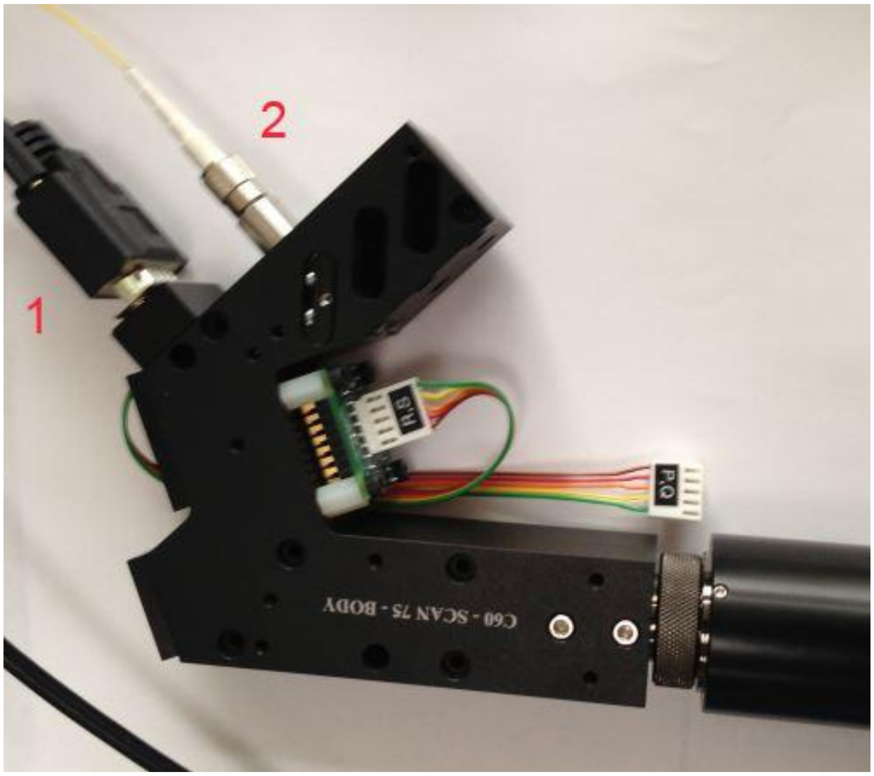
MicroMirror Drive Card User Guide

The MicroMirror Drive card is a module for the TG-1000 system. Using this card user can control X and Y axes of 2 MEMS MicroMirrors. The card can be operated in external input mode (0 to 5V analog signal) or in internal input mode (thru onboard DAC so the mirror can be positioned thru serial commands and built in routines).

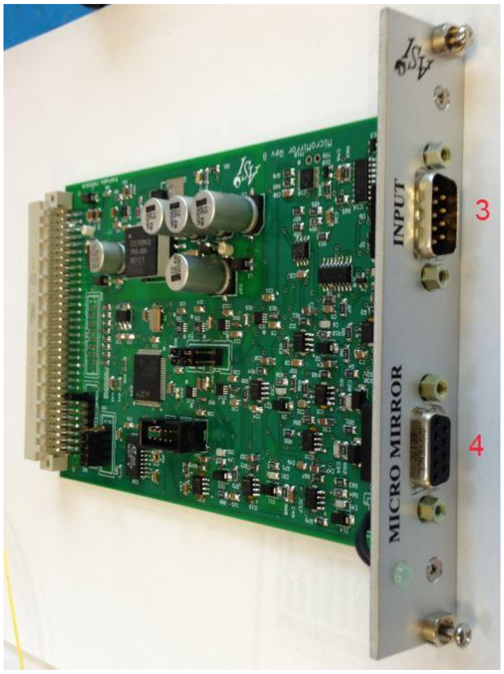
Although MEMS Mirrors are operated in open loop mode, a 5th order tunable Bessel filter is provided which can be used to avoid exciting the mechanical resonance of the MEMS mirrors.

Note: MEMS MicroMirrors are very sensitive to ESD, please observe precautions when handling this equipment. Even more than usual, it is important to never connect or disconnect the cables when system power is on.

Equipment and connectors



MicroMirror scanner body. The DB9 connector (1) connects to the MicroMirror Drive Card thru a DB9 straight cable. 2 is the fiber optic coupler. As shown in the picture, MicroMirror is connected to drive card as R and S axis. Newer scanners have the fiber connector on the end of the scanner body.



