



User Manual

# Lucid485 DI4/DI8

4/8 Channel Digital Input Serial Module

## Table of Content

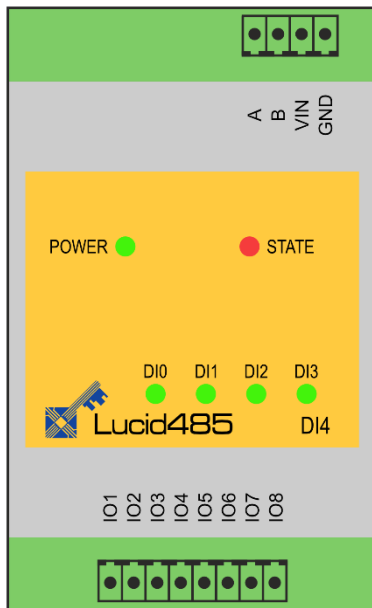
1	Introduction.....	3
2	Setup and Installation.....	3
2.1	Safety Information.....	3
2.2	RS-485 bus and Power Connection.....	4
2.3	LucidloCtrl Command Line Tool.....	4
2.3.1	First Steps.....	4
2.4	IO Configurations.....	5
3	Module Operation.....	8
3.1	Real-time Considerations.....	8
3.1.1	Input Signal Value Inversion.....	8
3.1.2	Filter.....	8
3.2	Operation Modes.....	8
3.2.1	Reflect Mode.....	9
3.2.2	Edge Detection.....	10
3.2.2.1	Rising Edge Detection Mode.....	11
3.2.2.2	Falling Edge Detection Mode.....	11
3.2.3	Count Mode.....	12
3.2.3.1	Count Mode Options.....	13
3.3	Commands.....	15
3.3.1	Getlo.....	15
3.3.2	GetloGroup.....	17
3.4	Parameters.....	20
3.4.1	inDiValue.....	20
3.4.2	inDiMode.....	20
3.4.3	Bit Parameter inDiFlags.....	22
3.4.3.1	inDiInverted.....	22
3.4.3.2	inDiAddCounter.....	23
3.4.3.3	inDiResetCounterOnRead.....	23
3.4.4	inDiScanTime.....	24
3.4.5	inDiCountTime.....	24
4	Modbus RTU Protocol.....	26
5	Specification.....	27
6	Order Information.....	29
7	Document Revision.....	29

## 1 Introduction

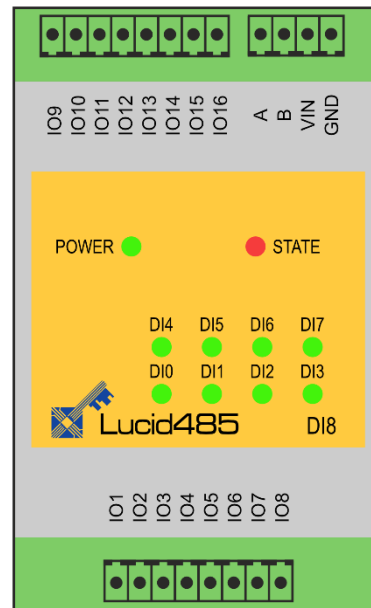
This document describes the functionality of the Lucid485 DI4/DI8 serial IO module providing 4/8 digital inputs controllable by RS-485 bus.

A basic description of the Lucid485 product family can be found in the general *Lucid485 User Manual*.

## 2 Setup and Installation



**Fig. 1 Lucid485 DI4 Module**



**Fig. 2 Lucid485 DI8 Module**

Fig. 1 and Fig. 2 show drawings of the DI4 and DI8 digital input modules with 4/8 digital input channels (DI0 – DI3 and DI4 to DI7).

The IO signals are connected to the lower (IO1 - IO8) and the upper (IO9 - IO16) IO terminals.

The upper IO terminal connector is present at the DI8 module only.

### 2.1 Safety Information

Lucid485 complies with regulations and industrial standards active in the EU. To keep the device functional, the following safety and maintenance information must be adhered.

The device must only be used for the intended purpose.

The device must not be used under the following conditions:

- It is obviously damaged

- An error was detected
- Outside humidity and temperature limits
- Unauthorized personnel



Never apply voltages higher than 30V (or lower than -30V) to any IO terminal. This would damage the device and may also destroy the host device.



All contacts of the modules are protected against ESD but not necessarily against overload, which is especially relevant for output modules.

## 2.2 RS-485 bus and Power Connection

Please see the general *Lucid485 User Manual*.

## 2.3 LucidIoCtrl Command Line Tool

The LucidIoCtrl command line tool gives full access to all Lucid485 modules operating with Frame Protocol enabled. Executables for different architectures and can be downloaded from our website:

<https://www.lucid-control.com/downloads>

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

Please see the general *Lucid485 User Manual* for more information.

### 2.3.1 First Steps

When the module is powered, the green power LED is switched on, signaling that the module is ready.

The following examples demonstrate the functionality of the module by using the LucidIoCtrl command line tool.

It is assumed that the module is configured with default address 11, 9600 baud, 8N1.

The serial to RS-485 adapter is connected to COM1 of the host.

