

# LUMIPLUS

MODBUS-HANDBUCH V2.0

© Fluidra, S.A. 2013. Alle Rechte vorbehalten.

**Alle Marken sind eingetragene Marken von Fluidra SA und/oder ihren verbundenen Unternehmen oder ihrer jeweiligen Eigentümer. Fluidra SA und die Eigentümer sind Eigentümer aller Rechte, Eigentumstitel oder Ansprüche am Handbuch, der Technologie und den Informationen, auch allen Teilen, Kopien oder Veränderungen davon.**

**Es wurde größtmögliche Sorgfalt aufgewendet, um dafür zu sorgen, dass die Informationen korrekt sind. Aufgrund der fortlaufenden Produktänderungen behält sich Fluidra jedoch das Recht vor, ohne Vorankündigung Änderungen an den Produkten und den technischen Informationen vorzunehmen.**

<b>1.</b>	<b>EINFÜHRUNG IN MODBUS UND DAS PRODUKT</b>	<b>4</b>
1.1.	Funktionsprinzip	4
1.2.	Grundlegende Merkmale	4
<b>2.</b>	<b>ELEKTRISCHE ANSCHLÜSSE</b>	<b>5</b>
<b>3.</b>	<b>MERKMALE DER VERKABELUNG</b>	<b>5</b>
<b>4.</b>	<b>ISOLIERUNG DES BUS UND ABSCHLUSSWIDERSTAND</b>	<b>6</b>
<b>5.</b>	<b>LED-ANZEIGE</b>	<b>7</b>
<b>6.</b>	<b>MODBUS-FUNKTIONEN</b>	<b>7</b>
6.1.	Unterstützte Funktionen	7
6.2.	Ausnahmeantworten	8
<b>7.</b>	<b>BESCHREIBUNG UND KONFIGURATION DES GERÄTS</b>	<b>8</b>
7.1.	Allgemeine Beschreibung	8
7.2.	Statusdiagramm	9
<b>7.3.</b>	<b>Auswahl der Adresse und der Baud-Rate</b>	<b>10</b>
7.3.1.	Adresseinstellung	10
7.3.2.	Auswahl der Baud-Rate	10
<b>7.4.</b>	<b>Sendung</b>	<b>11</b>
<b>7.5.</b>	<b>Watchdog</b>	<b>11</b>
7.5.1.	Auslösen des Watchdog	11
7.5.2.	Watchdog-Konfiguration	12
7.5.3.	Vordefinierte Farbe/Sequenz des Watchdogs	13
<b>8.</b>	<b>BETRIEBSMODI</b>	<b>14</b>
<b>8.1.</b>	<b>Basismodus</b>	<b>14</b>
8.1.1.	Vordefinierte Farbe/Sequenz bei Start	14
8.1.2.	Prüfen des aktuellen Status	15
8.1.3.	Änderungsanforderungen des Aktuators	17

8.1.4.	EIN-Anforderung	17
8.1.5.	AUS-Anforderung	18
8.1.6.	Ändern der Farbe	18
8.1.7.	Ändern der Sequenz	20
8.1.8.	Ändern der Geschwindigkeit	21
8.1.9.	Ruhezeit	22
8.1.10.	Überprüfung des Registers der selbthaltenden Alarme	22
8.1.11.	Überprüfung des Registers der Sofortalarne	23
<b>9.</b>	<b>GRUNDLEGENDE MODBUS-RTU-REGISTERZUORDNUNG</b>	<b>24</b>
<b>10.</b>	<b>PRODUKTVERSION</b>	<b>29</b>
<b>11.</b>	<b>FEHLERBEHEBUNGSLEITFADEN</b>	<b>29</b>

## 1. EINFÜHRUNG IN MODBUS UND DAS PRODUKT

Vielen Dank, dass Sie unser Lumiplus mit MODBUS-RTU-Funktionen erworben haben. Dieses Handbuch richtet sich an Installationsfachkräfte. Wenn Sie keine Installationsfachkraft sind, wenden Sie sich an Ihren zugelassenen Händler.

MODBUS ist ein offener Feldbus, mit dem derzeit Geräte in aller Welt über einen Hauptcontroller miteinander verbunden sind. Aus diesem Grund war MODBUS für uns die erste Wahl, um unseren Kunden und Partnern automatisierte Lösungen anzubieten, in die sich nicht nur unsere Produkte, sondern auch eine große Auswahl anderer Komponenten und Controller von Drittanbietern leicht integrieren lassen.

MODBUS, MODBUS-RTU und andere damit zusammenhängende Namen sind eingetragene Marken der MODBUS Organization. Diesbezügliche Informationen und Dokumentation finden Sie auf <http://www.Modbus.org/>

### 1.1. FUNKTIONSPRINZIP

In Lumiplus ist MODBUS-RTU als Kommunikationsprotokoll implementiert, das dessen Betrieb und Überwachung über die Automatisierungsumgebung von MODBUS ermöglicht. Auch vorbeugende Instandhaltung und Fehleranalyse sind dank der Implementierung der internen Register in Lumiplus möglich, in denen die wichtigsten Betriebs- und Fehlerereignisse erfasst werden.

Unabhängig vom Installationsort von Lumiplus ist keine Verbindung zu einem MODBUS-System erforderlich, sofern Sie keine externe Steuerung oder Überwachung planen. Lumiplus kann auf herkömmliche Weise im lokalen Modus ohne Verwendung des MODBUS-Layers betrieben werden.

Wir gehen jedoch davon aus, dass die Implementierung von MODBUS-RTU in Lumiplus unseren erfahrenen Kunden und Partnern dank der Einfachheit und Flexibilität des MODBUS-RTU-Layers eine große Bandbreite neuer Möglichkeiten und Implementierungsszenarios eröffnet.