MicroMirror Drive Card. Connect 4 to MicroMirror Scanner body thru a DB9 straight cable. 3 is the input to accept analog signals. Analog input range is 0-5Volts.

Straight DB9 cable to connect Driver card and scanner body.

Use the connector to connect coax cables with user's analog input to MicroMirror drive card. Coax connectors are marked 1,2,3,4. MicroMirror 1's X , Y axis and MicroMirror 2's X, Y axis respectively.

Electrical Characteristics of MicroMirror Drive Card

- Output Voltage : 0 to 100V
- Output Current: <1mA (100kOhm source resistance)
- Analog Input range: 0-5V
- Analog input amplification: x20

MEMS MicroMirror Details

- 1200um dia mirror
- Max angle x and y: 4.6 degrees
- Max voltage x and y axis: 131V
- Resonant frequency x and y axis ~2.5kHz
- Quality factor: 50
- Recommended LPF cutoff frequency, 6th order Bessel : 1KHz
- Further details can be found here: http://tinyurl.com/1200ummirror

Shortcomings of the Drive Card

No calibration has been performed to make sure the relationship between the applied input voltage (in external input mode) or between the axis positions (in internal input mode) are accurate. However we can make a rough guess based on the micro-mirror specifications of 9.2degrees for 131V; with analog multiplication of 20x we expect each volt of analog input (in external mode) to move the mirror by approximately 20V/131V*9.2degrees = 1.4degrees. We expect the user to cross-calibrate the move amount with a corresponding stage movement; the **DACK** command can also be used to set a scale factor for calibration.

Until and including version C of the Drive Card, the high-voltage amplifiers get very hot due to their quiescent current. In later versions of the card this issue is addressed by changing amplifier ICs. Occasionally cards (version C and before) fail due to overheating, so it is advised to turn of the TG-1000 controller when not in use. Contact ASI for assistance if the MicroMirror Drive Card appears to fail entirely (all the output pins on the DB9 connector to the Scanner will be 0V in that case).

MicroMirror-specific Serial Commands

Control of the MEMS MicroMirrors is similar to any other axis of the TG-1000 system. This means all the regular axis commands work (e.g. **`MOVE`**, **`MOVREL`**, **`WHERE`**, **`ZERO`**, **`INFO`**, etc.). The axis units are 1/1000 of a degree. So **`MOVE R=2000`** moves the beam by about 2 degrees.

Apart from the regular axis commands here are some MicroMirror-related commands, including some that are "recycled" for a different use than for motor axes.

Command: **MAINTAIN** **(deprecated in version v2.8+, use PM instead)**

Shortcut: **`MA`**

Format: **`MA [axis]=[0 or 1]`** …

Units: Integer code, 0-1

Type: Axis-Specific

Remembered: Using **`[Addr#]SS Z`**

Function: This command is "recycled" for a different use in MicroMirror axes than for motor axes. In the context of a MicroMirror axis this command is used to put the axis in internal or external mode.

0 is external input mode. Mirror is positioned based on analog input voltage. Default mode, except for SPIM-enabled systems.

1 is internal input mode. Mirror can be positioned thru serial command or onboard routines. Default mode for SPIM-enabled systems

Example:

```
MA R=1 S=1          Puts the axes in external input mode
:A
MA R? S?            Queries the mode of axes
:A R=1 S=1
```

Command: **PM** **(version v2.8+)**

Shortcut: **`PM`**

Format: **`PM [axis]=[0 or 1]`** …

Units: Integer code, 0-1

Type: Axis-Specific

Remembered: Using **`[Addr#]SS Z`**

Function: This command is "recycled" for a different use in MicroMirror axes than for piezo. In the context of a MicroMirror axis this command is used to put the axis in internal or external mode.

1 is external input mode. Mirror is positioned based on analog input voltage. Default mode, except for SPIM-enabled systems.

0 is internal input mode. Mirror can be positioned thru serial command or onboard routines. Default mode for SPIM-enabled systems.

Note: Options are indeed reverse of MAINTAIN command, to say consistent with PM's piezo options.

