# Desktop Optical Fiber Amplifier
# AMP-FL8612-OB
# Programing Manual

2-1-15, Ohara, Fujimino, Saitama, 356-8502, Japan
Tel: +81-49-278-7829, Fax: +81-49-263-9328
E-mail: info@fiberlabs.co.jp
URL: https://www.fiberlabs.com

# Table of Contents


**1. Remote Control Interface ..... 3**

1.1. RS-232C interface ..... 3

1.2. GP-IB interface ..... 3

**2. SCPI Status System ..... 4**

2.1. Summary of status system ..... 4

2.2. Type of registers in register group ..... 5

2.3. Status Byte Register and Service Request Enable Register ..... 5

2.4. Standard Event Status Group ..... 6

2.5. Operation Status Group ..... 6

2.6. Questionable Status Group ..... 6

2.7. Error queue ..... 7

**3. Basics of SCPI Commands ..... 8**

3.1. Rules of command syntax ..... 8

3.2. Command separator ..... 8

**4. Common Commands ..... 10**

4.1. SCPI common commands ..... 10

*CLS ..... 10

*ESE ☐<integer> / *ESE? ..... 10

*ESR? ..... 10

*IDN? ..... 10

*OPC / *OPC? ..... 10

*RST ..... 11

*SRE ☐<integer> / *SRE? ..... 11

*STB? ..... 11

*TST? ..... 11

*WAI ..... 11

4.2. STATus ..... 12

:STATus:OPERation:CONDition? ..... 12

:STATus:OPERation:NTRansition ☐<integer> / NTRansition? ..... 12

:STATus:OPERation:PTRansition ☐<integer> / PTRansition? ..... 12

:STATus:OPERation:ENABle ☐<integer> / ENABle? ..... 12

:STATus:OPERation[:EVENt]? ..... 12

:STATus:PRESet ..... 13

:STATus:QUEStionable:CONDition? ..... 13

:STATus:QUEStionable:NTRansition ☐<integer> / NTRansition? ..... 13

:STATus:QUEStionable:PTRansition ☐<integer> / PTRansition? ..... 13

:STATus:QUEStionable:ENABle ☐<integer> / ENABle? ..... 13

:STATus:QUEStionable[:EVENt]? ..... 14

4.3. SYSTem ..... 14

:SYSTem:ACKnowledge ☐ { OFF | ON | 0 | 1 } / ACKnowledge? ..... 14

:SYSTem:ERRor? 14
:SYSTem:REBoot 14
:SYSTem:DEFault:LOAd 14

**5. Product Specific Commands 15**

5.1. Monitor commands 15
[:] MONIN, 1 15
[:] MONOUT, 1 15
[:] MONRET, 1 (*only models with the back-reflection monitor) 15
[:] MONCTMP 15
[:] MONLDC, 1 15
[:] MONLDT, 1 15
5.2. Adjust commands 16
[:]SETMOD,1 [ , <MODE> ] 16
[:]SETACC,1 [ , <set> ] 16
[:]SETALC,1 [ , <set> ] 16
[:]SETAGC,1 [ , <set> ] (*only models with AGC) 16
[:]SAVEREF 17
5.3. Alarm commands 17
[:]ALMSTAT 17
[:]ALMOUT,1 [, <th>, <dtct>, <hyst> ] 17
[:]ALMIN,1 [, <th>, <dtct>, <hyst> ] 17
[:]ALMRET,1 [, <th>, <dtct>, <hyst> ] (*only models with the back-reflection monitor) 18
[:]SETIL [ , <set> ] 18
[:]ALMCTMP [, <th>, <dtct>, <hyst> ] 18
[:]ALMLDC,1 [, <th>, <dtct>, <hyst> ] 19
[:]ALMLDT,1 [, <th>, <dtct>, <hyst> ] 19

**Appendix 1. Command Table 20**

**Appendix 2. Error Code List 22**

**Appendix 3. Version History 23**

# 1. Remote Control Interface

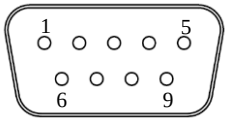
This instrument has a RS-232C and GP-IB interface port. Users can control and monitor the instrument from a remote PC with these ports.

## 1.1. RS-232C interface

Specifications of the RS-232C interface port are shown in the following table.

**Table 1-1. RS-232C port specifications**

| | |
|---|---|
| **Connector** | D-sub9p male |
| **Standard** | RS-232C |
| **Cable type** | straight |
| **Baud rate *** | 9600 / 19200 / 38400 / 57600 |
| **Parity** | none |
| **Data length** | 8 bit |
| **Stop bit** | 1 bit |
| **Flow control** | none |
| **Delimiter *** | ‘CR’ / LF” / ‘CR+LF’ |




| | |
|---|---|
| pin 1: | N/C |
| pin 2: | RXD |
| pin 3: | TXD |
| pin 4: | N/C |
| pin 5: | GND |
| pin 6: | N/C |
| pin 7: | RTS |
| pin 8: | CTS |
| pin 9: | N/C |

* Setting from front panel interface

## 1.2. GP-IB interface

Specifications of the GP-IB interface port are shown in the following tables.