#### Windows Examples

Reading the values of the first 4 input channels from a device.

```
LucidIoCtrl -drs485:COM1:11 -tL -c0,1,2,3 -r  
-> CH0:00 CH1:00 CH2:00 CH3:00
```

Windows requires a different argument for comport numbers 10 and above.

```
LucidIoCtrl -drs485:\\.\COM10:11 -tL -c0,1,2,3 -r  
-> CH0:00 CH1:00 CH2:00 CH3:00
```

### Linux Examples:

Reading the values of the first 4 input channels from a device connected to /dev/ttyACM0.

```
LucidIoCtrl -drs485:/dev/ttyACM0:11 -tL -c0,1,2,3 -r
-> CH0:00 CH1:00 CH2:00 CH3:00
```

## 2.4 IO Configurations

The DI4/DI8 module is available in different configurations which are described in this section.

Function Class	Value	Channels
DI4	0x8000	4
DI8	0x8010	8

Tab. 1 Digital Input Function Classes

Function Class Type	Value	Threshold Level
5	0x1000	5V
10	0x1001	10V
24	0x1005	24V

Tab. 2 Digital Input Function Class Types

Tab. 1 and Tab. 2 list the Function Classes and their types.

The digital inputs are available with different threshold levels:

Function Class Type	V <sub>LowMax</sub>	V <sub>HighMin</sub>	V <sub>HighMax</sub>
5	2.5 V	3.5 V	7.5V
10	6.0 V	8.5V	15V
24	16.0 V	21.0 V	30V

Tab. 3 Digital Input Threshold Levels

Tab. 3 shows the characteristic voltages for the different threshold levels.

Voltages below V<sub>LowMax</sub> result in a LOW value, voltage higher than V<sub>HighMin</sub> in a HIGH value. Because of the input hysteresis, voltages between V<sub>LowMax</sub> and V<sub>HighMin</sub> do not change the logic value.

In order to prevent excessive stress on the digital inputs, the maximum applied voltage must not exceed V<sub>HighMax</sub>.

A high input state is indicated by a green status LED.

### Example

When interfacing a 24 V signal, the applied voltage of a HIGH state must be higher than 21.0 V. The voltage of a low state needs to be lower than 16.0 V. For the voltages between, the last detected stable value remains.



Fig. 3 DI8 IO Connection

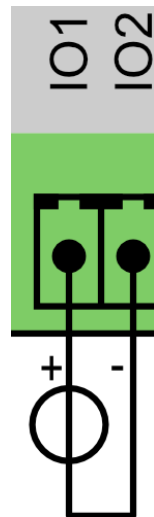


Fig. 4 DI8 Signal

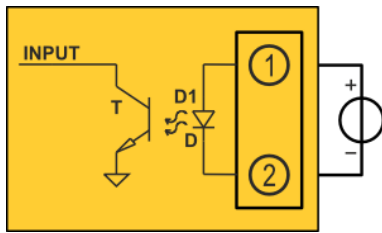
Fig. 3 shows the connection of the DI8 module.

Voltage sources are connected to all IO terminals IO1 to IO16.

Fig. 4 shows in detail the voltage source connected to IO terminals IO1 and IO2 (DI0).

IO Terminal	Signal	IO Channel Number
1	DI0 +	0
2	DI0 -	
3	DI1 +	1
4	DI1 -	
5	DI2 +	2
6	DI2 -	
7	DI3 +	3
8	DI3 -	
9	DI4 +	4
10	DI4 -	
11	DI5 +	5
12	DI5 -	
13	DI6 +	6
14	DI6 -	
15	DI7 +	7
16	DI7 -	

Tab. 4 DI4 / DI8 IO Terminal Connector



**Fig. 5 Digital Input Principle**

Fig. 5 illustrates the principle of the digital input channels. The input signal applied to the IO terminals 1 and 2 (DI0) powers the LED of the opto-coupler, which insulates the input signal and the acquisition hardware.

All inputs are floating and do not have any common connections e.g. to ground signal.

## 3 Module Operation

### 3.1 Real-time Considerations

Operating systems for personal computers are not made for deterministic real-time operation. Because of multitasking it cannot be ensured that a task will continue to run within a specified interval.

Assuming that short pulses (e.g. shorter than 10 ms) should be detected, the computer has to read the input value more than 100 times per second what is not realistic. A pulse located between two readings and the pulse would be lost.

Edge detection and count modes improve the real-time characteristics.

#### 3.1.1 Input Signal Value Inversion

Digital input channels consist of input signal value and a logical input value. The input signal value is represented by the voltage applied to the input channel. The logical value is evaluated by the input processing.

In case of inversion is disabled, the input signal values and logical values are identical.

In case of input inversion is enabled by configuration parameter *inDiInverted* ( $\rightarrow 0$ ) set to "on", the logical value is the inverted input signal value. This means that a voltage higher than  $V_{HighMin}$  results in a HIGH input signal value but a LOW logical input value.

All input modes support input signal value inversion.

#### 3.1.2 Filter

The digital input signal is filtered and only stable high or low signals are evaluated. This function allows the input channel to debounce the signal.

Signals are only considered valid if they are stable during the time  $T_{Scan}$  ( $\rightarrow$  3.4.4  $\rightarrow$  Fig. 7).

### 3.2 Operation Modes

This section explains the operation modes of the input channels and gives examples how to configure and to use them.

Digital input channels can operate in one of the modes:

- Reflect Mode
- Rising Edge Mode
- Falling Edge Mode



- Count Mode

In all modes the input values are captured and evaluated after a stable signal has been detected.

### 3.2.1 Reflect Mode

The reflect mode acquires the logical input value.