Example:

```
PM R=1 S=1
:A
```

Puts the axes in external input mode

```
PM R? S?
R=1 S=1 :A
```

Queries the mode of axes

Command: **BACKLASH**

Shortcut: `B`

Format: `B [axis]=[0.1 or 650]` …

Units: kHz

Type: Axis-Specific

Remembered: Using `[Addr#]SS Z`

Function: This command is "recycled" for a different use in MicroMirror axes than for motor axes. In the context of a MicroMirror axis this command is used to set the cut off frequency of the 5th order Bessel filter. Units are in KHz, lowest acceptable value is 0.1 (100Hz) and highest is 650 (650kHz). Please refer to "Setting Filter Frequency" section for more details on its usage.

Example:

```
B R=0.1 S=0.1
:A
```

Sets 100Hz filter cut off freq for R and S axes

```
B P? Q?
:P=0.4 B=0.4 A
```

Queries the filter cut off freq for P and Q axes

Command: **DACK**

Shortcut: `D`

Format: `D [axis]=[0 to 1]` …

Units: Unitless float between 0 and 1

Type: Axis-Specific

Remembered: Using `[Addr#]SS Z`

Function: This command is "recycled" for a different use in MicroMirror axes than for motor axes. For MicroMirror axes it sets up a calibration constant or scale factor that is used to attenuate the scanner motion when used in internal input mode (as the implementation is in software it does not apply in external input mode). In tests we found that the two axis of the MEMS MicroMirror does not tilt by the same amount when similar inputs are applied. This may be critical for some applications. This parameter sets an attenuation, so both axes can be made to tilt the same amount. For example, if the S axis was found to be doing 85% travel of R axis then we could attenuate R to compensate by setting "`D R=0.85 S=1`".

Example:

```
D R=0.85 S=1
:A
```

Attenuates the travels of R axis by 15%.

Command: **PZINFO**

Format: `[Addr#]PZINFO`

Units: None

Type: Card-Addressed

Remembered: Not applicable

Function: PZINFO is a diagnostic command, which for MicroMirror gives similar output to what is shown below. ASI reserves the right to change the format of the PZINFO command at any point as more diagnostic features are found to be useful.

Example:

```
1PZINFO
Main :23.6 V
I2C Check> DAC[OK] OSC1[OK] OSC2[OK]
Mode> R[EXT] S[EXT] T[EXT] U[EXT]
```

Command: **PR** **(version v2.83+)**

Shortcut: `PR`

Format: `PR [axis]=[5 to 10]` …

Units: Integer code, 5 to 10

Type: Axis-Specific

Remembered: Automatically

Function: This command is "recycled" for a different use in MicroMirror axes than for piezo. In the context of a MicroMirror axis this command is used to set the MicroMirror travel range. Settings is automatically saved into non-voltatile memory, however controller needs a system RESET or RESTART for setting to take effect.

PR[Axis Name] =	MicroMirror Range in degrees
5	5
6	6
8 (default)	8

10	10

Example:

```
PR R=5                  Sets range of R axis as 5 degrees
:A
PR R? S?                Queries the range of axes
R=5 S=8 :A
```

Setting Filter Frequency

The MEM MicroMirror is analogous to a spring-mass system, and like a spring mass system it has a tendency to ring. Below is the frequency response and step response of the 1200um MEMS mirror as provided by the manufacturer.