**Table 1-2. GPIB port specifications**

| | |
|---|---|
| **Standard** | IEEE 488.2 |
| **Address *** | 1 to 30 |
| **Delimiter *** | ‘CR’ / LF” / ‘CR+LF’ +EOI |

* Setting from front panel interface

**Table 1-3. Functional specifications**

| **Subset (Function)** | **Description** |
|---|---|
| **AH1 (Acceptor handshake)** | All capabilities of send handshake |
| **T2 (Talker)** | All capabilities of receive handshake |
| **L2 (Listener)** | Basic talker function |
| **SH1 (Source handshake)** | Basic listener function |
| **SR1 (Service request)** | All capabilities of service request |
| **RL2 (Remote local)** | Local lockout function not provided |
| **PP0 (Parallel port)** | not provided |
| **DC0 (Device clear)** | not provided |
| **DT0 (Device trigger)** | not provided |
| **C0 (Controller)** | not provided |

# 2. SCPI Status System

## 2.1. Summary of status system

SCPI specifies a status system to check various conditions of instruments. The status system in this instrument has the following registers and register groups.

- Status Byte Register
- Service Request Enable Register
- Standard Event group
- Operation Status Group
- Questionable Status Group

The following illustrates the SCPI status register system.

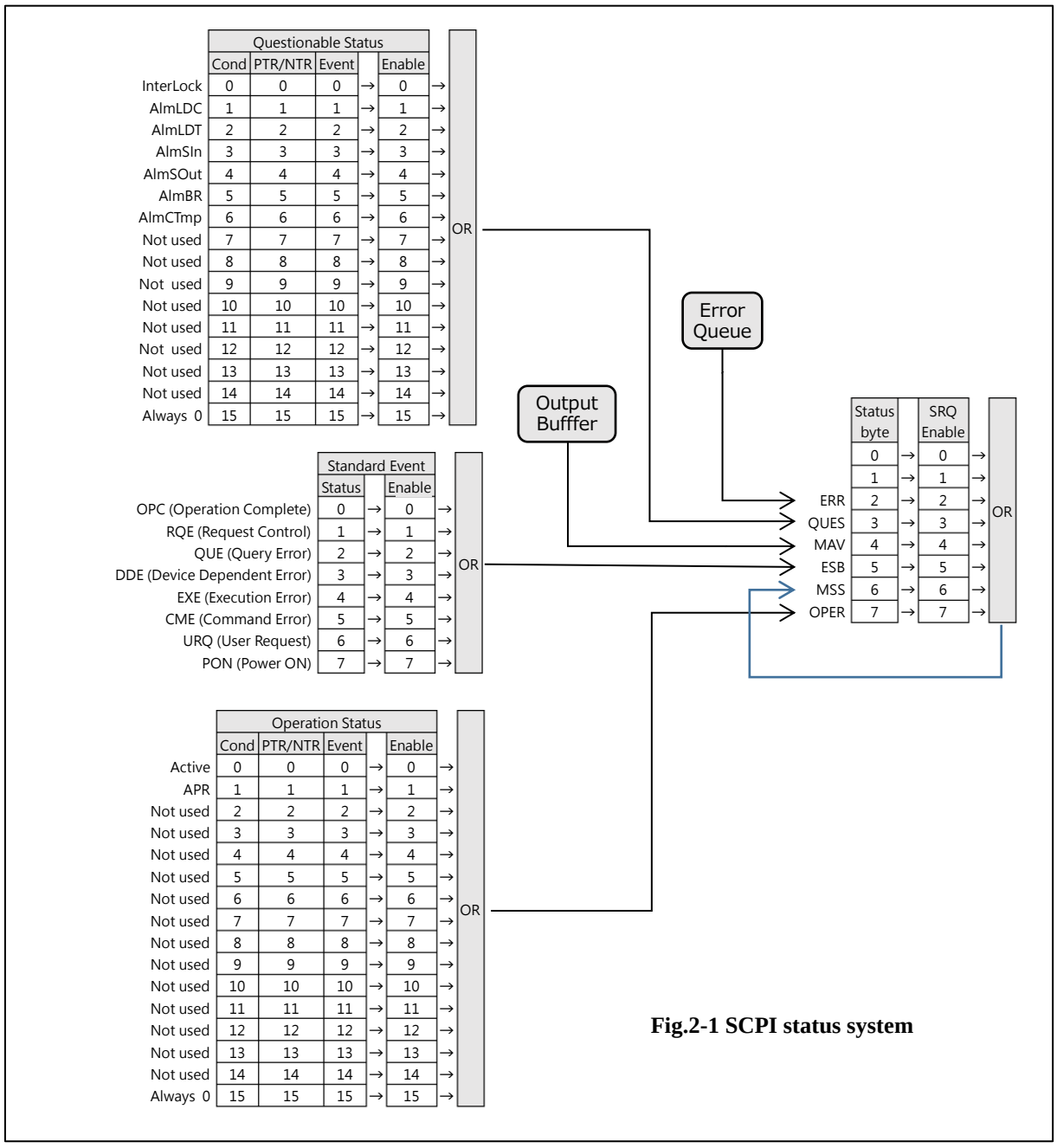



**Fig.2-1 SCPI status system**

## 2.2. Type of registers in register group

The Operation Status Group and Questionable Status Group consist of the registers and filters listed below. The Standard Event Status Group is similar but it has only an event register and enable register.

- **Condition Register :**

  This resister shows current conditions of the system. Each of bits are updated by the system and read-only for user.

- **Transition Filter :**

  Negative Transition (NTR) filter specifies the bits in the condition register that will output to the event register when their value changes from 1 to 0.

  Positive Transition (PTR) filter specifies the bits in the condition register that will output to the event register when their value changes from 0 to 1.

**Table 2-1. Working of Transition Filters**

| Condition | NTR | PTR | | Event |
|---|---|---|---|---|
| No-changed | × | × | ⇨ | Always 0 |
| 0 ⇨ 1 | 0 | 1 | ⇨ | 1 |
| 1 ⇨ 0 | 1 | 0 | ⇨ | 1 |
| 0 ⇨ 1 or 1 ⇨ 0 | 1 | 1 | ⇨ | 1 |
| × | 0 | 0 | ⇨ | Always 0 |
| × : don't care | | | | |

- **Event Register :**

  This register shows transition events of the condition register filtered by the NTR/PTR filter. Bits in this register are latched until cleared by a query or *CLS command.

- **Enable Register :**

  This register specifies the bits in the event register that output to the summary bit in the Status Byte Register.