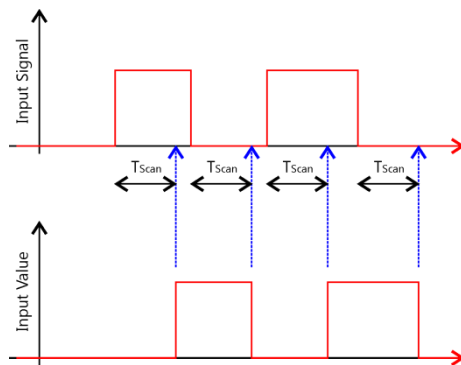
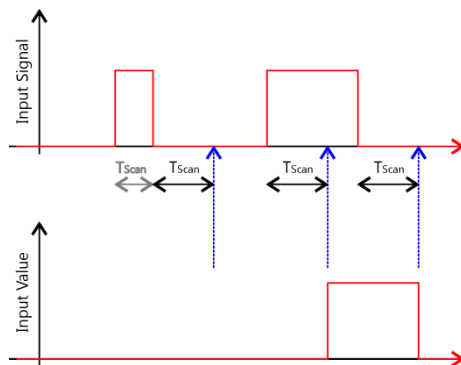


Fig. 6 illustrates the processing of the digital input channels in reflect mode.

After the rising edge of the input signal was detected and it remained stable for the interval  $T_{Scan}$  the input value is updated.

**Fig. 6 Digital Input Channel Processing in Reflect Mode**



In the case that a pulse of the input signal is shorter than  $T_{Scan}$ , it is ignored and the input value does not change. Fig. 7 shows this at the first pulse. The rising edge of the input signal starts the scan timer and the falling edge stops it (indicated by the gray  $T_{Scan}$  interval).

The second pulse is longer than  $T_{Scan}$  and considered valid. The input value is updated.

**Fig. 7 Reflect Mode Pulse Width**

Filtering digital signals can be used in order to suppress instable signals (debouncing). It makes the detection of digital input signals more reliable.

The scan interval  $T_{Scan}$  is configured by the parameter *inDiScanTime* (→ 3.4.4).

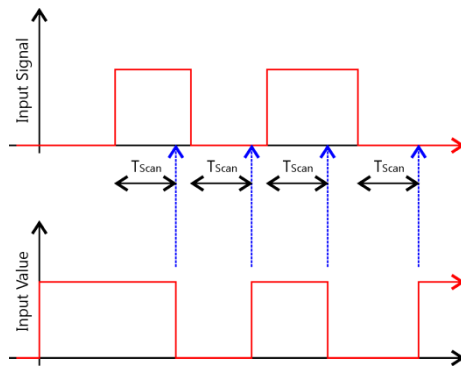


Fig. 8 illustrates the digital input signal and the inverted input value with parameter *inDiInverted* set to "on".

**Fig. 8 Inverted Reflect Mode**

### LucidIoCtrl Command Line Tool Example

Set digital input channel 0 to Reflect Mode

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiMode=reflect
```

## 3.2.2 Edge Detection

Digital input channels can operate in edge detection modes. In rising edge mode, the channel is sensitive for low-to-high transitions, in falling edge mode it recognizes high-to-low transitions.

The length of the shortest detectable pulse is configured by the scan interval time  $T_{Scan}$ . (Parameter *inDiScanTime* → 3.4.4)

### Example

The input should be able to detect an input signal pulse of 100  $\mu$ s length.

In theory it would be sufficient to read the digital input in reflect mode every 50  $\mu$ s (with safety time) in order to detect the transition by the computer. But probably the operating system of the computer would violate the timing restrictions and pulses are lost.

By using the input configured in one of the two edge detection modes this can be solved easily by setting  $T_{Scan} = 90 \mu$ s (including safety). A detected edge will be pending until it is read by the host computer. Assuming the host computer reads now every second it can check if one or more transition(s) occurred within the last second. The 1 second interval can be realized by the host computer without problems.

### LucidIoCtrl Command Line Tool Example

Configure input channel 0 for rising edge detection mode

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiMode=risingEdge
```

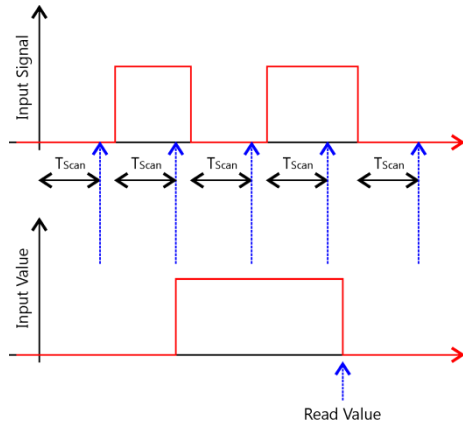
Set  $T_{Scan}$  to 90  $\mu$ s

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiScanTime=90000
```

## Read input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -tL -r
-> CH0:01
```

### 3.2.2.1 Rising Edge Detection Mode

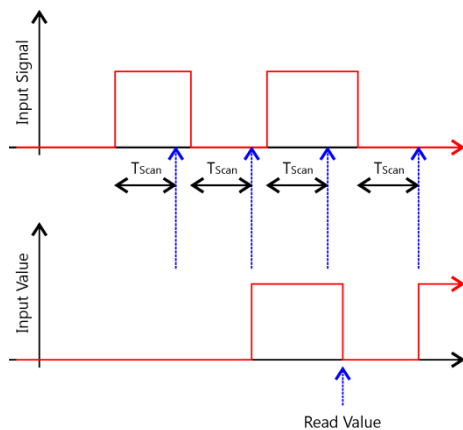


**Fig. 9 Rising Edge Detection Mode**

Fig. 9 shows a digital input signal and the corresponding input value in rising edge detection mode. After the HIGH input signal was detected as being valid, the input value remains pending until it was read by the host computer.