Anhand einer MODBUS-RTU-Nachricht kann Lumiplus zu einer bestimmten Farbe, Sequenz oder Geschwindigkeit wechseln, Fehler melden, Verlaufsdaten erfassen usw. Dadurch erhält der Benutzer/Installateur zahlreiche neue Funktionen, die auf der Automatisierung eines bereits vorhandenen und bewährten Lumiplus basieren.

### 1.2. GRUNDLEGENDE MERKMALE

Das MODBUS-Kommunikationssystem umfasst eine Master-Slave-Implementierung unter Geräten, die eine physische Verbindung teilen. Für Lumiplus erfolgt diese physische Verbindung über ein serielles RS-485 Halbduplex-Layer. Dieses wurde anderen Optionen wegen seiner weit verbreiteten Implementierung und Robustheit vorgezogen.

Für Lumiplus wurde eine RS-485 Halbduplex-Kabelverbindung implementiert, und für den Projektor wurde die Ausführung in einem Single-Master-Netzwerk gewählt. In dieser Implementierung sind die Rollen von Master und Slave klar definiert, was für ein korrektes Verständnis und die richtige Systemimplementierung wesentlich ist.

Master-Gerät: Das Gerät, das den Datenaustausch über den Bus steuert und erforderlichenfalls die Aufgaben unter den einzelnen Slaves koordiniert (z. B. PLC Programmable Logic Controller, SCADA usw.).

Slave-Gerät: Das an den Bus angeschlossene Gerät, das die Anforderungen des Masters bedient, sei es die Sendung von Informationen an den Master oder die Ausführung von Aufgaben, mit denen es vom Master beauftragt wurde.

## 2. ELEKTRISCHE ANSCHLÜSSE

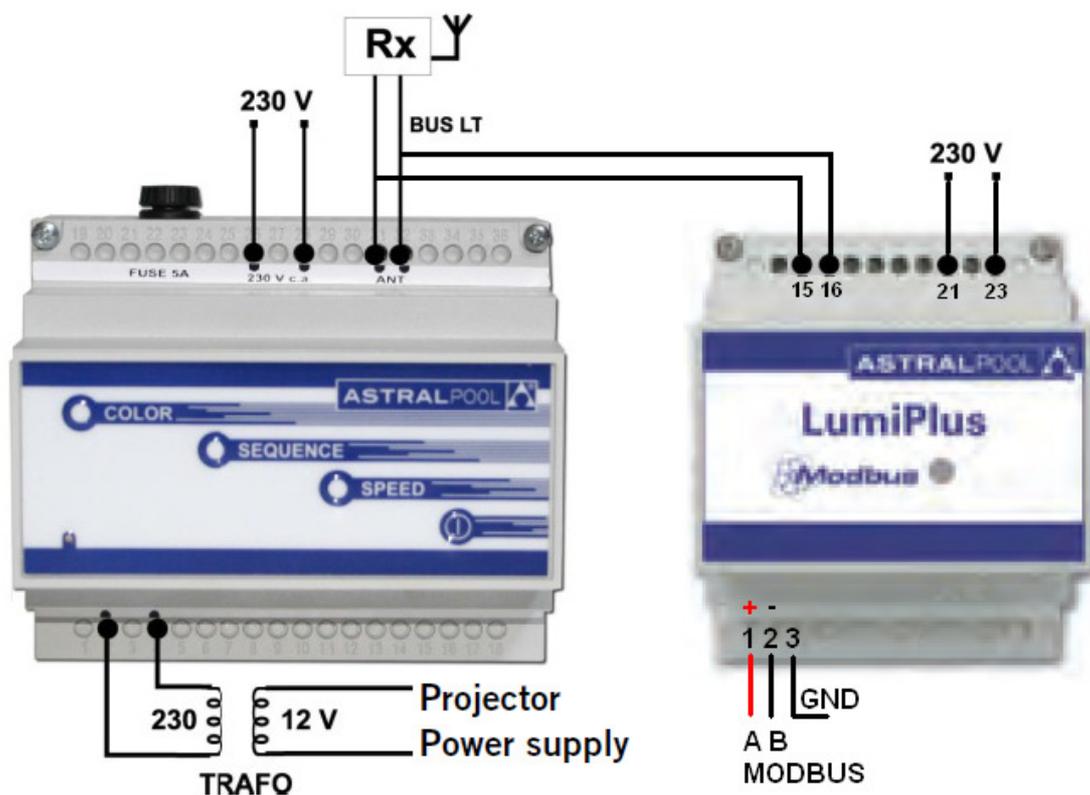


Abbildung 1 Elektrische Anschlüsse

Hinweis: Manche Hersteller weisen für den RS-485-Port den Anschluss „A“ als „+“ und „B“ als „-“ zu, während andere diese Nomenklatur umgekehrt verwenden. Lumiplus verwendet „A“ als „+“ und „B“ als „-“. Beachten Sie diesen Punkt, wenn Sie Geräte anderer Hersteller an den Bus anschließen.

## 3. MERKMALE DER VERKABELUNG

Die empfohlene Verkabelung für die MODBUS-RTU-Kommunikation basiert auf einer linearen Struktur mit aktivem Bus und Abschlüssen an beiden Enden. Es ist möglich, im laufenden Betrieb ohne Beeinträchtigung anderer Geräte weitere Geräte anzuschließen und zu trennen. Das Kabel muss gemäß der Norm EN 50 170 verdreht und abgeschirmt sein.

Die vom Gerät unterstützte Datenübertragungsrate gilt für eine Kabellänge von bis zu 1200 m ohne Verstärker und bis zu 10 km mit Verstärkern, sofern die Installation dem Standard entspricht.