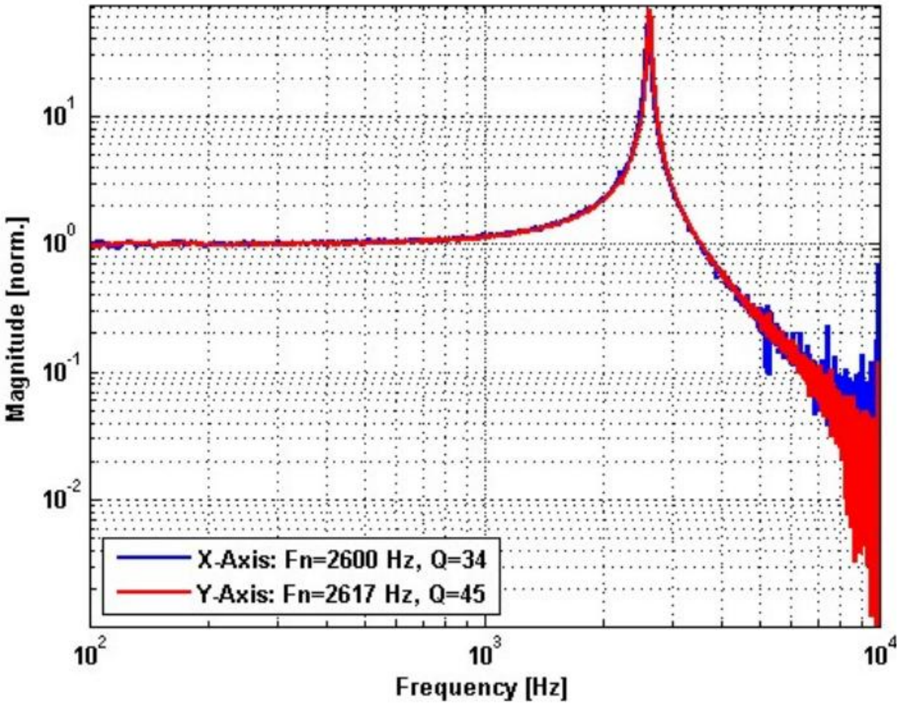


Fig: Frequency response of the 1200um MEMS MicroMirror, with resonance near 2600Hz.

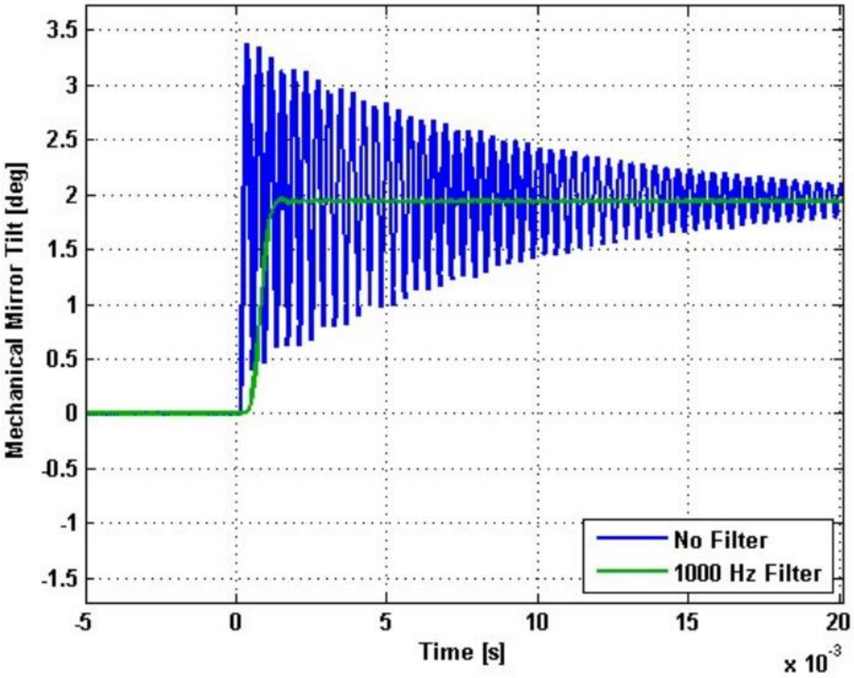


Fig: Step response of the 1200um MEMS MicroMirror

The MicroMirror card has a tunable Bessel filter to filter out resonating frequency elements from the input which helps reduce the mechanical ringing. The MEMS manufacturer recommends setting the filter frequency no more than 1/3 of the resonance frequency (about 850Hz), although based on ASI's testing it is possible to go higher. The default filter frequency is 400Hz, and it may be changed using the **BACKLASH** command.

Multi-axis function

The multi-axis firmware module lets the MicroMirror Card generate patterns like circle and spiral. It is implemented but has not yet undergone extensive testing. It can also be used for XY stages if the firmware module is present.

Command:	**MULTIMV**
Shortcut:	**MM**
Format:	[**Addr#] MM [X=radius] [Y= speed] [Z= width] [F=mode**
Units:	None
Type:	Card-Addressed
Remembered:	Using **[Addr#]SS Z** (X, Y, Z, and F settings)
Function:	The MULTIMV command allows several common multi-axis move patterns to be executed. Presently the patterns supported include circles and spirals. If users have other special requirements, they should contact ASI for assistance.

