

Peak Lock and Current Controller Command Set

List of commands for the ICE Peak Lock Servo. Please see [Common Laser Controller Command Set](#) for commands relating to the laser controller.

Phase?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
Phase?  
58
```

I₂C Command Number: 32

Returns the phase shift (in degrees) on the dither signal modulating the laser current.</div>

Phase

Description

Arguments:

```
[Float] PHASE
```

Example:

```
Phase 23  
22.5
```

I₂C Command Number: 33

Sets the phase shift to PHASE (in degrees) on the dither signal modulating the laser current. Returns the output of the command **Phase?** </div>

DitherA?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
DitherA?  
32
```

I₂C Command Number: 34

Returns the amplitude of the dither on the laser current. Output is integer from 0-255. </div>

DitherA

Description

Arguments:

```
[Int] AMP
```

Example:

```
DitherA 35  
35
```

I₂C Command Number: 35

Sets the amplitude of the current modulation to AMP. AMP is an integer ranging from 0 - 255. Command returns the output of the command **DitherA?** </div>

Dither?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
Dither?  
On
```

I₂C Command Number: 36

Returns the status of the dither on the laser current (on or off). </div>

Dither

Description

Arguments:

```
[ASCII] On/Off
```

Example:

```
Dither On
On
```

I₂C Command Number: 37

Enables or disables the dither on the laser current. Valid arguments are On or Off. Returns the output of the command **Dither?**. `</div>`

ReadVolt

Description

Arguments:

```
[Int] CHANNEL
```

Example:

```
ReadVolt 4
4.234
```

I₂C Command Number: 38

Returns the voltage measured on channel CHANNEL (in volts). The channels refer to:

1. Servo Out
2. Integrator
3. DC Error
4. Error Input
5. Laser Current (1V = 1A)
6. +2.5V Ref
7. -2.5V Re
8. Ground

`</div>`

Servo?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
Servo?
On
```

I₂C Command Number: 39

Returns the status of the laser servo (on or off). [</div>](#)

Servo

Description

Arguments:

```
[ASCII] On/Off
```

Example:

```
Servo On  
Off
```

I₂C Command Number: 40

Turns on the laser servo (engages the integrator). Returns the output of the command **Servo?**. [</div>](#)

DCOffst?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
DCOffst?  
-2.345
```

I₂C Command Number: 41

Returns DC offset (in volts) applied to the error signal. [</div>](#)

DCOffst

Description

Arguments:

```
[Float] OFFSET
```

Example:

```
DCOffst 2.340
```

```
2.342
```

I₂C Command Number: 42

Sets the DC offset (in volts) applied to the error signal to OFFSET. Returns the output of the command **DCOffst?**. [</div>](#)

Gain?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
Gain?  
25
```

I₂C Command Number: 43

Returns Servo Gain. Range is from 0-28 in steps of 2 dB. 0 is a special gain setting where there is no gain (error signal does not go to integrator). [</div>](#)

Gain

Description

Arguments:

```
[Int] GAIN
```

Example:

```
Gain 24  
24
```

I₂C Command Number: 44

Sets the Servo Gain. Range is from 0-28 in steps of 2 dB. 0 is a special gain setting where there is no gain (error signal does not go to integrator). Returns the output of the command **GetGain?** [</div>](#)

OpOffst?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
OpOffst?  
129
```

I₂C Command Number: 45

Returns Integrator Op-Amp's Offset Voltage adjustment value. Range is 0-255. </div>

OpOffst

Description

Arguments:

```
[Int] OFFSET
```

Example:

```
OpOffst 142  
142
```

I₂C Command Number: 46

Sets the Integrator Op-Amp's Offset Voltage adjustment value. This should be factory set and typically should be set to 128, but it controls the DC Error Input voltage that the servo locks to (should be 0V) OFFSET is an integer from 0-255. Returns the output of the command **OpOffst?**. </div>

SvOffst?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
SvOffst?  
2.341
```

I₂C Command Number: 47

Returns Servo Offset voltage (in volts). When the servo is engaged, this voltage is the starting voltage that the servo integrates from. </div>

SvOffst **Description**

Arguments:

```
[Float] OFFSET
```

Example:

```
SvOffst -1.232  
-1.23
```

I₂C Command Number: 48

Sets the Servo Offset voltage (in volts) to OUTPUT when the servo is turned off. When the servo is engaged, this voltage is the starting voltage that the servo integrates from. Returns the output of the command **SvOffst?**. </div>

DataChn? **Description**

Arguments:

```
No Arguments Taken
```

Example:

```
DataChn?  
2
```

I₂C Command Number: 49

Read Data Channel Mode. There are 3 modes for the Data Channel:

```
Mode 1: Record DC Error  
Mode 2: Record Error Input  
Mode 3: Record DC Error and Error Input
```

Note: In Mode 3, the ramp records two channels of data instead of one, doubling the amount of data stored. </div>

DataChn **Description**

Arguments:

```
[Int] CHN_MODE
```

Example:

```
DataChn 3  
3
```

I₂C Command Number: 50

Sets the Data Channel Mode. See **DataChn?** for more details Returns the output of the command **DataChn?**. </div>

RampSwp?

Description

Arguments:

```
No Arguments Taken
```

Example:

```
RampSwp?  
5.72
```

I₂C Command Number: 51

Reads the sweep range (in volts) for the ramp. </div>

RampSwp

Description

Arguments:

```
[Float] VOLTAGE
```

Example:

```
RampSwp 2.64  
2.65
```

I₂C Command Number: 52

Writes the sweep range (in volts) for the ramp. Range is from 0V to +10V. Returns the output from the command **RampSwp?**. The ramp will step from SvOffst - RampSwp/2 to SvOffst + RampSwp/2 by increments determined by RampSwp and RampNum and then return to SvOffst every time a RampRun command is initiated.

Note: RampSwp value is rounded and truncated to match range of the servo offset. </div>

RampNum? Description

Arguments:

```
No Arguments Taken
```

Example:

```
RampNum?  
100
```

I₂C Command Number: 53

Returns the number of data points to be acquired during the ramp. Number of data points is also the number of steps for the ramp.

Note that the amount of bytes stored by the ramp is 2*RampNum. The amount of bytes stored set how much data needs to be read back via the **ReadBlk** command.

</div>

RampNum Description

Arguments:

```
[Int] NUMBER
```

Example:

```
RampNum 100  
100
```

I₂C Command Number: 54

Sets the number of data points to be acquired during the ramp. Number of data points is also the number of steps for the ramp.

</div>

RampRun Description

Arguments:

```
No Arguments Taken
```

Example:

RampRun
Busy

I₂C Command Number: 55

Begins ramping the servo output and taking data according to the values set for **RampSwp** and **SvOffst** as described above. The Ramp begins at SvOffst - RampSwp/2 and ends of SvOffst + RampSwp/2. When the ramp is completed, the servo output is returned to its starting value of SvOffst. Data can be retrieved with the **ReadBlk** command. Board will not respond to any new commands while taking data. Returns status of execution of the ramp, either Busy or Finished. Returns fault if ramp is misconfigured. See **RampNum?** for details. Returns fail if laser is off. </div>

Poles?

Description

Arguments:

No Arguments Taken

Example:

Poles?
3 0

I₂C Command Number: 56

Returns two columns of data. First column is a number 1-5 representing the integrator pole value. Values are 3 kHz, 10 kHz, 30 kHz, 100 kHz and 300 kHz. 1 is the slowest integrator (3 kHz) and 5 is fastest integrator (300 kHz). The second column is a 0 when the differential pole is off and a 1 when it is on. When differential is on, the pole is ~3 times the integrator pole. So if PI pole is 30 kHz, the differential pole is 100 kHz. </div>

Poles

Description

Arguments:

[Int] INTEGRATOR
[Int] DIFFERENTIAL

Example:

Poles 3 1
3 1

I₂C Command Number: 57

Sets the integrator and differential pole positions. INTEGRATOR is a value from 1-5. DIFFERENTIAL is either on (1) or off (0). See **Poles?** for more details. </div>

EvtLOff?

Arguments:

Description

No Arguments Taken

Example:

```
EvtLOff?  
4
```

I₂C Command Number: 58

Reads the event address for turning the laser off. Address range is 0-7 where address 0 is no event.
</div>

EvtLOff

Arguments:

Description

[Int] ADDRESS

Example:

```
EvtLOff 2  
2
```

I₂C Command Number: 59

Sets the event address for turning the laser off. Address range is 0-7 where address 0 is no event.
</div>

From:

<https://www.vescent.com/manuals/> - **Product Manuals**

Permanent link:

<https://www.vescent.com/manuals/doku.php?id=ice:commands:peaklock>

Last update: **2018/11/26 17:33**