## 2.3. Status Byte Register and Service Request Enable Register

The Status Byte Register shows a summary (OR of all bits) of other status groups, and status of output buffer and error queue. Users can specify which bits in the Status Byte Register trigger a service request by the Service Request Enable Register. The information of its each bit is as below.

**Table 2-2. Status Byte Register**

| Bit | Event | Description | Decimal |
|---|---|---|---|
| **0** | unused | Always 0 | 1 |
| **1** | unused | Always 0 | 2 |
| **2** | EAV | Error queue is not empty | 4 |
| **3** | QUS | Summary of Questionable Status | 8 |
| **4** | MAV | Output buffer is not empty | 16 |
| **5** | ESB | Summary of Standard Event Status | 32 |
| **6** | MSS | Summary of Status Byte | 64 |
| **7** | OPS | Summary of Operation Status | 128 |

## 2.4. Standard Event Status Group

The Standard Event Status Register shows the status of standard events specified in IEEE 488.2. Users can specify which bits assert the ESB bit in the Status Byte Register by the Standard Event Enable Register. The information of its each bit is as below.

**Table 2-3. Standard Event Status Register**

| Bit | Event | Description | Decimal |
|---|---|---|---|
| **0** | OPC | Operation complete : All pending operations are completed | 1 |
| **1** | unused | Always 0 | 2 |
| **2** | QYE | Query error : Tried to read from an empty buffer or a new command line overwritten a previous buffered query | 4 |
| **3** | DDE | Device dependent error : Other device depending errors | 8 |
| **4** | EXE | Execution error : Failed to execute a command for invalid parameters, etc. | 16 |
| **5** | CME | Command error : Invalid command syntax | 32 |
| **6** | unused | Always 0 | 64 |
| **7** | PON | Power on | 128 |

## 2.5. Operation Status Group

The Operation Condition Register shows the status of the system operation. The Operation Event Register shows a change of the Condition Register filtered by the Operation NTR/PTR Filters. Users can specify which bits assert the OPS bit in the Status Byte Register by the Operation Event Enable Register. The information of its each bit is as below.

**Table 2-4. Operation Condition Register**

| Bit | Event | Description | Decimal |
|---|---|---|---|
| **0** | Active | Output status is active | 1 |
| **1** | APR | Output status is active and APR is working | 2 |
| **2** | | | 4 |
| ⋮ | Unused | Always 0 | ⋮ |
| **15** | | | 32768 |

## 2.6. Questionable Status Group

The Questionable Condition Register shows the status of the limitations or protects. The Questionable Event Register shows a change of the Condition Register filtered by the Questionable NTR/PTR Filters. Users can specify which bits assert the QUS bit in the Status Byte Register by the Questionable Event Enable Register. The information of its each bit is as below.