The command, without any arguments, initiates the multi-axis pattern move.

The patterns are initiated from the current stage position. The movement is parameterized in terms of the speed (feed rate) in mm/sec and pattern parameters. For circles, the radius in millimeters is the only required parameter. For spirals the width per spiral turn in millimeters is required as well as the maximum radius.

The mode is a bit-mapped character that determines the characteristics of the motion. The mode bits are used according to the following table.

Bit	**Set**	**Clear**
0	Lead-in Move Used	No Lead-in Move
1	Controlled acceleration along path, set by ACCEL command, to programmed speed.	No controlled acceleration
2	Move pattern repeated indefinitely	Only single cycle of move pattern executed
3	Reserved	
4	Reserved	
5	Reserved	
6 & 7	Motion pattern selector bits 7 & 6: 00 Reserved 01 Circle 10 Reserved 11 Spiral	

Circles: Lead-in move assumes start location is center of circle and moves the X-axis by the specified radius before the circular motion is started.

Spirals: Spirals start at the current location. Presently, no lead-in move is programmed. The spiral equation is $r = width \times \theta / 2\pi$. . Motion continues to the maximum radius. If mode BIT2 is set, the motion then continues spiraling inward, and continues inward and outward until halted.

Example:

```
3MM x=0.02 y=5 z=0.02 f=68
:A
```

This command sets up the parameters to do a circle pattern for axes on card with address 3.

```
3MM  x=0.02 y=2 z=0.002 f=196
:A
```

This command sets up the parameters to do a spiral pattern for axes on card with address 3.

```
3MM
```

Initiates the patterns

```
3MM
```

The second time disables the routine

Single-axis functions

The single-axis firmware module lets the MicroMirror card generate patterns including ramps, triangles and pulses. This module is intended to run off a 4Khz internal clock, though provision has been made to accept an external clock on the TTL input (only lightly tested). They can also be configured to output TTL pulses upon completion of each cycle. For both these TTL functions, the corresponding physical connectors need to be in place.

Provision has been made for the user to control the following features of the single-axis output pattern:

- Peak-to-peak amplitude (**SAA**)
- Period (**SAF**)
- Pattern and miscellaneous configuration settings (**SAP**)
- Offset or initial center point (**SAO**)

Note that some of these settings are also used by the SPIM firmware module. When a SPIM scan is initiated then any active single-axis function on the MicroMirror card is stopped. When SPIM is armed for a piezo card any active single-axis function on the card is stopped.

The **SAM** command is used to activate and also inactivate single-axis patterns.

While a single-axis mode is active (i.e. when the pattern is being generated and output), the axis move status character (accessed by **RDSTAT [axis]+**) is set to ***A***.

When single-axis mode is active then moving the axis (either using **MOVE** or **MOVEREL** or by using the joystick or wheels) will automatically update the offset. Thus, the single-axis pattern always has the same position on start-up as where it last was used, unless the **SAO** command has been used in the interim to select a new offset. However, moving the axis position while the single-axis mode is idle does not change the offset.

For MicroMirror cards with fast DACs (usually axes A and C, or axes P and Q), the single-axis functions update the fast DACs at a rate of 40kHz beginning with firmware version 2.8. Otherwise, the DAC update rate is 4kHz divided by the number of card axes. e.g. 1kHz for typical 4-channel MicroMirror cards and 4kHz for ADEPT piezo cards (but the mechanical response of piezos is much slower than 4kHz).

While the single-axis mode is active, changes to the parameters (e.g. executing **SAA**, **SAF**, **SAP**, or **SAO**) automatically take effect as of firmware version 2.82. However, axis synchronization may be lost during the update, so if synchronization is important then it should be performed again after any parameter changes are made by executing **SAM** with mode 3 again.

Command: **SAA**

Format: **SAA [axis]=###** …

Units: Axis units

Type: Axis-Specific

Remembered: Using **[Addr#]SS Z**

Function: This command sets the peak-to-peak amplitude of the pattern. Negative numbers will reverse the direction of the ramp pattern and make the first triangle sweep negative instead of positive.

Example: **`SAA R=1000`**

`:A`

Sets 1 degree as the total (peak-to-peak) amplitude for (MicroMirror) axis R.