Für die symmetrisch verdrehten Paare in einem RS-485-System ist ein Wellenwiderstand von mehr als 100 Ohm zu empfehlen, insbesondere für Kommunikationen mit einer Baud-Rate von 19200 oder höher.

#### 4. ISOLIERUNG DES BUS UND ABSCHLUSSWIDERSTAND

Wenn der Benutzer Zugriff auf den Kommunikationsbus hat, muss dieser doppelt isoliert sein. Generell hängt die Zugänglichkeit des Bus für Benutzer von der jeweiligen Installation ab. Lumiplus verfügt über eine Sicherheitsisolierung in der physischen Busschicht des Bus. Zudem wird aus Sicherheitsgründen empfohlen, sicherzustellen, dass die weiteren Geräte am Bus die gleiche Isolierung implementieren.

Zudem verbessert die Nutzung isolierter Busgeräte nicht nur die Sicherheitsstufe, sondern auch die Zuverlässigkeit der Ausrüstung und bietet höhere Immunität gegenüber elektromagnetischen Interferenzen, längere Lebensdauer, höhere Zuverlässigkeit und mehr Stabilität im Temperaturbereich.

Wenn ein oder mehrere Geräte angeschlossen werden, die den gleichen physischen Busanschluss nutzen, wird empfohlen, einen Abschlusswiderstand an jedem Ende des Bus zu verwenden, vor allem wenn ein längeres Kabel oder höhere Kommunikationsgeschwindigkeit verwendet wird. Der Abschlusswiderstand wird verwendet, um zu verhindern, dass ein Radiofrequenzsignal (RF) zurückreflektiert wird und Interferenzen verursacht. Er muss an beiden Enden des Bus verwendet und parallel angeschlossen werden (siehe die Abbildung unten). Ein typischer Wert für diesen Widerstand ist 120Ω, 0,5W. Der Wert des Widerstands muss an beiden Enden gleich sein. Die Abschlusswiderstände sind die Widerstände  $R_T$  in der Abbildung unten.

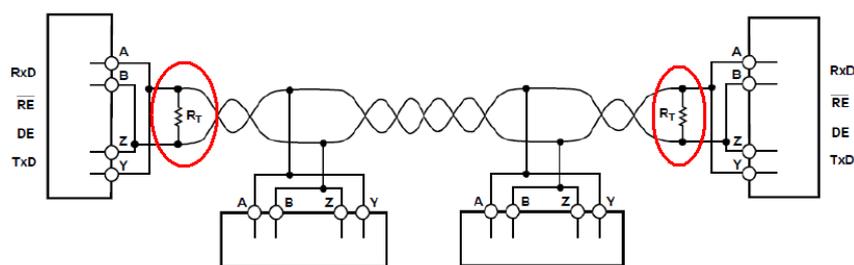


Abbildung 2 Abschlusswiderstände

## 5. LED-ANZEIGE

Das Lumiplus-Modul verfügt über eine LED, die bei Einschalten des Moduls aufleuchtet. Bei der Datenübertragung über Modbus blinkt diese LED.



Abbildung 3 Lumiplus-LED-Anzeige

## 6. MODBUS-FUNKTIONEN

### 6.1. UNTERSTÜTZTE FUNKTIONEN

**Bitte gehen Sie beim Verwenden dieser Funktionen sorgfältig vor und vergewissern Sie sich, dass Sie die richtige Funktion nutzen.**

#### BIT-ZUGRIFFSMODUS

Die Funktionen im Bit-Zugriffsmodus sind gemäß dem MODBUS-RTU-Standard implementiert, der in folgendem Dokument beschrieben ist:  
[http://www.Modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b.pdf](http://www.Modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf)

0x01 READ COILS

0x0F WRITE MULTIPLE COILS

0x05 WRITE SINGLE COIL. *Diese Funktion ist in Lumiplus nicht implementiert. Sie kann mit der Funktion „Write Multiple Coils“ für den Sonderfall, dass die Anzahl der zu schreibenden Coils eins beträgt, umgangen werden.*

0x02 READ DISCRETE INPUTS.

#### REGISTER-ZUGRIFFSMODUS

Die Funktionen im Register-Zugriffsmodus sind gemäß dem MODBUS-RTU-Standard implementiert, der in folgendem Dokument beschrieben ist:  
[http://www.Modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b.pdf](http://www.Modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf) Register sind in der Regel Ganzzahlen ohne Vorzeichen von 16 Bit.

0x03 READ HOLDING REGISTERS

0x04 READ INPUT REGISTERS

0x10 WRITE MULTIPLE REGISTERS

## 6.2. AUSNAHMEANTWORTEN

Die Ausnahmeantworten sind gemäß dem MODBUS-RTU-Standard implementiert, der im Kapitel über MODBUS-Ausnahmeantworten beschrieben ist:

[http://www.Modbus.org/docs/Modbus\\_Application\\_Protocol\\_V1\\_1b.pdf](http://www.Modbus.org/docs/Modbus_Application_Protocol_V1_1b.pdf)

Die Ausnahmeantworten 1 bis 3 sind implementiert.

## 7. BESCHREIBUNG UND KONFIGURATION DES GERÄTS

### 7.1. ALLGEMEINE BESCHREIBUNG

Wenn über Modbus eine Anforderung erfolgt und später eine weitere Anforderung über die Fernbedienung, gibt Modbus die über Modbus festgelegte Farbe an. Dies an der Unidirektionalität zwischen dem Modbus-Modul und dem Lumiplus-Modulator, der über die Fernbedienung gesteuert wird.

Wenn gleichzeitige Anforderungen über Modbus und die Fernbedienung eingehen und kein Konflikt vorliegt, tritt die letzte Anforderung in Kraft.

Generell wird die Einheitlichkeit der an spezifische Register gesendeten Werte nicht überprüft. Daher ist der Bediener dafür verantwortlich, auf diese Einheitlichkeit zu achten.

In diesem Handbuch sind hexadezimale Zahlen im Format **0xZZ** dargestellt, wobei ZZ die Zahl ist.

Die Registerzuordnung für das 7.2 Statusdiagramm und alle untenstehenden Erläuterungen finden Sie in Kapitel 9 Grundlegende Modbus-RTU-Registerzuordnung.

## 7.2. STATUSDIAGRAMM

Beim Einschalten des Systems ist der Status immer „Start“. Der Modus „Start“ gibt nicht an, dass eine Leuchte eingeschaltet ist, sondern weist nur auf den Stromanschluss von Lumiplus hin und darauf, dass nach dem Stromanschluss KEINE Anforderung erfolgt ist.

Die vordefinierte Farbe beim Start, die Sequenz und die Geschwindigkeit können geändert werden; der Startmodus kann auch mit allen Leuchten ausgeschaltet konfiguriert werden. Von „Start“ kann zum Status „Farbe anfordern“ gewechselt werden, indem eine Aktualisierung der vordefinierten Farbe angefordert wird; der Wechsel zum Status „Sequenz und Geschwindigkeit anfordern“ ist über die Anforderung der Aktualisierung von Sequenz und Geschwindigkeit möglich, und in den Status „Watchdog“ kann gewechselt werden, falls dieser ausgelöst wird.

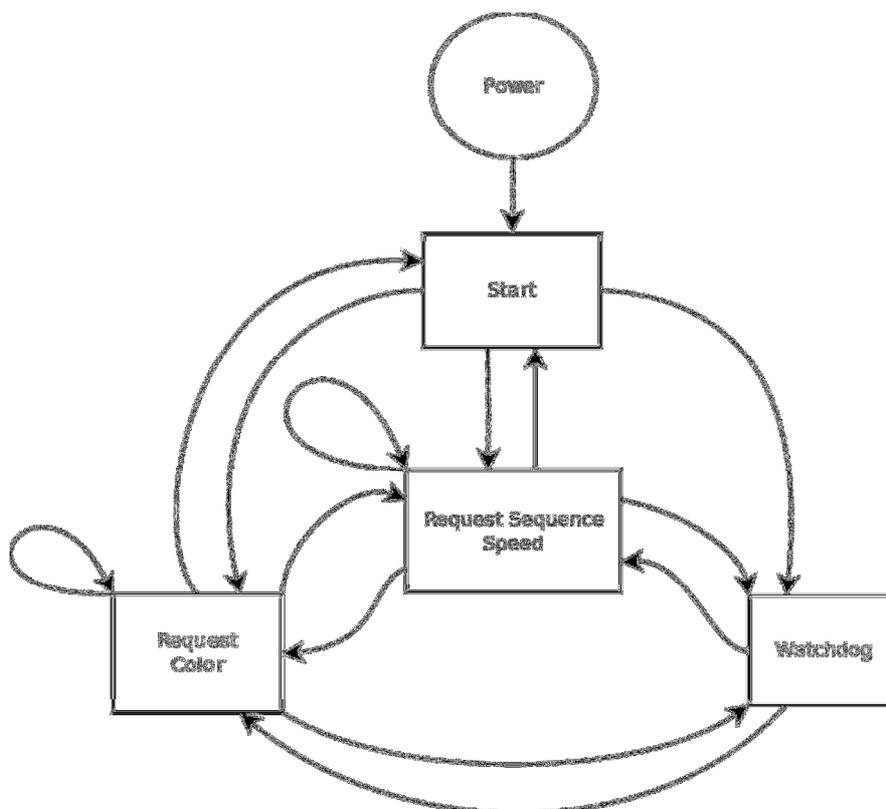


Diagramm 1 Statusdiagramm

## 7.3. AUSWAHL DER ADRESSE UND DER BAUD-RATE

### 7.3.1. ADRESSEINSTELLUNG

Die Adresse von Lumiplus im Bus wird über das Holding Register 0x00 festgelegt.

ID\_Address:        Adresse von Lumiplus im Bus.

Werkseitiger Wert:        0x30.

Vorgeschlagener Bereich: 0x30 - 0x3F.

Der werkseitige Standardwert für Lumiplus ist (0x30). Sie können diesen Wert jedoch durch Schreiben in diese Holding Registers ändern, sofern Sie darauf achten, dass Sie keine Kollisionen oder Konflikte mit anderen Slave-Adressen einfügen.

Beispiel: Ändern der Adress-ID von 0x30 (Standard) zu 0x31.

Gesendete Nachricht: 30 10 00 00 00 01 02 00 31 3E 15

Dabei gilt:

30 ist die Adresse des Slave. (die derzeitige ID-Adresse).

10 ist die verwendete Funktion. Write Multiple Registers.

00 00 ist die Adresse des Holding Register, das geschrieben wird.

00 01 ist die Anzahl der Holding Register, die geschrieben werden. In diesem Fall 1.

02 ist die Anzahl der Datenbyte, die gesendet werden.

00 31 ist die neue ID-Adresse.

3E 15 ist die CRC.

### 7.3.2. AUSWAHL DER BAUD-RATE

Die Auswahl der Baud-Rate in der Kommunikation in Serie mit Lumiplus erfolgt über das Holding Register 0x01. Standardmäßig sind 9600 bps und 8E1 (8 Datenbits, gerade Parität, 1 Stoppbit) implementiert. Es werden aber auch 19200 bps und 2 Stoppbit unterstützt (wenn KEINE Parität implementiert ist).

N2-Frames müssen unterstützt werden, damit die Anforderung des MODBUS-Standards, dass 11 Bit pro Byte gesendet werden, eingehalten wird (1 Startbit + 8 Datenbits + 1 Paritätsbit + 1 Stoppbit). Wenn eine Konfiguration ohne Parität gewählt wird, werden 2 Stoppbits eingefügt, um die vom Standard geforderte 11-Bit-Struktur beizubehalten.

Aus Gründen der Kompatibilität werden auch die N1-Frames unterstützt. Beachten Sie jedoch, dass damit der MODBUS-Standard nicht eingehalten wird, da nur 10 Bit pro Byte verwendet werden.

Daher wird die Auswahl der Baud-Rate und Frames durch Definieren der Baud-Rate (in Baud), der Anzahl der Datenbits, der Parität und Anzahl der Stoppbits abgeschlossen.

## COM\_Setup: Konfiguration der Kommunikation

Werkseitiger Wert:	0	9600, 8E1
Unterstützte Werte:	0	9600, 8E1
	1	19200, 8E1
	2	9600, 8N2
	3	19200, 8N2
	4	9600, 8N1
	5	19200 8N1

### 7.4. SENDUNG

Sendung wird von Lumiplus nicht unterstützt.

### 7.5. WATCHDOG

Der Watchdog ist eine in Lumiplus implementierte Funktion, die prüft, ob die Kommunikation im Bus aktiv ist. Dies ist lediglich eine Sicherheitsfunktion, mit der Fehler in der Kommunikation festgestellt werden sollen und die den Wechsel in einen vordefinierten Status ermöglicht.

Der Watchdog wird ausgelöst, wenn das Gerät nicht innerhalb der in der Watchdog-Zeit angegebenen Zeitspanne zwei Nachrichten erhält.

Entsprechend müssen die Wartezeit zum Auslösen des Watchdog und die in diesem Fall auszuführenden Aufgaben definiert werden.

Wenn der Watchdog aktiviert wurde, werden alle über Modbus erfolgten Anforderungen nicht ausgeführt. In diesem Status kann Lumiplus nur gelesen werden und akzeptiert keine Änderungen. Der Alarm des Watchdogs im Holding Register 0x20 befindet sich auf 1; nur wenn dieser Alarm über Modbus auf 0 gesetzt oder die Stromzufuhr unterbrochen wird, lässt Lumiplus Änderungen oder Anforderungen zu.

Um eine Fehlerkorrektur zu ermöglichen, falls der Watchdog nicht korrekt konfiguriert wurde, wird der Watchdog für die ersten 30 Sekunden nach Einschalten von Lumiplus angehalten. So erhält der Benutzer die Möglichkeit, die Watchdog-Zeit zu ändern, wenn eine zu kurze Zeit konfiguriert wurde, oder den Watchdog zu deaktivieren.

#### 7.5.1. AUSLÖSEN DES WATCHDOG

Die Auslösezeit für den Watchdog wird in Holding Register 0x10 definiert. Die Zeit wird in Sekunden festgelegt. „0s“ bedeutet, dass der Watchdog deaktiviert ist. Dies ist der Standardwert.

Um die Watchdog-Funktion zu aktivieren, legen Sie die Watchdog-Zeit auf einen anderen Wert als 0 fest. Beachten Sie jedoch die Auswirkungen dieses Werts, da lange Zeiten sinnvolle Konfigurationen verzögern können.

Beispiel: Legen Sie die Watchdog-Auslösezeit auf 30 s fest:

Gesendete Nachricht: 30 10 00 10 00 01 02 00 1E 7D 59

Dabei gilt:

30 ist die Adresse des Slave.

10 ist die verwendete Funktion. Write Multiple Registers.

00 10 ist die Adresse des ersten Holding Register, das geschrieben wird.

00 01 ist die Anzahl der Holding Register, die geschrieben werden. In diesem Fall 1.

02 ist die Anzahl der Datenbyte, die gesendet werden.

00 1E ist der gesendete Wert. 30 in dezimal.

7D 59 ist die CRC.

Jetzt ist die Watchdog-Auslösezeit auf 30 s festgelegt. Wenn daher zwei korrekt aufgebaute Nachrichten in weniger als 30 s gelesen werden, selbst wenn diese nicht an Lumiplus gerichtet sind, wird der Watchdog nicht ausgelöst. Andernfalls wird er ausgelöst.

Zum Erfahren der Watchdog-Zeit müssen die Holding Registers gelesen werden.

Die Antwort von „Watchdog\_time“ ist in diesem Fall folgende: 30 03 02 00 1E 45 88

30 ist die Adresse des Slave.

03 ist die verwendete Funktion. Read holding registers.

02 ist die Anzahl der Datenbyte, die gelesen werden.

00 1E ist die konfigurierte Zeit. 30 in dezimal.

45 88 ist die CRC.

---

## 7.5.2. WATCHDOG-KONFIGURATION

Die Watchdog-Konfiguration, Register 0x11, muss im Zusammenhang mit dem Register zum Auslösen des Watchdogs und der vordefinierten Farbe/Sequenz des Watchdogs konfiguriert werden.

Das **höherwertige Byte** des Registers definiert, wie bei Auslösen des Watchdogs vorgegangen wird. Wenn es auf 0 festgelegt ist, wird die vordefinierte Farbe/Sequenz des Watchdogs berücksichtigt. Wenn es auf 1 festgelegt ist, wird die Kommunikationsbrücke zurückgesetzt.

Das **niederwertige Byte** wird in dieser Funktion nicht verwendet. Legen Sie es auf 0 fest.

Beispiel: Konfigurieren des Watchdog zum Ausführen des definierten Vorgangs in der vordefinierten Farbe/Sequenz des Watchdogs:

Gesendete Nachricht: 30 10 00 11 00 01 02 00 00 FC 80

Dabei gilt:

30 ist die Adresse des Slave.  
10 ist die verwendete Funktion. Write multiple registers.  
00 11 ist die Adresse des ersten Registers, das geschrieben wird.  
00 01 ist die Anzahl der Register, die geschrieben werden. In diesem Fall 1.  
02 ist die Anzahl der Byte, die gesendet werden.  
00 ist das höherwertige Byte. Der Watchdog führt den definierten Vorgang in der vordefinierten Farbe/Sequenz des Watchdogs aus.  
00 ist das niederwertige Byte. Wird nicht verwendet, 00 ist ein korrekter Wert.  
FC 80 ist die CRC.

7.5.2 Watchdog-Konfiguration und 7.5.3 Vordefinierte Farbe/Sequenz des Watchdogs sollten zusammen konfiguriert werden. Siehe das Beispiel unter 7.5.3 Vordefinierte Farbe/Sequenz des Watchdogs, Definieren der Farbe/Sequenz, die bei Auslösen des Watchdogs ausgeführt werden.

### 7.5.3. VORDEFINIERTER FARBE/SEQUENZ DES WATCHDOGS

Die vordefinierte Farbe/Sequenz des Watchdogs, Holding Register 0x14, bestimmt die vordefinierte Farbe bzw. Sequenz bei Auslösen des Watchdogs.

Das **höherwertige Byte** legt die **Geschwindigkeit** der Sequenz fest. Es kann Werte von 0 bis 8 haben. Wenn es auf 0 festgelegt ist, beträgt die Geschwindigkeit 0, was bedeutet, dass es sich um eine feste Farbe handelt.

Das **niederwertige Byte** legt die **Farbe** fest, wenn das höherwertige Byte 0 beträgt, oder die **Sequenz**, wenn im höherwertigen Byte ein anderer Wert als 0 festgelegt ist. Das niederwertige Byte kann Werte von 0 bis 12 aufweisen; dabei kann aus 12 möglichen Farben oder 8 möglichen Sequenzen ausgewählt werden. Wenn das niederwertige Byte auf 0 festgelegt ist, wurde keine Farbe ausgewählt, was bedeutet, dass für Lumiplus keine Leuchten eingeschaltet sind.

Wenn das Register als (0xFFFF) konfiguriert ist, behalten die Leuchten die Farbe/Sequenz bei, die sie vor dem Alarmstatus hatten.

Beispiel:

Vordefinierte Farbe/Sequenz des Watchdogs für das Auslösen des Watchdogs festlegen; in diesem Beispiel sind für Lumiplus alle Leuchten ausgeschaltet, und beim Auslösen ist der Status „Watchdog“:

Gesendete Nachricht: 30 10 00 14 00 01 02 00 00 FC D5

Dabei gilt:

30 ist die Adresse des Slave.  
10 ist die verwendete Funktion. Write multiple registers.  
00 14 ist die Adresse des ersten Registers, das geschrieben wird.  
00 01 ist die Anzahl der Register, die geschrieben werden. In diesem Fall 1.  
02 ist die Anzahl der Byte, die gesendet werden.  
00 ist das höherwertige Byte. 0 ist die Geschwindigkeit, 0 ist eine festgelegte Farbe.

00 ist das niederwertige Byte. Farbe 0 ausgewählt. (Farbe 0 ist nur in der vordefinierten Farbe/Sequenz von „Watchdog“ und von „Start“ vorhanden.) Es ist keine Farbe vorhanden, was bedeutet, dass die Leuchten ausgeschaltet sind.

F5 D5 ist die CRC.

HINWEIS: Beachten Sie, dass die Watchdog-Funktion aus Sicherheitsgründen implementiert ist. Ob sie aktiviert wird oder nicht liegt jedoch immer im Ermessen des Installateurs/Integrators. Es muss immer bedacht werden, was ihre Implementierung bedeutet. Verwenden Sie diese Eigenschaft auf Ihre Verantwortung.

## 8. BETRIEBSMODI

### 8.1. BASISMODUS

In diesem Abschnitt wird davon ausgegangen, dass eine korrekte Verbindung mit Lumiplus eingerichtet wurde, und dass daher Adresse, Baud-Einstellungen und Watchdog-Verhalten bereits festgelegt wurden.

#### 8.1.1. VORDEFINIERTER FARBE/SEQUENZ BEI START

Die vordefinierte Farbe/Sequenz bei Start ist die Farbe bzw. Sequenz, die bei Einschalten von Lumiplus verwendet wird.

Das **höherwertige Byte** legt die **Geschwindigkeit** der Sequenz fest. Es kann Werte von 0 bis 8 haben. Wenn es auf 0 festgelegt ist, beträgt die Geschwindigkeit 0; es handelt sich also um eine feste Farbe.

Das **niederwertige Byte** legt die **Farbe** fest, wenn das höherwertige Byte 0 beträgt, oder die **Sequenz**, wenn im höherwertigen Byte ein anderer Wert als 0 festgelegt ist. Das niederwertige Byte kann Werte von 0 bis 12 aufweisen; diese stehen für 12 mögliche Farben oder 8 mögliche Sequenzen. Wenn der Wert auf 0 festgelegt ist, sind alle Leuchten eingeschaltet, wenn Lumiplus eingeschaltet wird.