**Table 2-5. Questionable Condition Register**

| Bit | Event | Description | Decimal |
|---|---|---|---|
| **0** | Interlock | APR or Remote interlock is working | 1 |
| **1** | AlmLDC | Pump-LD drive current alarm | 2 |
| **2** | AlmLDT | Pump-LD temperature current alarm | 4 |
| **3** | AlmSIn | Signal input level alarm | 8 |
| **4** | AlmSOut | Signal output level alarm | 16 |
| **5** | AlmBR | Back-reflection alarm | 32 |
| **6** | AlmCTmp | Case temperature alarm | 64 |
| **7** | | | 128 |
| ⋮ | unused | Always 0 | ⋮ |
| **13** | | | 32768 |

| | | | |
|---|---|---|---|
| **14** | ComErr | Internal Communication Error | 16384 |
| **15** | unused | Always 0 | 32768 |

## 2.7. Error queue

Error messages with an error code are buffered to an error queue when any error events occurred. Users can fetch the oldest message from the queue with SYSTem:ERRor command, and the fetched message is removed from the queue. This system can buffer up to 23 messages, the 24$^{th}$ message will be replaced with the overflow message "-350, Queue Overflow" and the overflown message will be lost. Refer to **Appendix.2 Error Code List** for the detail of error messages.

# 3. Basics of SCPI Commands

## 3.1. Rules of command syntax

The symbols in a command syntax is as follows. All commands are sent and received using ASCII characters.

**Table 3-1. Symbols in command syntax**

| Symbol | Meanings |
|---|---|
| **[···]** | Optional parameters, the default value is used if this parameter is omitted. |
| **<···>** | Required parameters to execute commands, integer, string, etc. |
| **{···\|···}** | Select from multiple parameters, {A \| B} means either A or B, but not both simultaneously |
| □ | White space |
| **Upper-Case**<br>**Lower-Case** | Upper-case characters in commands mean a short form. Upper-case + Lower-case characters mean a long form. For shorter program lines, use the short form. For better program readability, use the long form. |

All command are case-insensitive, so the following commands are interpreted as the same command.

```
> :STATus:OPERation:EVENt?
> :STATUS:OPERATION:EVENT?
> :status:operation:event?
> :stat:oper:even?
> :stat:OPER:even?
```

## 3.2. Command separator

Users can send some commands together by separating with semi-colon (;).

```
> :STAT:OPER:EVEN?;:STAT:QUES:EVEN?
```

The above is equivalent to the following two commands:

```
> :STAT:OPER:EVEN?
> :STAT:QUES:EVEN?
```

Users can omit a command line to send the same level commands together by separating with semi-colon.

```
> :STAT:OPER:ENAB 16;ENAB?
```

The above is equivalent to the following two commands:

```
> :STAT:OPER:ENAB 16
> :STAT:OPER:ENAB?
```

The SCPI common commands can be executed regardless the current command level as the followings.

```
> :STAT:OPER:ENAB 16;*STB?;ENAB?
> :STAT:OPER:ENAB?;*STB?;:STAT:QUES:ENAB?
```

The multiple replies is separated with semi-colon as the followings.

```
> :STAT:OPER:ENAB?;ENAB 5;ENAB?
< 16;5                          // * reply of the 1st and 3rd command
```

The subsequent commands are not executed and its output buffer is cleared if any errors occurred in an intermediate command.

```
> :STAT:OPER:ENAB 16;ENAB?;ENAB -10;ENAB? // * the 4th command is not executed, the 3rd has an invalid value
<                                         // * reply of the 2nd command is discarded
```

# 4. Common Commands

## 4.1. SCPI common commands

### *CLS

The command clears all Event Registers, and the Status Byte Register.

**Example**
```
> *CLS
```

### *ESE □<integer> / *ESE?

The command is to set / query the Standard Event Enable Register. This value is reset to the default value when the system reboot.

<integer> 0 to 255, default: 0

**Example**
```
> *ESE?
< 0
> *ESE 255;*ESE?
< 255
```

### *ESR?

The command is to query the Standard Event Status Register. The register is cleared after being read.

**Example**
```
> *ESR?;*ESR?
< 127;0          // * the register is cleared at the first query
```

### *IDN?

The command is to query the instrument's identification string.

**Example**
```
> *IDN?
< FIBERLABS,WLTF-FL-O,1.0.0.0     // * Manufacturer, Product, Firmware revision
```

### *OPC / *OPC?

The command is to set / query the OPC bit in the Standard Event Status Register when the pending operations are completed. This system has no overlapped commands, so *OPC? always returns 1.

**Example**
```
> *OPC?
< 1
```

## *RST

The command reset all settings to factory default settings. The system will reboot automatically after this command executed.