The rising edge detection mode allows detecting low-to-high transitions of the input signal without the host computer being involved.

### 3.2.2.2 Falling Edge Detection Mode



**Fig. 10 Falling Edge Detection Mode**

The falling edge detection mode (Fig. 10) operates similar to the rising edge detection mode (→ 3.2.2.1) but triggers at high-to-low transition of the input signal.

In falling edge detection mode, a high-to-low transition of the input signal sets the input value to "1".

### 3.2.3 Count Mode

The count mode accumulates valid pulses of the input signal within a count interval specified by  $T_{\text{Count}}$ .

The counter is 16 bit wide. When 65535 is reached it rolls over to 0 with the next increment.

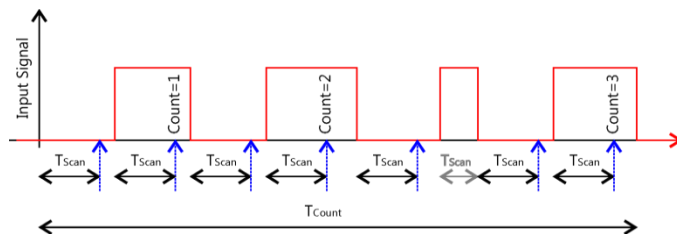


Fig. 11 Count Mode

Fig. 11 illustrates a typical periodical input signal. In count mode all valid pulses are accumulated until the count interval  $T_{\text{Count}}$  finishes.

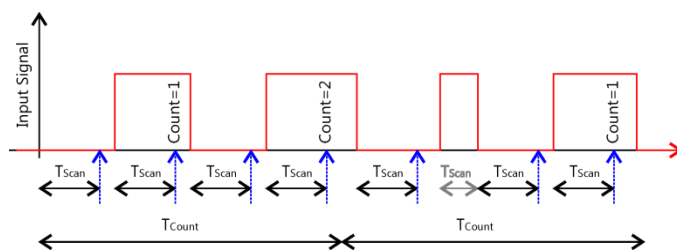


Fig. 12 Count Mode Interval

Fig. 12 shows the same input signal, but with a shorter count interval. After count interval  $T_{\text{Count}}$  has passed, the count value is accessible and a new measurement cycle starts.

The input signal is considered as valid after it was stable for at least the scan time  $T_{\text{Scan}}$ . Only valid pulses are accumulated (see the gray interrupted pulse).

$T_{\text{Scan}}$  time starts when the rising edge of the input signal was detected. In the case that the parameter *inDiInverted* is set to "on",  $T_{\text{Scan}}$  time starts when the falling edge of the input signal is detected.

Reading the input value in count mode returns the count value of the last finished count interval cycle.

#### LucidIoCtrl Command Line Tool Example

##### Configure input channel 0 for Count Mode

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiMode=count
```

##### Set Scan Time $T_{\text{Scan}}$ to 1ms

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiScanTime=1000
```

##### Set Count Time $T_{\text{Count}}$ to 1s

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiCountTime=1000000
```

Read count value (number of pulses)