Wenn das Register als (0xFFFF) konfiguriert ist, behalten die Leuchten die Farbe/Sequenz bei, die sie vor dem letzten Ausschalten hatten.

Die Einrichtung dieses Parameters erfolgt auf die gleiche Weise wie für die vordefinierte Farbe/Sequenz des Watchdogs; der einzige Unterschied ist die Nummer des Holding Register. In diesem Fall ist das Holding Register 0x15.

Für die vordefinierte Farbe/Sequenz für Start bzw. Watchdog ist keine Aktualisierungsanforderung zum Ändern der Farbe erforderlich, wenn das System startet bzw. der Watchdog aktiviert wird.

Spezifische Farbanforderungen und -sequenzen sind entsprechend der folgenden Codierung definiert:

FARBNUMMER	FARBE
1	Rot
2	Grün
3	Blau
4	Gelb
5	Cyan
6	Magenta

FARBNUMMER	FARBE
7	Hellviolett
8	Hellblau
9	Orange
10	Rosa
11	Smaragdgrün
12	Weiß

SEQUENZ	FARBREIHENFOLGE											
1	Rot	Grün	Blau									
2	Cyan	Magenta	Gelb									
3	Grün	Smaragdgrün	Cyan	Hellblau								
4	Rot	Orange	Grün	Orange								
5	Rot	Rosa	Blau	Rosa								
6	Rot	Orange	Grün	Cyan	Blau	Rosa						
7	Hellviolett	Magenta	Cyan	Gelb	Weiß	Gelb	Cyan	Magenta				
8	Rot	Grün	Blau	Gelb	Cyan	Magenta	Hellviolett	Hellblau	Orange	Rosa	Smaragdgrün	Weiß

### 8.1.2. PRÜFEN DES AKTUELLEN STATUS

Der Status, der aktuell für Lumiplus festgelegt ist, kann über das Input Register 0x00 für den Status abgerufen werden. Dieses Register hat für das niederwertige und das höherwertige Byte unterschiedliche Bedeutungen.

Das niederwertigste Bit wird verwendet, um anzuzeigen, ob ein Fehler aufgetreten ist, und wird auf 1 festgelegt, wenn ein Fehler vorhanden ist. Detaillierte Informationen zu den festgestellten Fehlern können aus dem Input Register für Alarme unter 0x01 abgerufen werden. Nur der Watchdog-Alarm ist implementiert. Wenn ein Fehler vorliegt, muss es sich also um einem Watchdog-Fehler handeln (Watchdog wurde ausgelöst). Neben der Ausgabe eines Fehlers berichtet Lumiplus einen nicht wiederherstellbaren Fehler, der ein Ausschalten und Wiedereinschalten des Geräts zum Zurücksetzen bzw. ein Festlegen des Alarms auf 0 erfordert.

Bis zum Zurücksetzen der Alarme akzeptiert Lumiplus keine Änderungen und gibt nur informative Antworten zurück (schreibgeschützt).

Das höherwertige Byte wird verwendet, um den aktuellen Status von Lumiplus anzuzeigen.

Die Codes für die verschiedenen im höherwertigen Byte implementierten Status sind in der folgenden Tabelle aufgeführt:

0x00 Start.  
0x01 Watchdog.  
0x02 Farbe anfordern.  
0x03 Sequenz anfordern.

Beachten Sie die Unidirektionalität, die in Kapitel 7.1 Allgemeine Beschreibung beschrieben ist.

Codierbeispiele (höherwertiges Byte zuerst):

0x0002 2 Status „Start“. EIN, kein Fehler.  
0x0101 257 Status „Watchdog“, Alarm. Leuchte AUS. Zurücksetzung erforderlich. (Dieser Fall ist möglich, wenn die vordefinierte Farbe/Sequenz für Watchdog zum Ausschalten der Leuchten konfiguriert ist).  
0x0103 259 Status „Watchdog“, Alarm. Leuchte EIN. Zurücksetzung erforderlich. (Dieser Fall ist möglich, wenn die vordefinierte Farbe/Sequenz für Watchdog eine Farbe oder eine Sequenz ist).

Beispiel:

Um zu erfahren, in welchem Status sich Lumiplus befindet, muss die gesendete Nachricht wie folgt lauten:

30 04 00 00 00 01 35 EB

Dabei gilt:

30 ist die Adresse des Slave.  
04 ist die verwendete Funktion. Read input registers.  
00 00 ist die Adresse des ersten Input Register, das gelesen wird.  
00 01 ist die Anzahl der Input Registers, die gelesen werden, in diesem Fall 1.  
35 EB ist die CRC.

Die Antwort lautet 30 04 02 00 02 45 35, was den Status „Start“ ohne Alarm und mit eingeschalteten Leuchten angibt.

Dabei gilt:

30 ist die Adresse des Slave.  
04 ist die verwendete Funktion. Read input registers.  
02 ist die Anzahl der Byte, die gelesen werden.  
00 ist das höherwertige Byte; es gibt den Modus „Start“ an.  
02 ist das niederwertige Byte; es gibt an, dass Leuchten eingeschaltet sind.

45 35 ist die CRC.

Im folgenden Beispiel wurde Watchdog aktiviert, und die Zeit ohne Kommunikation hat die Watchdog-Zeit überschritten:

Die Antwort lautet 30 04 02 01 01 04 A4, was den Status „Watchdog“ mit Watchdog-Alarm angibt.

Dabei gilt:

30 ist die Adresse des Slave.

04 ist die verwendete Funktion. Read input registers.

02 ist die Anzahl der Byte, die gelesen werden.

01 ist das höherwertige Byte; es gibt den Status „Watchdog“ an.

01 ist das niederwertige Byte; es gibt an, dass Alarme vorliegen.