**Example** > ***RST** // * The system will reboot after some seconds

## *SRE □<integer> / *SRE?

The command is to set / query the Service Request Enable Register. This value is reset to the default value when the system reboot.

<integer> 0 to 255, efault: 0

**Example**
```
> *SRE?
< 0
> *SRE 255;*SRE?
< 255
```

## *STB?

The command returns the Status Byte Register.

**Example**
```
> *STB?
< 251
```

## *TST?

The command execute a self-test and returns its result. 0 means no errors

**Example**
```
> *TST?
< 0
```

## *WAI

The command wait the pending operations are completed. All commands of this system are the sequential command, so this command is ignored

**Example** > ***WAI** // * ignored

## 4.2. STATus

### :STATus:OPERation:CONDition?

The command returns the Operation Status Condition Register.

**Example**
```
> :STAT:OPER:COND?
< 0
```

### :STATus:OPERation:NTRansition □<integer> / NTRansition?

The command is to set / query the Operation Status Negative Transition Filter. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 0

**Example**
```
> :STAT:OPER:NTR?
< 0
> :STAT:OPER:NTR 32767;NTR?
< 32767
```

### :STATus:OPERation:PTRansition □<integer> / PTRansition?

The command is to set / query the Operation Status Positive Transition Filter. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 32767

**Example**
```
> :STAT:OPER:PTR?
< 32767
> :STAT:OPER:PTR 0;PTR?
< 0
```

### :STATus:OPERation:ENABle □<integer> / ENABle?

The command is to set / query the Operation Status Enable Register. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 0

**Example**
```
> :STAT:OPER:ENAB?
< 0
> :STAT:OPER:ENAB 32767;ENAB?
< 32767
```

### :STATus:OPERation[:EVENt]?

The command returns the Operation Status Event Register. The register is cleared after being read.

**Example**
```
> :STAT:OPER:EVEN?
< 0
> :STAT:OPER?
< 0
```

## :STATus:PRESet

The command clears event registers and resets enable registers and transition filters in the Operation Status Group and Questionable Status Group.

**Example**
```
> :STAT:PRES
```

## :STATus:QUEStionable:CONDition?

The command returns the Questionable Status Condition Register.

**Example**
```
> :STAT:QUES:COND?
< 0
```

## :STATus:QUEStionable:NTRansition □<integer> / NTRansition?

The command is to set / query the Questionable Status Negative Transition Filter. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 0

**Example**
```
> :STAT:QUES:NTR?
< 0
> :STAT:QUES:NTR 32767;NTR?
< 32767
```

## :STATus:QUEStionable:PTRansition □<integer> / PTRansition?

The command is to set / query the Questionable Status Positive Transition Filter. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 32767

**Example**
```
> :STAT:QUES:PTR?
< 32767
> :STAT:QUES:PTR 0;PTR?
< 0
```

## :STATus:QUEStionable:ENABle □<integer> / ENABle?

The command is to set / query the Questionable Status Enable register. This value is reset to the default value when the system reboot.

<integer> 0 to 32767, default: 0

**Example**
```
> :STAT:QUES:ENAB?
< 0
> :STAT:QUES:ENAB 32767;ENAB?
< 32767
```

### :STATus:QUEStionable[:EVENt]?

The command returns the Questionable Status Event Register. The register is cleared after being read.

**Example**
```
> :STAT:QUES:EVEN?
< 0
> :STAT:QUES?
< 0
```

## 4.3. SYSTem

### :SYSTem:ACKnowledge □ { OFF | ON | 0 | 1 } / ACKnowledge?

The command is to set / query the setting of the command acknowledge message. Non-query commands return the following messages when this setting to ON.

| | |
|---|---|
| **OK** | : Command is completed |
| **??FAIL** | : Execution failure |
| **??ARG** | : Invalid argument |
| **??CMD** | : Syntax error |

This setting is reset to the default value when the system reboot.

| | |
|---|---|
| OFF | Set acknowledge response to OFF * default |
| ON | Set acknowledge response to ON |
| 0 | = OFF |
| 1 | = ON |

**Example**
```
> :SYST:ACK ON
< OK
> :SOURC:WAV:1300
< OK
```

### :SYSTem:ERRor?

The command is to fetch the error code and message from the error queue. “0, “No error”” will be returned if the error queue is empty.

**Example**
```
> :SYST:ERR?
< 0,"No error"
```

### :SYSTem:REBoot

The command reboots the system.

**Example**
```
> :SYST:REB          // * The system will reboot after some seconds
```

### :SYSTem:DEFault:LOAd

The command loads the factory default settings. The system will reboot automatically after loading.

**Example**
```
> :SYST:DEF:LOA      // * The system will reboot after some seconds
```

# 5. Product Specific Commands

The product specific commands are as follows. Please note that these commands are not compatible for SCPI syntax rules for compatibility with our older products.

## 5.1. Monitor commands

### [:] MONIN, 1

The command returns the signal input level monitor (dBm).

**Example**
```
> MONIN,1
< -0.05
```

### [:] MONOUT, 1

The command returns the signal output level monitor (dBm).

**Example**
```
> MONOUT,1
< 23.5
```

### [:] MONRET, 1 (*only models with the back-reflection monitor)

The command returns the back-reflection level monitor at the optical output port (dBm).

**Example**
```
> MONRET,1
< 6.5
```

### [:] MONCTMP

The command returns the case temperature monitor (°C).

**Example**
```
> MONCTMP
< 26.5
```

### [:] MONLDC, 1

The command returns the pump-LD drive current monitor (mA).

**Example**
```
> MONLDC,1
< 2000
```

### [:] MONLDT, 1

The command returns the pump-LD temperature monitor (°C).

**Example**
```
> MONLDT,1
< 36.5
```

## 5.2. Adjust commands

### [:]SETMOD,1 [ , <MODE> ]

The command is to set / query the pump-LD drive mode.

| | |
|---|---|
| MODE | Set the pump-LD drive mode |
| 0 | = ALC (Auto-Current-Control) |
| 1 | = ACC (Auto-Light-Control) |
| 2 | = AGC (Auto-Gain-Control) *only models with AGC |

**Example**

```
> SETMOD,1
< SETMOD,1,1        // query current setting = ACC
> SETMOD,1,0
< SETMOD,1,0        // new setting = ALC
```

### [:]SETACC,1 [ , <set> ]

The command is to set / query the pump-LD drive current for ACC. The setting by this command will be reset when the power is turned off or pump-LDs are interlocked. Use command 'SAVEREF' to save this settings.

| | |
|---|---|
| **<set>** | Drive current for ACC [mA] |

**Example**

```
> SETACC,1
< SETACC,1,2000.0       // query current setting = 2000 mA
> SETMOD,1,3000
< SETACC,1,3000.0       // new setting = 3000 mA (*not saved yet)
```

### [:]SETALC,1 [ , <set> ]

The command is to set / query the output power for ALC. The setting by this command will be reset when the power is turned off or pump-LDs are interlocked. Use command 'SAVEREF' to save this settings.

| | |
|---|---|
| **<set>** | Output power for ALC [dBm] |

**Example**

```
> SETALC,1
< SETALC,1,23.0       // query current setting = 23 dBm
> SETMOD,1,20
< SETALC,1,20.0       // new setting = 20 dBm (*not saved yet)
```

### [:]SETAGC,1 [ , <set> ] (*only models with AGC)

The command is to set / query the signal gain for AGC. The setting by this command will be reset when the power is turned off or pump-LDs are interlocked. Use command 'SAVEREF' to save this settings.

| | |
|---|---|
| **<set>** | Signal gain for AGC [dB] |

**Example**

```
> SETAGC,1
< SETAGC,1,20.0       // query current setting = 20 dB
> SETMOD,1,20
< SETAGC,1,16.0       // new setting = 16 dB (*not saved yet)
```

### [:]SAVEREF

The command is to save ACC/ALC/AGC temporary settings.

**Example**
```
> SAVEREF
< OK              // Saved settings
```

## 5.3. Alarm commands

### [:]ALMSTAT

The command is to query the current alarm status by a hexadecimal number. Each bit indicates the state of alarm ( 0:Pass / 1:Fail ). The assignment of each alarm is shown in the following table.

| bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|
| Alarm assignment | Unused | Unused | Unused | Unused | Case temp. / LD temp. | Input level / reflection | LD current | Output level |

**Example**
```
> ALMSTAT
< ALMSTAT,10          // 0x10 = B00010000 : output alarm is raised.
```

### [:]ALMOUT,1 [, <th>, <dtct>, <hyst> ]

The command is to set / query the output signal level alarm parameters. The arguments given '*' are skipped

| | |
|---|---|
| <th> | Alarm threshold [dBm] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [dB] |

**Example**
```
> ALMOUT,1
< ALMOUT,1,10,1,0.5       // Alarm will rise at < 10dBm, recover at >10.5dBm
> ALMOUT,1,7,*,*
< ALMOUT,1,7,1,0.5        // Change threshold to 7dBm
```

### [:]ALMIN,1 [, <th>, <dtct>, <hyst> ]

The command is to set / query the input signal level alarm parameters. The arguments given '*' are skipped

| | |
|---|---|
| <th> | Alarm threshold [dBm] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [dB] |

**Example**
```
> ALMIN,1
< ALMIN,1,-10,1,0.5       // Alarm will rise at < -10dBm, recover at >-9.5dBm
> ALMIN,1,*,*,1
< ALMIN,1,-10,1,1.0       // recover at >-9.0dBm
```

