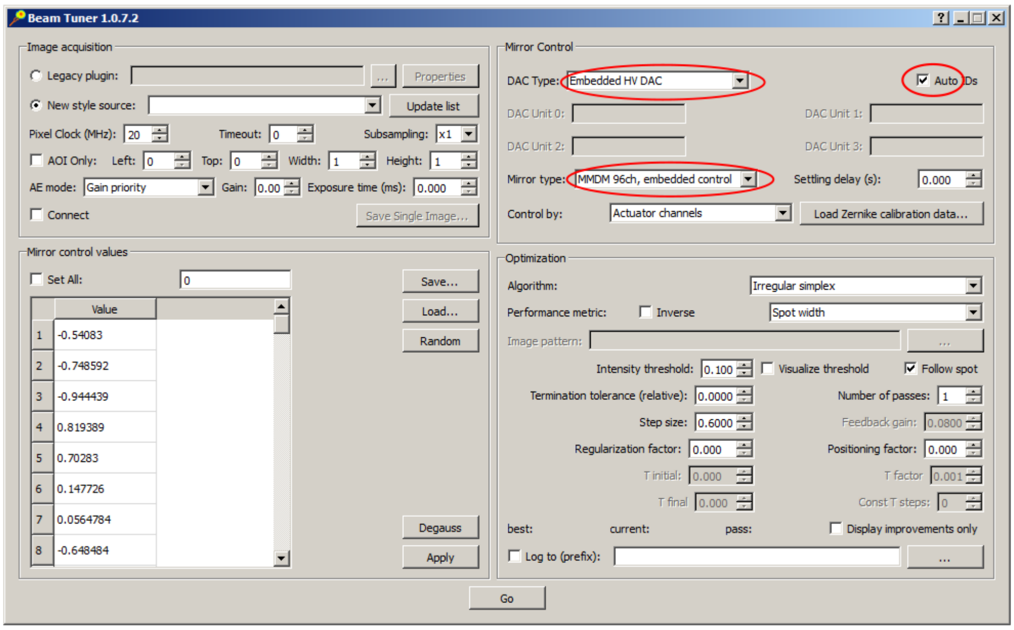

Figure 11: Settings for BeamTuner application

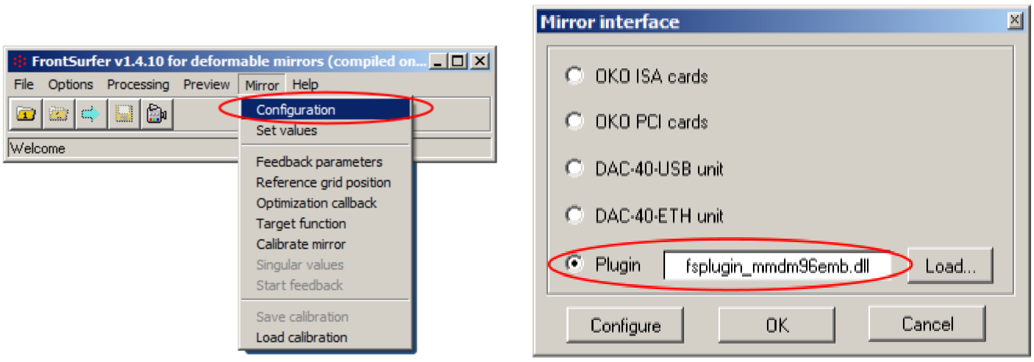

Figure 12: Settings for FrontSurfer software

4.6 Zemax simulation support

We provide a software support for realistic simulation in Zemax of optical systems that include deformable mirrors manufactured by Flexible Optical B.V. The module is implemented as Zemax “user defined surface” and available for free from our web site: `http://www.okotech.com/zemax-dm-uds`. Particular mirror model (“mdm96”) should be specified in the *Comment* column of *Lens Data Editor* for the surface in question and correct data (≤ 12.7) should be entered for *Semi-Diameter*. Normalized control voltages are to be specified in *Extra Data Editor* in columns *Act 1…Act 96*. Please note that For full details please refer to the *User Guide* of the corresponding product which is available as a part of installation package as well as from our web site.

```
#include <cstdio>
#include <stdexcept>
#include "okodac.h"
int main()
{
    try
    {
        DAC *dac=new DAC_DALSA(); // built-in HV DAC with first available ID
        Mirror *mirror=new Mirror_MMDM96_EMB(dac); // attach to the mirror
        int n=mirror->numberOfActuators(); // number of channels
        double *data=(double *)malloc(n*sizeof(double));  // control values
        for(int i=0; i<n; i++) data[i]=0.5; // some fixed value to all channels
        mirror->setChannels(data,n);
        mirror->updateAllChannels(); // only necessary if immediate_update is off
    }
    catch(exception &e)
    {
        printf("Operation failed: %s",e.what());
    }
    return 0;
}
```

Figure 13: Example minmal program in C++ demonstrating control of the mirror

```
import okodm
import time
import sys

h=okodm.open("MMDM 96ch, embedded control","Embedded HV DAC")
if h==0:
    sys.exit("Error opening OKODM device: "+okodm.lasterror())
n=okodm.chan_n(h)
stage=1
try:
    while True:
        if not okodm.set(h, ([1] if stage else [-1])*n):
            sys.exit("Error writing to OKODM device: "+okodm.lasterror())
    time.sleep(1)
    stage^=1
except KeyboardInterrupt:
    pass
okodm.close(h)
```

Figure 14: Example program in Python

5 Warranty

The equipment is covered by a one-year factory-defect warranty.

If the mirror is damaged during shipping, it will be replaced by a similar device within two months. A photo of the damaged device should be sent to Flexible Optical B.V. (OKO Technologies) within 3 days after the damaged device is received.

EXCEPT WHEN OTHERWISE STATED IN WRITING FLEXIBLE OPTICAL B.V. (OKO TECHNOLOGIES) AND/OR OTHER PARTIES PROVIDE THE SYSTEM "AS IS" WITHOUT WARRANTY OF ANY MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE EQUIPMENT IS WITH YOU.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL FLEXIBLE OPTICAL B.V. (OKO TECHNOLOGIES) BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE HARDWARE AND SOFTWARE DESCRIBED IN THIS DOCUMENT.

6 Contact

All questions about the technology, quality and applications of adaptive mirror should be addressed to:

Flexible Optical B.V.
Polakweg 10–11,
2288 GG Rijswijk ZH,
the Netherlands

Date:

Signature: