User Manual

LucidControl RI4/RI8

4/8 Channel RTD Input USB Module

1 Introduction

This document describes the functionality of the LucidControl RI4/RI8 USB module with 4/8 Pt100/Pt1000 RTD input channels.

A genera description of the complete LucidControl product family can be found in the document *LucidControl User Manual*.

This document explains the topics that are specific to the RI4/RI8 USB module.

2 Setup and Installation

Fig. 1 RT8 Input Module

Fig. 1 shows the sketch of the RI8 module with 8 Pt100/Pt1000 input channels.

The upper IO connector (IO9 to IO16) is available on RI8 modules only.

Each IO connector has 8 IO pins - two for each input channel.

The intended use of the RI4/RI8 module is the acquisition of temperatures. The module must only be used for the intended use.

For this device it is explicitly stated that no potential (e.g. voltage) of any external power source must be applied to any connector of the module. The modules must only be used within the specified conditions.

2.1 Interface and Interconnection

2.1.1 USB Connection

LucidControl USB modules are connected to the computer by using a standard USB cable, which must not extend a length of 5 m. They are “bus powered” which means that the host computer supplies the module with power.

LucidControl RI4/RI8 module is rated with a maximum current of 40 mA.

Note
Please consider that the total power of one USB port is limited to 500 mA.

Note
Using an active USB-Hub with its own power supply allows the connection of additional devices in the case that the host is not able to supply them.

2.1.2 IO Connection

The LucidControl RI4/RI8 module provides 4/8 channels measuring temperatures of Pt1000 or Pt100 RTDs.

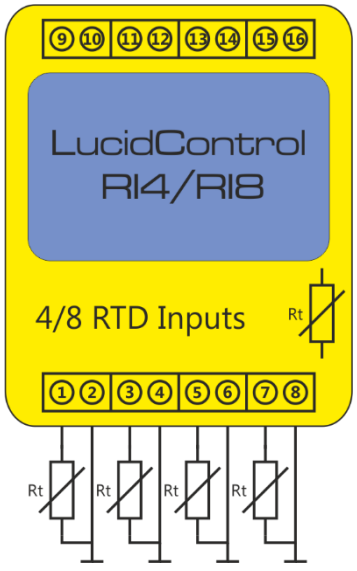

Fig. 2 RI4/RI8 Module Connection

Fig. 2 shows the interconnection of the RI8 module with 4 RTDs connected to the lower IO-Connector.

The terminals IO2, IO4, IO6, IO8 (and also IO10, IO12, IO14, IO16) are internally connected to ground.

The terminals IO9 to IO16 are for RI8 only.

The input channel numbers start with CH0 (IOI1 and IO2) and end with CH3 (IO7 and IO8) for RI4 and CH7 (IO15 and IO16) for RI8.

Only Resistors (RTDs) are allowed to be connected to the IO-Connector of the LucidControl RI4/RI8 module.

2.1.3 Isolation of USB Interface (-ISO option)

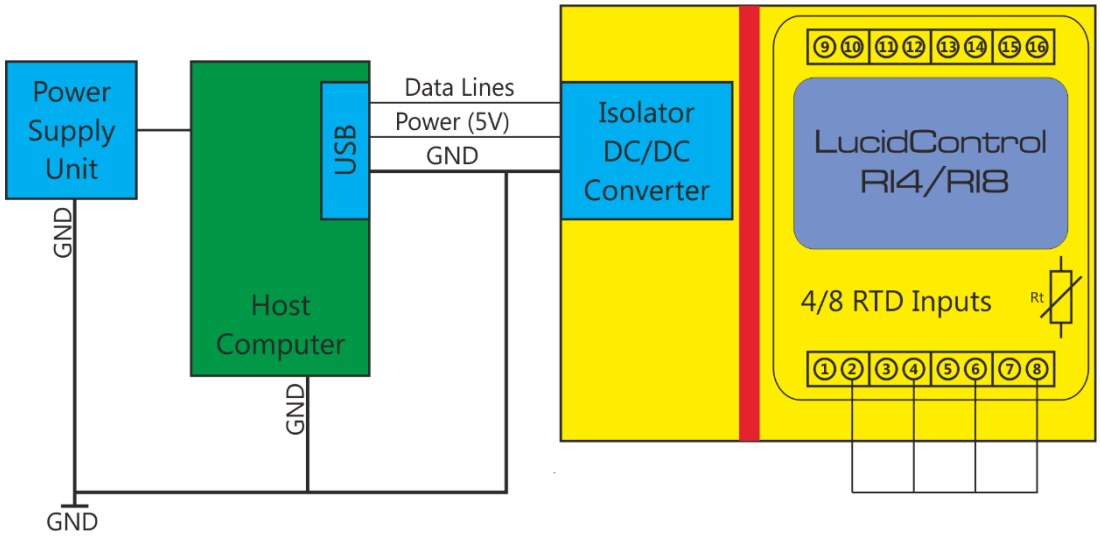

Fig. 3 RI4/RI8 Module with isolated USB Interface

RI4 and RI8 modules are optionally available with isolated USB interface (-ISO option). Fig. 3 shows the RI4/RI8 module with isolated USB interface.

The isolation consists of a galvanic barrier (red area) that isolates the IO module entirely from the USB data lines and power supply lines. An isolated DC/DC converter separates the power supply.

The main purpose of the isolated LucidControl module is the separation of the IO module from the data processing equipment (e.g. the host computer). Non-Isolated IO modules are conductive connected to the USB port also sharing a common ground line.

Harsh or noisy environments (e.g. with disturbances or long cables) may cause measurement errors or malfunction of the data processing equipment or the IO module caused by ground loops. This can be solved by the isolation of the USB port.

Another aspect is the protection of the data processing equipment from overvoltage. If for example a voltage above the limits of the module is applied to the terminals this can damage the module and the data processing equipment also.

The isolation limits the possible damage to the IO module itself.

USB isolation can be an option if a higher protection level required or if LucidControl IO modules are operating in harsh environments.

Even when the isolation protects the data processing equipment from overvoltage damage it does not protect from voltages > 50V!

2.2 Setup of Hard- and Software

Setting up LucidControl hardware is very easy:

1 Ensure that no signal is applied to the IO Connector
2 Connect LucidControl via USB with the computer
3 Applies for Microsoft Windows older than Windows 10 only: The system asks for an installation file. This is not a driver but only an information file (INF). The file can be downloaded from our website www.lucid-control.com/downloads
4 That's all. LucidControl switches the green power LED on and the module is ready for usage.

2.2.1 Windows

As mentioned the installation under Microsoft Windows (older than Windows 10) requires the information file.

After finished installation the Windows Device Manager contains a new serial port (COM). The module can be accessed using this port.

Note
Even if more than one module is connected to a computer Windows ensures that the same serial port number is assigned to the module(s) after restart.

2.2.2 Linux

Despite to Windows installation under Linux the module is usable immediately after connection without any additional steps. Linux installs /dev/ttyACM devices for any module connected to the computer.

Note
By default Linux cannot ensure that the same /dev/ttyACM device is assigned to the same module on restart. But as long as only one module is connected to the computer it is ensured that it is accessible via /dev/ttyACM0.

This problem can be solved by the LucidIoCtrl command line tool which can create static devices always pointing to a specific module. Moreover the device can be given useful names e.g. dev/digitalIoKitchen.

2.2.3 Get command line LucidIoCtrl

LucidIoCtrl command line tool can be downloaded from our website:

www.lucid-control.com/downloads

This page provides the command line tool LucidIoCtrl for different architectures.

After downloading the program can be stored in a folder of choice.

Please see the section 3 of the general LucidControl User Manual for more information about this helpful tool.

2.2.4 Ready to Start

After the module was installed successfully (if it was necessary at all) the green *Power LED is switched* on signaling that the module is ready for use.

Since the module was preconfigured for standard input mode, it can be used without further configuration. The following examples demonstrate the functionality of the module by using the LucidIoCtrl command line tool.

Windows Examples
For all examples it is assumed that the module is connected to COM1.