```
LucidIoCtrl -drs485:COM4:11 -c0 -tN -r
-> CH0:0x0064 (100)
```

In this example 100 pulses of at least 1 millisecond length have been acquired within a count interval of 1 second.

Since the input value is updated after count interval has passed it takes 1 second to update the value. Decreasing the count interval results in a faster update of the input value.

In Count Mode the value type N is supported. The count value is returned in hexadecimal and decimal format.

### 3.2.3.1 Count Mode Options

The functionality of the count mode can be controlled by the parameters *inDiAddCounts* and *inDiResetCountsOnRead*.

The number counted pulses becomes valid after count interval  $T_{\text{Count}}$  has passed:

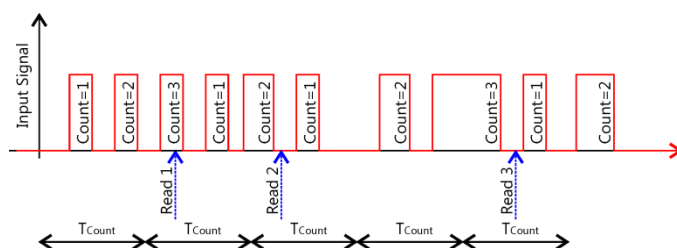
- In the case that *inDiAddCounter* is set to "off", the current counter value is overwritten by the new counter value and previous counter value might be lost.
- In the case that *inDiAddCounter* is set to "on", the new counter value is added to the current counter value.

When the count number is read by the host:

- If *inDiResetCounterOnRead* is set to "off", reading the counter value does not affect the counter value.
- If *inDiResetCounterOnRead* is set to "on", the counter value is reset after reading it.

The parameter *inDiResetCounterOnRead* has only an effect when *inDiAddCounter* set to "on".

In order to avoid counter overflows parameter *inDiAddCounter* "on" should be combined with *inDiResetCounterOnRead* "on" parameter.



**Fig. 13 Count Value Add Mode**

Fig. 13 shows a non-periodic input signal with 10 pulses in total. The counter value is read at 3 times. In the following it is explained how the options affect the count value.

The value "Count" in the diagram refers to the internal counter which is accumulated during count interval  $T_{\text{Count}}$ .

Mode	Value Read 1	Value Read 2	Value Read 3
<i>inDiAddCounter</i> = "on" <i>inDiResetCounterOnRead</i> = "on"	2	3	3
<i>inDiAddCounter</i> = "on" <i>inDiResetCounterOnRead</i> = "off"	2	5	8
<i>inDiAddCounter</i> = "off" <i>inDiResetCounterOnRead</i> = "off" (Default)	2	3	2

**Tab. 5 Count Mode Options Value Read Results**

Tab. 5 lists the count values at the 3 reading positions for the *inDiAddCounter* and *inDiResetCounterOnRead* options.

The options *inDiAddCounter* and *inDiResetCounterOnRead* are set to "on" which is the most useful combination. (→ Tab. 5, first row)

On Read 1 the returned counter value is 2 since the internal counter value when  $T_{\text{Count}}$  ended was 2. The counter value is reset on reading and 1 pulse is carried over to the next  $T_{\text{Count}}$  interval.

On Read 2 the returned counter value is 3 because of the internal counter value 2 plus the carried over counter 1 from the last count interval. The counter value is reset on read.

At the reading 3 the read value is 3. The counter value is reset on read.

#### Count mode with *inDiAddCounter* = "on"

In the case that *inDiAddCounter* is "on" and *inDiResetCounterOnRead* is "off" the pulses are accumulated but the counter is not reset on reading the value. This causes the counter value will be updated when count interval has finished and the current counter is added to the last counter value. This may result in an overflow when the counter value rolls over its maximum value of 65535.

### 3.3 Commands

Lucid485 IO Modules can be accessed by the Network Frame Protocol, which is documented in the general *Lucid485 User Manual*.

This section describes in detail the commands, which are supported by the DI4/DI8 modules.

#### 3.3.1 Getlo

This command reads the logic value of an input.

Mode	Value
Reflect	Logic input value 0 or 1
Edge Detection	Pending edge event detected value 0 or 1
Count	Counter value 0 to 65535

Tab. 6 Value of Digital Input Channel

Tab. 6 lists the digital input channel modes and how the IO value is interpreted.

Command	Getlo	Access	Read				
Opcode	0x46						
LucidIoCtrl Command Line Tool							
Call (-tL)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -tL -r						
Return	CHn:L <table border="1"> <tr> <td>n</td> <td>Input Channel</td> </tr> <tr> <td>L</td> <td>Input Digital Value</td> </tr> </table>			n	Input Channel	L	Input Digital Value
n	Input Channel						
L	Input Digital Value						
Call (-tN)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -tN -r						
Return	CHn:V <table border="1"> <tr> <td>n</td> <td>Input Channel</td> </tr> <tr> <td>V</td> <td>Input Count Value</td> </tr> </table>			n	Input Channel	V	Input Count Value
n	Input Channel						
V	Input Count Value						

Tab. 7 Getlo Command

#### LucidIoCtrl Command Line Tool Example

Read input channel 0. The module is operating in Reflect or Edge Detection Mode

```
LucidIoCtrl -drs485:COM4:11 -c0 -tL -r
-> CH0:01
```

Read input channel 0 operating in Count Mode

```
LucidIoCtrl -drs485:COM4:11 -c0 -tN -r
-> CH0:0x0064 (100)
```

#### Request Frame

DST	SRC	OPC	P1	P2	LEN	CRC
DST	SRC	0x46	Channel	Value Type	0	CRC-16

Tab. 8 Getlo Network Request Frame

Value	Description									
Channel	Number of input or output channel (Range: 0 to 3/7)									
Value Type	Value Type Supported Value Types									
	<table border="1"> <thead> <tr> <th>Value Type</th> <th>Value Range</th> <th>Size</th> </tr> </thead> <tbody> <tr> <td>Digital Logic Value (0x00)</td> <td>0 / 1</td> <td>1 Byte</td> </tr> <tr> <td>Digital Counter Value (0x0A)</td> <td>0 ~ 65,535</td> <td>2 Bytes</td> </tr> </tbody> </table>	Value Type	Value Range	Size	Digital Logic Value (0x00)	0 / 1	1 Byte	Digital Counter Value (0x0A)	0 ~ 65,535	2 Bytes
Value Type	Value Range	Size								
Digital Logic Value (0x00)	0 / 1	1 Byte								
Digital Counter Value (0x0A)	0 ~ 65,535	2 Bytes								

Tab. 9 GetIo Request

Response Frame:

DST	SRC	Status	LEN	Data Field	CRC
DST	SRC	Status	Length	Value	CRC-16

Tab. 10 Get Io Network Response Frame

Returns Execution Status Code, documented in the general *Lucid485 User Manual*.

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



### 3.3.2 GetloGroup

This command reads the logic input values of a group of input of the same Value Type.

Command	GetloGroup	Access	Read				
Opcode	0x48						
LucidIoCtrl Command Line Tool							
Call (-tL)	LucidIoCtrl -drs485:[COMx:addr] -c[Channels] -tL -r  <u>Channels:</u> Comma separated list of channels e.g. -c0,1,3						
Return	List of values sorted from lower to higher channels CHn:LL <table border="1" style="margin-left: 20px;"> <tr> <td>n</td> <td>Input Channel</td> </tr> <tr> <td>LL</td> <td>Input Digital Value</td> </tr> </table>			n	Input Channel	LL	Input Digital Value
n	Input Channel						
LL	Input Digital Value						
Call (-tN)	LucidIoCtrl -drs485:[COMx:addr] -c[Channels] -tN -r  <u>Channels:</u> Comma separated list of channels e.g. -c0,1,3						
Return	List of values sorted from lower to higher channels CHn:vv <table border="1" style="margin-left: 20px;"> <tr> <td>n</td> <td>Input Channel</td> </tr> <tr> <td>vv</td> <td>Input Counter Value</td> </tr> </table>			n	Input Channel	vv	Input Counter Value
n	Input Channel						
vv	Input Counter Value						

Tab. 11 GetloGroup Command

#### LucidIoCtrl Command Line Tool Example

Read input values of input channel 0, 1 and 3:

```
LucidIoCtrl -drs485:COM4:11 -c0,1,3 -tL -r
CH0:00 CH1:01 CH3:01
```

#### Request Frame

DST	SRC	OPC	P1	P2	LEN	CRC
DST	SRC	0x48	Channel Mask	Value Type	0	CRC-16

Tab. 12 GetloGroup Network Request Frame

Value	Description																											
Channel Mask	<p>Channel Mask Specifies the output channels to access</p> <table border="1" data-bbox="395 365 1046 799"> <thead> <tr> <th>Channel</th> <th>Bit Position</th> <th>Value</th> </tr> </thead> <tbody> <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> </tbody> </table> <p>Values are bitwise or combined Size of P1 is 1 or 2 bytes. If Bit 7 of P1 is set, a subsequent P1A is expected.</p> <p><u>Examples:</u> Accessing channel numbers: 0 and 3      Value = 0x01 OR 0x08 = 0x09 1 and 2      Value = 0x02 OR 0x04 = 0x06 1, 2 and 7   Value P1 = 0x02 OR 0x04 = 0x86                  Value P1A = 0x01 (for channel 7)</p>	Channel	Bit Position	Value	0	0	0x01	1	1	0x02	2	2	0x04	3	3	0x08	4	4	0x10	5	5	0x20	6	6	0x40	7	P1A 0	P1=0x80 P1A = 0x01
Channel	Bit Position	Value																										
0	0	0x01																										
1	1	0x02																										
2	2	0x04																										
3	3	0x08																										
4	4	0x10																										
5	5	0x20																										
6	6	0x40																										
7	P1A 0	P1=0x80 P1A = 0x01																										
Value Type	<p>Supported Value Types</p> <table border="1" data-bbox="395 1256 1345 1476"> <thead> <tr> <th>Value Type</th> <th>Value Range</th> <th>Response Len</th> </tr> </thead> <tbody> <tr> <td>Digital Logic Value (0x00)</td> <td>0 / 1</td> <td>1 Byte</td> </tr> <tr> <td>Digital Counter Value (0x0A)</td> <td>0 ~ 65,535</td> <td>2 Bytes</td> </tr> </tbody> </table>	Value Type	Value Range	Response Len	Digital Logic Value (0x00)	0 / 1	1 Byte	Digital Counter Value (0x0A)	0 ~ 65,535	2 Bytes																		
Value Type	Value Range	Response Len																										
Digital Logic Value (0x00)	0 / 1	1 Byte																										
Digital Counter Value (0x0A)	0 ~ 65,535	2 Bytes																										

**Tab. 13 GetIoGroup Command**

Response Frame:

DST	SRC	Status	LEN	Data Field	CRC
DST	SRC	Status	Length	Value(s)	CRC-16

**Tab. 14 GetIoGroup Network Response Frame**

Returns Execution Status Code, documented in the general *Lucid485 User Manual*.

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

Example of GetloGroup Request:

Request frame reads input channels 0, 1, and 7

DST	SRC	OPC	P1	P1A	P2	LEN	CRC
0x11	0x10	0x48	0x83	0x01	0x00	0x00	CRC

**Tab. 15 GetloGroup Request Example**Response Frame:

Channel 0 = "0", channel 1 = "1", channel 7 = "1"

DST	SRC	Status	LEN	Data Field			CRC
0x10	0x11	0x00	0x03	CH 0	CH 1	CH 7	CRC
				0x00	0x01	0x01	

**Tab. 16 GetloGroup Response Example**

### 3.4 Parameters

Lucid485 modules are configured by a set of System Configuration Parameters and IO Configuration Parameters.

The parameters are accessible by the SetParam and GetParam commands, which are described in the general *Lucid485 User Manual*.

Parameter values can be made persistent in the non-volatile memory of the microcontroller e.g., by adding the -p argument to LucidIoCtrl or by setting the persistent parameter in the API function to true. Values of persistent parameters are restored when Lucid485 is powered on.

The number of write cycles to the non-volatile memory is limited. Write operations wear out the non-volatile memory and periodical updates of persistent parameters should be avoided in order not to destroy the device over time.

#### 3.4.1 inDiValue

This IO Configuration Parameter reflects the value of the input channel.

In Reflect Mode and Edge Detection Mode the parameter contains the same value which GetIo or GetIoGroup commands return.

If the input is configured in Count Mode this parameter is "0".

<b>Parameter</b>	<i>inDiValue</i>	<b>Access</b>	Read
<b>Address</b>	0x1000		
<b>Values</b>	Input Value		
<b>Default Value</b>	0x00	<b>Parameter Type</b>	1 Byte unsigned
<b>LucidIoCtrl Command Line Tool</b>			
<b>Parameter Name</b>	<i>inDiValue</i>	<b>Parameter Values</b>	0 / 1
<b>Call (Get)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiValue		

Tab. 17 IO Configuration Parameter *inDiValue*

#### LucidIoCtrl Command Line Tool Example

Read value of input channel 0:

```

    LucidIoCtrl -drs485:COM4:11 -c0 -ginDiValue
->    inDiValue=0

```

#### 3.4.2 inDiMode

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

<b>Parameter</b>	<i>inDiMode</i>	<b>Access</b>	Read / Write
<b>Address</b>	0x1100		
<b>Values</b>	Input Mode		
	<b>Byte</b>	<b>Mode</b>	
	0x00	inactive	
	0x01	reflect	
	0x10	risingEdge	
	0x11	fallingEdge	
	0x20	count	
<b>Default Value</b>	0x01	<b>Parameter Type</b>	1 Byte unsigned
<b>LucidIoCtrl Command Line Tool</b>			
<b>Parameter Name</b>	<i>inDiMode</i>	<b>Parameter Values</b>	Mode
<b>Call (Set)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiMode=[Mode] {-p} {--default}		
<b>Call (Get)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiMode		

**Tab. 18 IO Configuration Parameter inDiMode**

LucidIoCtrl Command Line Tool Example

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

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiMode=count -p
```

Read the operation mode of input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiMode  
-> inDiMode=count
```

### 3.4.3 Bit Parameter inDiFlags

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

Parameter	<i>inDiFlags</i>	Access	Read / Write
Address	0x1101		
Values	Bit Parameters		
	Bit Parameter	Bit Position	
	<i>inDiAddCounter</i>	Bit 0	
	<i>inDiResetCounterOnRead</i>	Bit 1	
	<i>inDiInverted</i>	Bit 2	
Default Value	0x00	Parameter Type	1 Byte unsigned

Tab. 19 IO Configuration Parameter *inDiFlags*

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

If *inDiFlags* is changed by the SetParam command the update must be performed in a read-modify-write sequence in order to prevent overwriting other bit parameters.

#### 3.4.3.1 *inDiInverted*

This Bit Parameter configures the input signal value inversion.

Parameter	<i>inDiFlags</i>	Access	Read / Write
Address	0x1101	Parameter bit <i>inDiFlags</i>	
Values	Bit Parameter	Bit Position	
	<i>inDiInverted</i>	Bit 2	
Default Value	Off	Parameter Type	1 Bit
LucidIoCtrl Command Line Tool			
Parameter Name	<i>inDiInverted</i>	Parameter Values	on / off
Call (Set)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiInverted=[Value] {-p} {--default}		
Call (Get)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiInverted		

Tab. 20 IO Configuration Parameter *inDiInverted*

#### LucidIoCtrl Command Line Tool Example

Enable input signal value inversion of channel 0 and make the setting persistent.

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiInverted=on -p
```

Read input signal value inversion flag of channel 0.

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiInverted
```

```
-> inDiInverted=on
```

### 3.4.3.2 inDiAddCounter

This Bit Parameter controls how the counter value is updated after count interval time  $T_{\text{Count}}$  has finished. It is relevant in Count Mode only (→ 3.2.3).

<b>Parameter</b>	<i>inDiFlags</i>	<b>Access</b>	Read / Write
<b>Address</b>	0x1101	Parameter bit <i>inDiFlags</i>	
<b>Values</b>	<b>Bit Parameter</b>		<b>Bit Position</b>
	<i>inDiAddCounter</i>		Bit 0
<b>Default Value</b>	Off	<b>Parameter Type</b>	1 Bit
<b>LucidIoCtrl Command Line Tool</b>			
<b>Parameter Name</b>	<i>inDiAddCounter</i>	<b>Parameter Values</b>	on / off
<b>Call (Set)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiAddCounter=[Value] {-p} {--default}		
<b>Call (Get)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiAddCounter		

Tab. 21 IO Configuration Parameter *inDiAddCounter*

#### LucidIoCtrl Command Line Tool Example

Enable counter add on update for input channel 0 and make the setting persistent.

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiAddCounter=on -p
```

Read counter add setting of input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiAddCounter  
inDiAddCounter=on
```

### 3.4.3.3 inDiResetCounterOnRead

This Bit Parameter controls how to update the counter value after it was read. It is relevant in Count Mode only (→ 3.2.3).

<b>Parameter</b>	<i>inDiFlags</i>	<b>Access</b>	Read / Write
<b>Address</b>	0x1101	Parameter bit <i>inDiFlags</i>	
<b>Values</b>	<b>Bit Parameter</b>		<b>Bit Position</b>
	<i>inDiResetCounterOnRead</i>		Bit 1
<b>Default Value</b>	Off	<b>Parameter Type</b>	1 Bit
<b>LucidIoCtrl Command Line Tool</b>			
<b>Parameter Name</b>	<i>inDiResetCounterOnRead</i>	<b>Parameter Values</b>	on / off
<b>Call (Set)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiResetCounterOnRead=[Value] {-p} {--default}		
<b>Call (Get)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiResetCounterOnRead		

Tab. 22 IO Configuration Parameter *inDiResetCounterOnRead*

### LucidIoCtrl Command Line Tool Example

Enable counter reset on read for input channel 0 and make the setting persistent.

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiResetCounterOnRead=on -p
```