## [:]ALMRET,1 [, <th>, <dtct>, <hyst> ] (*only models with the back-reflection monitor)

The command is to set / query the back-reflection alarm parameters. The arguments given '*' are skipped.
*The threshold of this alarm is set in the return-loss (output – reflection level).

| | |
|---|---|
| <th> | Alarm threshold [dB] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [dB] |

**Example**

```
> ALMRET,1
< ALMRET,1,20,1,0.5        // Alarm will rise at B/R ratio : < 20dB, recover at >20.5dB
> ALMRET,1,*,0,*
< ALMRET,1,20,0,0.5        // Alarm is ignored
```

## [:]SETIL [ , <set> ]

The command is to set / query the setting of Auto-Power-Reduction on the input / b-r alarm.

| | |
|---|---|
| <set> | Setting of Auto-Power-Reduction |
| 0 | = OFF |
| 1 | = ON |

**Example**

```
> SETIL
< SETMOD,1     // query current setting = ON
> SETIL,0
< SETIL,0      // new setting = OFF
```

## [:]ALMCTMP [, <th>, <dtct>, <hyst> ]

The command is to set / query the case temperature alarm parameters. The arguments given '*' are skipped.

| | |
|---|---|
| <th> | Alarm threshold [°C] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [°C] |

**Example**

```
> ALMCTMP
< ALMRET,1,40,1,0.5        // Alarm will rise at > 40°C, recover at <39.5°C
> ALMCTMP,50,*,*
< ALMRET,1,50,1,0.5        // Change threshold to 50°C
```

## [:]ALMLDC,1 [, <th>, <dtct>, <hyst> ]

The command is to set / query the pump-LD drive current alarm parameters. The arguments given '*' are skipped.

| | |
|---|---|
| <th> | Alarm threshold [mA] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [mA] |

**Example**

```
> ALMLDC,1
< ALMLDC,1,5000,1,5.0      // Alarm will rise at > 5000mA, recover at < 4995mA
> ALMLDC,1,3000,*,10
< ALMLDC,1,3000,0,10.0     // Alarm will rise at > 3000mA, recover at < 2990mA
```

## [:]ALMLDT,1 [, <th>, <dtct>, <hyst> ]

The command is to set / query the pump-LD temperature alarm parameters. The arguments given '*' are skipped.

| | |
|---|---|
| <th> | Alarm threshold [°C] |
| <dtct> | Alarm detection |
| 0 | = invalid |
| 1 | = valid |
| <hyst> | Alarm hysteresis [°C] |

**Example**

```
> ALMLDT,1
< ALMLDT,1,40,1,0.5      // Alarm will rise at > 40°C, recover at < 39.5°C
> ALMLDT,1,30,*,*
< ALMLDT,1,30,0,0.5      // Change threshold to 30°C
```

# Appendix 1. Command Table

| Syntax | Summary |
|---|---|
| **< IEEE488 Common Commands >** | |
| ***CLS** | Clear all Event Registers and the Status Byte Register |
| ***ESE** | Set the Standard Event Enable Register |
| ***ESE?** | Query the Standard Event Enable Register |
| ***ESR?** | Query the Standard Event Status Register |
| ***IDN?** | Query the instrument's identification string |
| ***OPC** | Set the OPC bit in the Standard Event Status Register |
| ***OPC?** | Query the OPC bit in the Standard Event Status Register |
| ***RST** | Reset to the factory default settings |
| ***SRE** | Set the Service Request Enable Register |
| ***SRE?** | Query the Service Request Enable Register |
| ***STB?** | Query the Status Byte Register |
| ***TST?** | Execute the self-test |
| ***WAI** | Wait the pending operations are completed |
| | |
| **< SCPI Standard Commands >** | |
| **:STATus:OPERation:CONDition?** | Query the Operation Condition Register |
| **:STATus:OPERation:ENABle ☐<integer>** | Set the Operation Event Enable register |
| **:STATus:OPERation:ENABle?** | Query the Operation Event Enable register |
| **:STATus:OPERation[:EVENt]?** | Query the Operation Event Register |
| **:STATus:OPERation:NTRansition ☐<integer>** | Set the Operation Negative Transition Filter |
| **:STATus:OPERation:NTRansition?** | Query the Operation Negative Transition Filter |
| **:STATus:OPERation:PTRansition ☐<integer>** | Set the Operation Positive Transition Filter |
| **:STATus:OPERation:PTRansition?** | Query the Operation Positive Transition Filter |
| **:STATus:PRESet** | Clears event and resets enable and filters |
| **:STATus:QUEStionable:CONDition?** | Query the Questionable Condition Register |
| **:STATus:QUEStionable:ENABle ☐<integer>** | Set the Questionable Event Enable register |
| **:STATus:QUEStionable:ENABle?** | Query the Questionable Event Enable register |
| **:STATus:QUEStionable[:EVENt]?** | Query the Questionable Event Register |
| **:STATus:QUEStionable:NTRansition ☐<integer>** | Set the Questionable Negative Transition Filter |
| **:STATus:QUEStionable:NTRansition?** | Query the Questionable Negative Transition Filter |
| **:STATus:QUEStionable:PTRansition ☐<integer>** | Set the Questionable Positive Transition Filter |
| **:STATus:QUEStionable:PTRansition?** | Query the Questionable Positive Transition Filter |
| | |
| **:SYSTem:ACKnowledge ☐{0\|1\|OFF\|ON}** | Set the command acknowledge message |
| **:SYSTem:ACKnowledge?** | Query the command acknowledge message |
| **:SYSTem:DEFaunlt:LOAd** | Load the factory default settings |
| **:SYSTem:ERRor?** | Fetch the error message from the error queue |
| **:SYSTem:REBoot** | Reboot the system |
| | |
| **< Non-SCPI Commands >** | |
| **[:]MONIN,1** | Query the input power monitor |
| **[:]MONOUT,1** | Query the output power monitor |
| **[:]MONRET,1** | Query the back-reflection monitor |
| **[:]MONCTMP,1** | Query the case temperature monitor |
| **[:]MONLDC,1** | Query the pump-LD driving current monitor |
| **[:]MONLDT,1** | Query the pump-LD temperature monitor |
| **[:]ACTIVE [,{0\|1}]** | Query and activate the optical output |
| **[:]SETMOD,1 [,<mode>]** | Query and set the driving mode |
| **[:]SETACC,1 [,<current>]** | Query and set the ACC target value *volatile |