Reading the values of all 4 input channels

```
      LucidIoCtrl -dCOM1 -tT -c0,1,2,3 -r [ENTER]
->    CH0:25.000  CH1:25.000  CH2:25.000  CH3:25.000
```

Linux Examples
For all examples it is assumed that the module is connected to /dev/ttyACM0.

Reading the values of all 4 input channels

```
      LucidIoCtrl -d/dev/ttyACM0 -tT -c0,1,2,3 -r [ENTER]
->    CH0:25.000  CH1:25.000  CH2:25.000  CH3:25.000
```

Note
Obviously, the temperatures can only be read if RTDs are connected to the clamps. Otherwise, the module returns:

```
->    CH0:ERR_OPEN  CH1:ERR_OPEN  CH2:ERR_OPEN  CH3:ERR_OPEN
```

3 Module Operation

3.1 Channel Processing

The data acquisition of the RI4/RI8 selects the active input channels subsequently, sourcing them with a current for a configurable time T_{Setup} and measuring the current and voltage at the input channel.

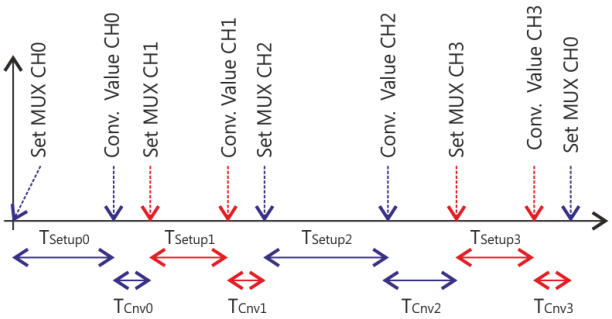

Fig. 4 explains the measurement procedure. The diagram shows the subsequent measurement of the input channels CH0 to CH3.

Fig. 4 Input Processing

At a given time, the algorithm selects one input channel and all others are deselected.

At first, channel CH0 is selected and the multiplexer is set to channel CH0, sourcing the RTD with the measurement current. After the time T_{Setup} has passed the data conversion starts. Depending on the conversion settings the result is ready after the conversion time T_{Conv}. When the result is ready, it is stored and the next channel is selected.

This procedure continues for channels CH1, CH2 and CH3, and for the RI8 module also for the channels CH4, CH5, CH6 and CH7. After the last channel was processed, the first channel is selected again and the conversion loop continues from beginning.

3.1.1 Measurement Timing

The procedure explained in Fig. 4 takes some time until it completes and the new result is ready USB.

The acquisition time of the measurement can be changed for each channel by the parameters *inRtSetupTime* (see 3.6.5) and *inRtNrSamples* (see 3.6.4).

The parameter inRtSetupTime specifies the time T_{Setup}. After a channel is selected the conversion will be started after T_{Setup} has passed.

The parameter *inRtNrSamples* specifies the number of oversampling cycles. Oversampling can give additional accuracy by sampling the analog values multiple times. In theory, this can also give additional resolution.

Channels that are not used can be set can be set to Inactive Mode (see 3.2.1 and 3.6.2). Inactive channels are skipped and not processed, improving the timing of active channels.

Typical Measurement Timing Examples

Nr. of active Channels	Nr. of Samples	T_{Setup} [ms]	T_{Cycle} [ms]	Remarks
4	16	25	160	Default for RI4 module
4	16	10	100	
4	16	5	80	
2	16	25	80	
2	16	5	40	
4	4	25	116	
4	4	5	36	
8	16	25	330	Default for RI8 module
8	16	5	170	
8	4	5	72	

The table above shows the time T_{Cycle} for some parameter settings of *inRtNrSamples* and *inRtSetupTime*. T_{Cycle} is the acquisition time interval and after T_{Cycle} all channels have been updated.

Using the default settings, the value of the channels is updated every 160ms for the RI4 module and every 330ms for the RI8 module.

Note

There are many combinations of *inRtSetupTime* and *inRtNrSamples* possible but we recommend only using the settings from the table above.

3.2 Operation Modes

3.2.1 Inactive Mode

In Inactive Mode the RTD measurement is disabled and the channel is skipped.

3.2.2 Standard Mode

In Standard Mode the RTD the inputs are measured as configured.

3.3 Offset Compensation

In order to compensate the offset of an input channel the Parameter *inRtOffset* can be used. (see 3.6.6)

This parameter allows setting up an offset resistance in 0.1 Ω steps for Pt1000 sensors and 0.01 Ω steps for Pt100 sensors. The value in Ω is added to the measured resistor value. Since *inRtOffset* can also be a negative value, compensation in both directions to higher or lower values is possible.

For the RI4/RI8 module a 0 offset compensation can be made by connecting a load resistor.

Measured Resistance: $R = R_M + R_{Offset}$

Since the temperature is approx. proportional $T[℃] \sim \frac{R}{10} * 0.256 \left[\frac{℃}{\Omega}\right]$ the value R_M changes the resulting temperature of a Pt1000 sensor by approx. $0.0256 \left[\frac{℃}{Digit}\right]$.

Example
When connecting a 1000 Ω resistor to the input channel 0 the resulting temperature should be exactly 0 °C. In this example a value of +0.5 °C is measured caused by outer influences e.g. the RTD itself. This offset value is compensated in the following procedure.

With the equations above it can be calculated that the deviation of +0.5 °C results in an offset correction value of -19.53 (rounded -20, which means -2.0 Ω)

The following function call adjusts the measured value to the load of 1000 Ω representing a value close to 0°C and stores the setting:

```
LucidIoCtrl –dCOM1 –c0 –sinRtOffset=-20 –p [ENTER]
```

3.4 Line Status Detection

The inputs are able to detect a broken or short cut sensor line.

In case that the measurement result is lower than a limit the module disregards the measured value and return a specified “line short cut” (ERR_SHORT) value.

In the case that the measurement result is higher than a limit the module disregards the measured value and return a specified “line broken” (ERR_OPEN) value.

Defined values for line status detection:

Value Type	Condition	Returned value
TMS4	$T < T_{NegLimit}$ (ERR_SHORT)	0x80000000
	$T > T_{PosLimit}$ (ERR_OPEN)	0x7FFFFFFF
TMS2	$T < T_{NegLimit}$ (ERR_SHORT)	0x8000
	$T > T_{PosLimit}$ (ERR_OPEN)	0x7FFF

3.5 Commands

Accessing inputs and outputs is a very common function that is mostly identical for all LucidControl modules. Input modules provide the commands GetIo for reading of a single value and GetIoGroup for reading of a group of values of the same type.

For more comprehensive information covering reading and writing of inputs and outputs please see the sections 3.2.1.1, 3.2.1.2 and 4.3 of the general LucidControl manual.

The following sections describe in detail the commands supported by the RI4/RI8 module.

3.5.1 GetIo

This command reads values of the RTD input channel.

<table>
<tr><td>Command</td><td>GetIo</td><td>Access</td><td>Read</td></tr>
<tr><td>Opcode</td><td>0x46</td><td></td><td></td></tr>
<tr><td colspan="4">LucidIoControl Command Line Tool</td></tr>
<tr><td>Call (-tT)</td><td colspan="3"><code>LucidIoCtrl -d[COMx] -c[Channel] -tT -r</code></td></tr>
<tr><td>Return</td><td colspan="3"><code>CHn:tt</code><table><tr><td>n</td><td>Input Channel</td></tr><tr><td>tt</td><td>Temperature in °C</td></tr></table></td></tr>
<tr><td>Call (-tR)</td><td colspan="3"><code>LucidIoCtrl -d[COMx] -c[Channel] -tR -r</code></td></tr>
<tr><td>Return</td><td colspan="3"><code>CHn:rr</code><table><tr><td>n</td><td>Input Channel</td></tr><tr><td>rr</td><td>Resistance in Ω</td></tr></table></td></tr>
</table>

Note
When using the LucidIoCtrl command line tool the distinction between GetIo and GetIoGroup commands is not necessary since the program handles this automatically.

LucidIoCtrl Command Line Tool Example

Read temperature from input channel 0.

```
      LucidIoCtrl –dCOM4 –c0 –tT -r [ENTER]
->    CH0:100.200
```

Read the corresponding resistance of the same input.

```
      LucidIoCtrl –dCOM4 –c0 –tR -r [ENTER]
->    CH0:1385.8
```

Line Status Detection

The module is able to detect RTD shortcut and wire breaks in the connection cable (See 3.4). In the case of a wire break is detected the command returns `ERR_OPEN`. In the case that a shortcut is detected the command returns `ERR_SHORT`.

Request Frame

OPC	P1	P2	LEN
0x46	Channel	Value Type	0

Value	Description
Channel	Number of input or output channel (Range: 0 ~ 7)
Value Type	Supported Value Types

Value Type	Value Range	Size
Temperature Resolution : 1/100 °C (0x41)	-20,000 ~ 20,000 -200.00 °C ~ 200.00°C	4 Bytes
Temperature Resolution: 1/10 °C (0x40)	-2,000 ~ 2,000 -200.0 ~ 200.0 °C	2 Bytes
Resistance Resolution: 1/10 Ω (0x50)	0 ~ 65,535 0 ~ 6,553.5 Ω	2 Bytes
Resistance Resolution: 1/1000 Ω (0x51)	0 ~ 2^{32} -1 mΩ	4 Bytes

Tab. 1 GetIo Request

Response

Status	LEN	Data
Status	Length	Value

In case of successful execution, the command returns the value of the specified channel number.

In the case of an error, the command returns Execution Status Code documented in section 4.4 of the LucidControl User Manual.

3.5.2 GetIoGroup

This command reads the input values of a group of inputs of the same Value Type. See also section 3.5.1.

<table>
<tr><td>Command</td><td>GetIoGroup</td><td>Access</td><td>Read</td></tr>
<tr><td>Opcode</td><td>0x48</td><td></td><td></td></tr>
<tr><td colspan="4">LucidIoControl Command Line Tool</td></tr>
<tr><td>Call (-tT)</td><td colspan="3"><code>LucidIoCtrl –d[COMx] –c[Channels] –tT –r</code>

<u>Channels:</u>
Comma separated list of channels e.g. –c0,1,3</td></tr>
<tr><td>Return</td><td colspan="3">List of values sorted from lower to higher channels
<code>CHn:tt</code><table><tr><td>n</td><td>Input Channel</td></tr><tr><td>tt</td><td>Temperature in °C</td></tr></table></td></tr>
<tr><td>Call (-tR)</td><td colspan="3"><code>LucidIoCtrl –d[COMx] –c[Channels] –tR –r</code>

<u>Channels:</u>
Comma separated list of channels e.g. –c0,1,3</td></tr>
<tr><td>Return</td><td colspan="3"><code>CHn:rr</code><table><tr><td>n</td><td>Input Channel</td></tr><tr><td>rr</td><td>Resistance in Ω</td></tr></table></td></tr>
</table>

<u>LucidIoCtrl Command Line Tool Example</u>

Read temperatures from all input channels 0, 1, 2 and 7.

```
      LucidIoCtrl –dCOM4 –c0,1,2,7 –tT –r [ENTER]
->    CH0:100.000  CH1:0.500  CH2:-100.300  CH7:78.250
```

Read temperatures form input channels 0, 1, 2 and 7.

```
      LucidIoCtrl –dCOM4 –c0,1,2,7 –tT –r [ENTER]
->    CH0:100.000  CH1:0.500  CH2:ERR_SHORT  CH7:ERR_OPEN
```

In this case the RTD connected to input channel 2 is shortcut and no RTD is connected to input channel 7.

<u>Line Status Detection</u>

The module is able to detect RTD shortcut and wire breaks in the connection cable (See 3.4). In the case of a wire break is detected the command returns `ERR_OPEN`. In the case that a shortcut is detected the command returns `ERR_SHORT`.