Command: **SAF**

Format: **`SAF [axis]=###`** …

Units: milliseconds or clock edges

Type: Axis-Specific

Remembered: Using **`[Addr#]SS Z`**

Function: This command sets the period of the pattern, in units of milliseconds in case of internal clock (default, see **`SAP`**) or in number of clock edges for external clock. Note that the triangle pattern and square wave pattern will automatically force themselves to have a period of an even number of milliseconds, even if the user specifies a period of an odd number of milliseconds. Also note that the filter (tunable with **`BACKLASH`**) the actual waveform output may be significantly distorted when the period is set to be less than 10msec. When period is set to be 1msec, the resulting behavior is undefined.

Example: **`SAF r=10`**

`:A`

Sets up the period of the pattern at 10msec or 100Hz.

Command: **SAM**

Format: **`SAM [axis]=###`** …

Units: Integer code, 0-3 (see below)

Type: Axis-Specific

Remembered: No

Function: Sets the single-axis mode according to the integer code.

Code	Meaning
0	Puts single-axis mode in idle state (i.e. stops it if running)
1	Puts the single-axis mode in active state (i.e. starts generating the pattern)
2	Arms the trigger; on receipt of a TTL trigger pulse the single-axis mode becomes active
3	Makes the single-axis mode active and restarts the pattern of any other axis on the same card so they will be synchronized

Example: **`SAM r=0`**

`:A`

Disables the routine for the R axis

```
SAM r=1
:A
```

Enables the routine for the R axis

```
SAM s=2
:A
```

Arms the routine for the S axis. The routine for S will start running on receipt of a TTL pulse

```
SAM s=3
:A
```

Enables the routine for the S axis. Also the routine for R axis is reset, so they will both run in sync.

Command: **SAP**

Format: `SAP [axis]=###` …

Units: Integer code, 0-255 (see below)

Type: Axis-Specific

Remembered: Using `[Addr#]SS Z`

Function: This command sets the type of pattern to generate and configures the clocks. The parameter is a bit-mapped number that determines the characteristics of the motion, with the lowest bits determining the type of pattern. The code is interpreted according to the following table:

Bit	Clear	Set
7	Internal Clock	External clock on TTL input
6	Polarity of Clk, positive edge	Polarity of Clk, negative edge
5	No TTL out	TTL out
4	Polarity of TTL out, active high	Polarity of TTL out, active low
3	reserved	Reserved
2-0	000 Ramp/sawtooth (code 0) 001 Triangle (code 1) (period always even number of msec) 010 Square wave (code 2) (period always even number of msec)	

Example:

```
SAP r=0
:A
```

Sets up for ramp (sawtooth) pattern, running off internal clock with no TTL outs

```
SAP r=129
:A
```

Sets up for triangle pattern, running off positive edge external TTL clock with no TTL outs.

```
SAP r=161
:A
```

Sets up for triangle pattern, running off positive edge external TTL clock with TTL outs. A 250usec pulse is put out at the start of the pattern.

Command: **SAO**

Format: `SAO [axis]=### …`

Units: Axis units

Type: Axis-Specific

Remembered: Using `[Addr#]SS Z`

Function: This command sets the position of the center position of the single-axis pattern. For example, if the offset is 1000 and the peak-to-peak amplitude is set to 1000, the pattern will go between positions 500 and 1500.

Note that manual moves (e.g. with joystick or wheels) while a single-axis pattern is being generated automatically adjust the single-axis offset value, such that this command may not be needed.

As of firmware v2.82, using the "+" operator instead of specifying a position will store the current position to the offset for the specified axis. For example, `SAO A+` set the position of axis A to the center position of the single-axis pattern.

Example:

```
SAO R=1000
:A
```

Sets offset of the offset of the pattern to be 1 degree for (MicroMirror) axis R.

Change log

First Draft	Vik 3/28/2013
Singleaxis function	Vik 4/4/2013
MM Cal command	Vik 4/9/2013
SAM cmd edited	Vik 6/5/2013
PZINFO cmd added	Vik 6/6/2013
Formatting update to match other manuals	Vik 7/3/2013
Added SPIM commands, updated formatting	Jon 11/11/2013
Implemented PM cmd for MicroMirror	Vik 11/13/2013

Implemented PR cmd for MicroMirror	Vik 3/6/2014