| | |
|---|---|
| **[:]SETALC,1 [,<power>]** | Query and set the ALC target value *volatile |
| **[:]SETAGC,1 [,<gain>]** | Query and set the AGC target value *volatile |
| **[:]SAVEREF** | Save target values |
| **[:]ALMSTAT** | Query the status of alarms |
| **[:]ALMOUT,1 [,<th>,<dtct>,<hyst>]** | Query and set the output alarm parameters |
| **[:]ALMIN,1 [,<th>,<dtct>,<hyst>]** | Query and set the input alarm parameters |
| **[:]ALMRET,1 [,<th>,<dtct>,<hyst>]** | Query and set the back-reflection alarm parameters |
| **[:]SETIL [,<set>]** | Query and set the APR |
| **[:]ALMCTMP [,<th>,<dtct>,<hyst>]** | Query and set the case temperature alarm parameters |
| **[:]ALMLDC,1 [,<th>,<dtct>,<hyst>]** | Query and set the pump-LD current alarm parameters |
| **[:]ALMLDT,1 [,<th>,<dtct>,<hyst>]** | Query and set the pump-LD temperature alarm parameters |

# Appendix 2. Error Code List

| Code | Message | Summary |
|---|---|---|
| 2 | Auto power reduction | The APR is working |
| 1 | Turn on optical output | Turned on output |
| | | |
| 0 | No error | The error queue is empty |
| | | |
| -76 | GPIB-Rx overflow | The GPIB input buffer overflowed |
| -77 | GPIB-Tx overflow | The GPIB output buffer overflowed |
| -78 | GPIB-Tx overwrite | The GPIB output buffer is overwritten |
| -79 | GPIB-Tx empty | Tried to read from the empty buffer |
| -80 | GPIB-Handshake fault | The GPIB handshake process failed |
| -100 | Serial-Rx overflow | The RS232C input buffer overflowed |
| | | |
| -151 | Pump current alarm | Pump current alarm is raised |
| -152 | Pump temperature alarm | Pump temperature alarm is raised |
| -153 | Signal input alarm | Signal input alarm is raised |
| -154 | Signal output alarm | Signal output alarm is raised |
| -155 | Back reflection alarm | Back reflection alarm is raised |
| -156 | Case temperature alarm | Case temperature alarm is raised |
| -157 | Other alarms | Other alarm is raised |
| -158 | Feedback Limitation | Feedback operation reached its limitation |
| -270 | output interlock | Remote interlock is working |
| | | |
| -298 | Internal comm error | The internal board-to-board communication error |
| -299 | Mem cheksum error | Data in the internal non-volatile memory has error |
| -300 | NvMemory error | The internal non-volatile memory communication error |
| | | |
| -350 | Queue overflow | The error queue overflowed |

# Appendix 3. Version History

| Date | Ver. No. | Description |
|---|---|---|
| 2021.04.19 | 1.0.0 | New making |
| | | |
| 2021.07.15 | 1.1.0 | Modify the text / Fix the typo |
| | | 2.SCPI Status System, 4.Common Commands, 5.Product Specific Commands |

**Version rule**

**1 . 2 . 3**

3rd digit: Options / Customization
2nd digit: Fixed contents, compatible with old ver.
1st digit : Change in spec, NOT compatible with old ver.