Request Frame

OPC	P1	P2	LEN
0x48	Ch-Mask	Value Type	0

<table>
<tr><th>Value</th><th>Description</th></tr>
<tr><td>Channel Mask</td><td>Channel Bit Mask specifying the channel number(s)
<table>
<tr><th>Channel</th><th>Bit Position</th><th>Value</th></tr>
<tr><td>0</td><td>0</td><td>0x01</td></tr>
<tr><td>1</td><td>1</td><td>0x02</td></tr>
<tr><td>2</td><td>2</td><td>0x04</td></tr>
<tr><td>3</td><td>3</td><td>0x08</td></tr>
<tr><td>4</td><td>4</td><td>0x10</td></tr>
<tr><td>5</td><td>5</td><td>0x20</td></tr>
<tr><td>6</td><td>6</td><td>0x40</td></tr>
<tr><td>7</td><td>P1A 0</td><td>P1=0x80
P1A = 0x01</td></tr>
</table>
Values can be bitwise combined.

Size of P1 is 1 or 2 bytes. If Bit 7 of P1 is set, a subsequent P1A is expected.

Examples
Accessing channel numbers:
0 and 3 Value P1 = 0x01 OR 0x08 = 0x09
1 and 2 Value P1 = 0x02 OR 0x04 = 0x06
1, 2 and 7 Value P1 = 0x02 OR 0x04 = 0x86
Value P1A = 0x01 (for channel 7)</td></tr>
<tr><td>Value Type</td><td>Supported Value Types
<table>
<tr><th>Value Type</th><th>Value Range</th><th>Size</th></tr>
<tr><td>Temperature
Resolution : 1/100 °C
(0x41)</td><td>-20,000 ~ 20,000
-200,00 °C ~
200,00°C</td><td>4 Bytes</td></tr>
<tr><td>Temperature
Resolution: 1/10 °C
(0x40)</td><td>-2,000 ~ 2000
-200,0 ~ 200,0 °C</td><td>2 Bytes</td></tr>
<tr><td>Resistance
Resolution: 1/10 Ω
(0x50)</td><td>0 ~ 65535
0 ~ 6553,5 Ω</td><td>2 Bytes</td></tr>
<tr><td>Resistance
Resolution: 1/1000 Ω
(0x51)</td><td>0 ~ 2^{32} -1 mΩ</td><td>4 Bytes</td></tr>
</table></td></tr>
</table>

Tab. 2 GetIoGroup Request

Response Frame

Status	LEN	Data Field
Status	Length	Value(s)

In case of successful execution, the command returns the read values of the channels specified in the Channel Mask.

In the case of an error, the command returns Execution Status Code documented in section *4.4 of the* LucidControl User Manual.

Example of GetIoGroup Request

The following request frame reads temperatures from input channels 0 and 1

Opcode	P1	P2	Length
0x48	0x03	0x41	0x00

Channel Mask (P1): 0x01 OR 0x02 = 0x03

Response Frame

For input 0 = 50°C , input 1 = -25°C

Values in Data Field are in ascending order Channel 0, Channel 1.

Header Field		Data Field							
Status	LEN	Value Channel 0				Value Channel 1			
0x00	0x08	0x88	0x13	0x00	0x00	0x3C	0xF6	0xFF	0xFF

3.6 Parameters

LucidControl modules allow configuration by a set of System Configuration Parameters and IO Configuration Parameters.

The Parameters are accessible by the commands SetParam and GetParam. The sections 4.3.5 and 4.3.6 of the LucidControl User Manual describe them in detail.

3.6.1 inRtValue

This IO Configuration Parameter contains the measured resistance value with a resolution of 0.1 Ω.

Parameter	inRtValue	**Access**	Read
Address	0x1000		
Values	Measured Resistance in 0.1 Ω		
Default Value	0x00	**Parameter Type**	2 Bytes unsigned
LucidIoControl Command Line Tool			
Parameter Name	inRtValue	**Parameter Values**	Resistor [1/10 Ω] 0 ~ 6553,5 Ω
Call (Get)	`LucidIoCtrl –d[COMx] –c[Channel] –ginRtValue`		

LucidIoCtrl Command Line Tool Example

Read value parameter of input channel 0.

```
LucidIoCtrl –dCOM4 –c0 –ginRtValue [ENTER]
-> inRtValue=100
```

The measured value of 10.0 Ω is returned.