Read counter reset setting of input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiResetCounterOnRead
-> inDiResetCounterOnRead=on
```

### 3.4.4 inDiScanTime

This IO Configuration Parameter specifies the scan time  $T_{Scan}$  of the digital input.

$T_{Scan}$  defines the scan time interval within the input signal must be stable in order to detect it as valid.

Parameter	<i>inDiScanTime</i>	Access	Read / Write
Address	0x1111		
Values	$T_{Scan}$ in $\mu$ s (micro seconds) $80 \mu\text{s} \leq T_{Scan} \leq 1 \text{ s}$		
Default Value	50,000 (50 ms)	Parameter Type	4 Bytes unsigned
<b>LucidIoCtrl Command Line Tool</b>			
Parameter Name	<i>inDiScanTime</i>	Parameter Values	Time [ $\mu$ s]
Call (Set)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiScanTime=[Time] {-p} {--default}		
Call (Get)	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiScanTime		

**Tab. 23 IO Configuration Parameter inDiScanTime**

### LucidIoCtrl Command Line Tool Example

Set  $T_{Scan}$  of input channel 0 to 1.5 s and make the setting persistent.

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiScanTime=1500000 -p
```

Read  $T_{Scan}$  parameter of input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiScanTime
-> inDiScanTime=1500000
```

### 3.4.5 inDiCountTime

This IO Configuration Parameter specifies the count time  $T_{Count}$  of the digital input in Count Mode ( $\rightarrow$  3.2.3).

The count interval specifies the time during which pulses of the input signal are accumulated.



<b>Parameter</b>	<i>inDiCountTime</i>	<b>Access</b>	Read / Write
<b>Address</b>	0x1112		
<b>Values</b>	T <sub>Count</sub> in $\mu$ s (micro seconds) 1 ms $\leq$ T <sub>Count</sub> $\leq$ 1 h		
<b>Default Value</b>	5,000,000 (5 s)	<b>Parameter Type</b>	4 Bytes unsigned
<b>LucidIoCtrl Command Line Tool</b>			
<b>Parameter Name</b>	<i>inDiCountTime</i>	<b>Parameter Values</b>	Time [ $\mu$ s]
<b>Call (Set)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -sinDiCountTime=[Time] {-p} {--default}		
<b>Call (Get)</b>	LucidIoCtrl -drs485:[COMx:addr] -c[Channel] -ginDiCountTime		

**Tab. 24 IO Configuration Parameter *inDiCountTime***

### LucidIoCtrl Command Line Tool Example

Set T<sub>Count</sub> of input channel 0 to 10 s and make the setting persistent.

```
LucidIoCtrl -drs485:COM4:11 -c0 -sinDiCountTime=10000000 -p
```

Read T<sub>Count</sub> parameter of input channel 0

```
LucidIoCtrl -drs485:COM4:11 -c0 -ginDiCountTime  
->inDiCountTime=10000000
```

## 4 Modbus RTU Protocol

Address	Type	Width	Description
0x1000	Holding	32	IO Channel Number 0 Value
0x1002	Holding	32	IO Channel Number 1 Value

Tab. 25 Modbus RTU Registers

## 5 Specification

Parameter		Condition	Value
<b>Inputs</b>			
	No of Input Channels		4/8
<b>Inputs - Electrical Characteristics</b>			
Input Signal Maximum Low Level	5V	$U_{05MaxLow}$	2.5 V
	10V	$U_{10MaxLow}$	6.0 V
	24V	$U_{24MaxLow}$	16.0 V
Input Signal Minimum High Level	5V	$U_{05MinHigh}$	3.5 V
	10V	$U_{10MinHigh}$	8.5 V
	24V	$U_{24MinHigh}$	21.0 V
Input Signal Maximum Voltage <sup>Note1</sup>	5V	$U_{5Max}$	7.5 V
	10V	$U_{10Max}$	15 V
	24V	$U_{24Max}$	30 V
Input Impedance		$R_{In}$	> 1 k $\Omega$
<b>Inputs – Timing Characteristic</b>			
	$T_{Scan}$		$t_{Min} < T_{Scan} < 1\text{ s}$
	$T_{Count}$		$1\text{ ms} < T_{Count} < 1\text{ h}$
	Minimum pulse length	$t_{Min}$	100 $\mu\text{s}$
Maximum Frequency of Input Signal in Count Mode	DC 50 %		2,000 Hz
	20% > DC < 80%		500 Hz
<b>Module – Host Communication</b>			
	Baudrate		1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps
	Parity		NONE, EVEN, ODD
	Stopbits		1 or 2
	Databits		8
	Bus Termination (If enabled)		120 $\Omega$
<b>Module – Environment</b>			
	Temperature	Storage	-20 °C ... +70 °C
		Operation	0 °C ... +55 °C
	Humidity		< 85 % RH, non-condensing
<b>Module – Housing</b>			
	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

Tab. 26 Device Specification

Note1 Maximum forward and reverse voltage

Module is specified at environmental temperature of 25°C.

## 6 Order Information

Order Code	Product
L485-DI4-I-5	Lucid485 Digital Input Serial Module with 4 insulated Channels for 5 Volt Signals
L485-DI4-I-10	Lucid485 Digital Input Serial Module with 4 insulated Channels for 10 Volt Signals
L485-DI4-I-24	Lucid485 Digital Input Serial Module with 4 insulated Channels for 24 Volt Signals
L485-DI8-I-5	Lucid485 Digital Input Serial Module with 8 insulated Channels for 5 Volt Signals
L485-DI8-I-10	Lucid485 Digital Input Serial Module with 8 insulated Channels for 10 Volt Signals
L485-DI8-I-24	Lucid485 Digital Input Serial Module with 8 insulated Channels for 24 Volt Signals

Tab. 27 Order Codes

## 7 Document Revision

Date	Rev.	
2024/08/22	1.0	Initial Document

Tab. 28 Document Revision



deciphe it GmbH  
 Schäferstr. 5  
 87600 Kaufbeuren / Germany  
[www.lucid-control.com](http://www.lucid-control.com)