Note

For normal operation is recommended to use the function GetIo (see 3.5.1) in order to read the input value. The parameter provides the resistance value only.

3.6.2 inRtMode

This IO Configuration Parameter configures the operation mode of the input.

Parameter	inRtMode	**Access**	Read / Write
Address	0x1100		
Values	Input Mode **Byte** 0x00 = inactive 0x01 = standard		
Default Value	standard	**Parameter Type**	1 Byte unsigned
LucidIoControl Command Line Tool			
Parameter Name	inRtMode	**Parameter Values**	inactive / standard
Call (Set)	`LucidIoCtrl –d[COMx] –c[Channel] –sinRtMode=[Mode] {-p} {--default}`		
Call (Get)	`LucidIoCtrl –d[COMx] –c[Channel] –ginRtMode`		

LucidIoCtrl Command Line Tool Example

Set operation mode of input channel 0 to standard mode and make the setting persistent.

```
LucidIoCtrl –dCOM4 –c0 –sinRtMode=standard –p [ENTER]
```

Read the operation mode of input channel 0.

```
LucidIoCtrl –dCOM4 –c0 –ginRtMode [ENTER]
->inRtMode=standard
```

3.6.3 Bit Parameter inRtFlags

This IO Configuration Parameter groups Bit Parameters which are represented by one bit e.g. *having an "on"* or "off" state only).

Parameter	inRtFlags	**Access**	Read / Write
Address	0x1101		
Values	Consists of the following Bit Parameters **Bit Parameter** / **Bit Postion**		
Default Value	0x00	**Parameter Type**	1 Byte unsigned

Note

The parameter *inRtFlags* cannot be accessed directly by using the Command Line Tool. The Bit Parameters can be used instead.

Note

When *inRtFlags* is changed by the SetParam command which is described in section 4.3.5 of the LucidControl User Manual the current setting of *inRtFlags* must be read before updating it in order to prevent overwriting other Bit Parameters.

3.6.4 inRtNrSamples

This IO Configuration Parameter configures the number of oversampling cycles. See also section 3.1.1.

Parameter	inRtNrSamples	**Access**	Read / Write
Address	0x1113		
Values	1, 2, 4, 8, 16, 32, 64, 128, 256 oversampling cycles		
Default Value	16	**Parameter Type**	2 Bytes unsigned
LucidIoControl Command Line Tool			
Parameter Name	inRtNrSamples	**Parameter Values**	Cycles
Call (Set)	`LucidIoCtrl –d[COMx] –c[Channel] –sinRtNrSamples=[cycles] {-p} {--default}`		
Call (Get)	`LucidIoCtrl –d[COMx] –c[Channel] –ginRtNrCycles`		

LucidIoCtrl Command Line Tool Example

Set number or oversampling cycles for channel 0 to 8 and make the setting persistent.

```
LucidIoCtrl –dCOM4 –c0 –sinRtNrSamples=8 –p [ENTER]
```

Read number of oversampling cycles of channel 0

```
LucidIoCtrl –dCOM4 –c0 –ginRtNrSamples [ENTER]
-> inRtNrSamples=8
```

3.6.5 inRtSetupTime

This IO Configuration Parameter configures the input channel setup time T_{Setup}. See also section 3.1.

Parameter	inRtSetupTime	**Access**	Read / Write
Address	0x1112		
Values	T_{Scan} in ms (milli seconds) 5 ms ≤ T_{Scan} ≤ 1 s		
Default Value	25 (25 ms)	**Parameter Type**	2 Bytes unsigned
LucidIoControl Command Line Tool			
Parameter Name	inRtSetupTime	**Parameter Values**	Time [ms]
Call (Set)	`LucidIoCtrl –d[COMx] –c[Channel] –sinRtSetupTime=[Time] {-p} {--default}`		
Call (Get)	`LucidIoCtrl –d[COMx] –c[Channel] –ginRtSetupTime`		

LucidIoCtrl Command Line Tool Example

Set T_{Setup} of input channel 0 to 25 ms and make the setting persistent.

```
LucidIoCtrl –dCOM4 –c0 –sinRtSetupTime=25 –p [ENTER]
```

Read T_{Scan} parameter of input channel 0

```
LucidIoCtrl –dCOM4 –c0 –ginRtSetupTime [ENTER]
-> inRtSetupTime=25
```

3.6.6 inRtOffset

This IO Configuration Parameter configures the Input Offset Compensation value which is described in section 3.3.

Parameter	inRtOffset	**Access**	Read / Write
Address	0x1120		
Values	Pt1000: Offset Compensation in 0.1 Ω steps (-1,000 Ω ~ 1,000 Ω) -10,000 ~ 10,000 Pt100: Offset Compensation in 0.01 Ω steps (-100 Ω ~ 100 Ω) -10,000 ~ 10,000		
Default Value	0	**Parameter Type**	2 Bytes signed
LucidIoControl Command Line Tool			
Parameter Name	inRtOffset	**Parameter Values**	Resistance [0.1 Ω]
Call (Set)	`LucidIoCtrl –d[COMx] –c[Channel] –sinRtOffset=[Offset] {-p} {--default}`		
Call (Get)	`LucidIoCtrl –d[COMx] –c[Channel] –ginRtOffset`		

LucidIoCtrl Command Line Tool Example

Set Input Offset Compensation value of the Rt1000 input channel 0 to -2 Ω and make the setting persistent.

```
LucidIoCtrl –dCOM4 –c0 –sinRtOffset=-20 –p [ENTER]
```

Read Input Offset Compensation value

```
LucidIoCtrl –dCOM4 –c0 –ginRtOffset [ENTER]
-> inRtOffset=20
```

4 Specification

Parameter		Condition	Value	
Inputs				
	No of Input Channels		4/8	
Input - Electrical Characteristics				
	Measurement Method		RTD Two wire measurement	
	Resolution		typ. 0.1°C	
	RTD Type		Pt100 / Pt1000 DIN IEC 751	
	Measurement Error	Pt100	typ. +/- 1.0°C	
		Pt1000	typ. +/- 0.5°C	
	Constant Measurement	Pt100	1mA	
	Current	Pt1000	0.5 mA	
Input – Timing Characteristic				
	Setup Time		T_{Setup}	5 ms ≤ t ≤ 1 s
Module – Communication				
	USB		2.0 Full Speed CDC Profile	
Module – Electrical Characteristics				
	Power Supply		USB Bus Powered with +5V No additional Power Supply needed.	
	Maximum Rated Supply Current			40 mA
Module – Environment				
	Temperature	Storage	-20 °C ~ +70 °C	
		Operation	0 °C ~ +55 °C	
	Humidity		< 85 % RH, non-condensing	
Module – Housing				
	Dimension L x W x H		90 x 54 x 62 mm	
	Weight (in total)		120 g	
	Assembly		Rail-Mount (EN 50022, TS35)	
	Protection Class (DIN 40050)		IP20	
Module - Indicators				
	• Operation and Error Indicator • Communication Indicator			

5 Order Information and Accessories

RTD Temperature Measurement Devices

General type number format

LCTR-RIn-RTDType-Range(-ISO)

Order Code	Product
LCTR-RI4-1000	LucidControl RTD Input USB Module with 4 Channels for Pt1000 Sensors Measurement Range (-180°C ~ +180°C)
LCTR-RI4-100	LucidControl RTD Input USB Module with 4 Channels for Pt100 Sensors Measurement Range (-180°C ~ +180°C)
LCTR-RI8-1000	LucidControl RTD Input USB Module with 8 Channels for Pt1000 Sensors Measurement Range (-180°C ~ +180°C)
LCTR-RI8-100	LucidControl RTD Input USB Module with 8 Channels for Pt100 Sensors Measurement Range (-180°C ~ +180°C)

Order Code (Range)	Product
-C0C360	Measurement Range (0°C ~ +360°C)

Order Code (ISO)	Product
-ISO	With galvanic isolation of USB Interface

The following accessories are available:

Order Code	Product
64.200.0005	Plug-In Terminal 8-way 1,5 mm^2 wire

6 Document Revision

Date	Rev.	
2018/02/20	2.0	• Added documentation of RI8 module • Added documentation of USB Isolation