04 A4 ist die CRC.

---

### 8.1.3. ÄNDERUNGSANFORDERUNGEN DES AKTUATORS

Änderungsanforderungen des Aktuators können über das Holding Register 0x21 an Lumiplus gesendet werden. Diese Anforderungen können aber auch im Bit-Adressmodus beginnend mit Coil-Bit 0x201 gesendet werden. Beachten Sie, dass Sie entweder den Register-Adressmodus oder den Bit-Adressmodus wählen müssen. Zur Klarstellung und um Missverständnisse zu vermeiden, sollten Sie die beiden Ansätze nicht mischen, wenn Sie sich nicht vollständig über die Vorgehensweise im Klaren sind.

---

### 8.1.4. EIN-ANFORDERUNG

Der einfachste Weg zum Einschalten der Leuchten von Lumiplus über MODBUS besteht darin, Coil 0x210 auf 1 zu setzen.

Beispiel, Bit-Adressmodus:

Gesendete Nachricht: 30 0F 02 10 00 01 01 01 ED AE

Dabei gilt:

30 ist die Adresse des Slave.

0F ist die verwendete Funktion. Write multiple coils.

02 10 ist die Adresse des ersten Coils, der festgelegt wird.

00 01 ist die Anzahl der festzulegenden Coils, in diesem Fall 1.

01 ist die Anzahl der Datenbyte, die gesendet werden.

01 dient zum Festlegen der Anforderung und anschließenden Einschalten der Leuchten.

ED AE ist die CRC.

Beispiel, Byte-Modus, Holding Register:

Gesendete Nachricht: 30 10 00 21 00 01 02 00 01 38 B0

Dabei gilt:

30 ist die Adresse des Slave.  
10 ist die verwendete Funktion. Write multiple registers.  
00 21 ist die Adresse des ersten Holding Register, das festgelegt wird.  
00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.  
02 ist die Anzahl der Datenbyte, die gesendet werden.  
00 01 sind die gesendeten Daten.  
38 B0 ist die CRC.

---

### 8.1.5. AUS-ANFORDERUNG

Der einfachste Weg zum Ausschalten der Leuchten von Lumiplus über MODBUS besteht darin, Coil 0x211 auf 1 zu setzen.

Beispiel:

Gesendete Nachricht: 30 0F 02 11 00 01 01 01 D0 6E

Dabei gilt:

30 ist die Adresse des Slave.  
0F ist die verwendete Funktion. Write multiple coils.  
02 11 ist die Adresse des ersten Coils, der festgelegt wird.  
00 01 ist die Anzahl der festzulegenden Coils, in diesem Fall 1.  
01 ist die Anzahl der Datenbyte, die gesendet werden.  
01 dient zum Festlegen der Anforderung und anschließenden Ausschalten der Leuchten .  
D0 6E ist die CRC.

Eine AUS-Anforderung ist auch möglich, indem das Holding Register 0x21 auf 0x02 gesetzt wird.

---

### 8.1.6. ÄNDERN DER FARBE

Um die Farbe von Lumiplus über Modbus zu ändern, sind zwei Schritte erforderlich. Diese Schritte sind in der vordefinierten Farbe/Sequenz für Watchdog und Start nicht erforderlich.

- 1. Die vordefinierte Farbe muss ausgewählt werden.**
- 2. Es muss eine Anforderung zum Aktualisieren der vordefinierten Farbe gestellt werden.**

**Auswahl der vordefinierten Farbe:**

Die Farbe wird anhand des Holding Register für Anforderungen (0x25) gewählt.

Es kann unter 12 vordefinierten Farben ausgewählt werden. Von 0x1 bis 0xC (von 1 bis 12), entsprechend der Tabelle oben.

FARBNUMMER	FARBE
1	Rot
2	Grün
3	Blau
4	Gelb
5	Cyan
6	Magenta

FARBNUMMER	FARBE
7	Hellviolett
8	Hellblau
9	Orange
10	Rosa
11	Smaragdgrün
12	Weiß

Beispiel:

Gesendete Nachricht: 30 10 00 25 00 01 02 00 01 39 34

Dabei gilt:

- 30 ist die Adresse des Slave.
- 10 ist die verwendete Funktion. Write multiple registers.
- 00 25 ist die Adresse des ersten Holding Register, das festgelegt wird.
- 00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.
- 02 ist die Anzahl der Datenbyte, die gesendet werden.
- 00 01 ist die gesendete Farbe (Rot).
- 39 34 ist die CRC.

#### **Anfordern der Aktualisierung der vordefinierten Farbe:**

Eine Aktualisierung der vordefinierten Farbe kann anhand des Bit-Adressmodus des Holding Register für Anforderungen 0x21 angefordert werden. Die einfachste Möglichkeit besteht darin, Coil 0x213 auf 1 festzulegen.

Gesendete Nachricht: 30 0F 02 13 00 01 01 01 A9 AE

Dabei gilt:

- 30 ist die Adresse des Slave.
- 0F ist die verwendete Funktion. Write multiple coils.
- 02 13 ist die Adresse des ersten Coils, der festgelegt wird.
- 00 01 ist die Anzahl der festzulegenden Coils, in diesem Fall 1.
- 01 ist die Anzahl der Datenbyte, die gesendet werden.
- 01 ist die Anforderung zur Aktualisierung.
- A9 AE ist die CRC.

Eine Aktualisierung der vordefinierten Farbe kann auch angefordert werden, indem das Holding Register 0x21 auf 0x08 gesetzt wird.

### 8.1.7. ÄNDERN DER SEQUENZ

Zum Ändern der Sequenz für Lumiplus über Modbus sind zwei Schritte erforderlich.

1. **Die vordefinierte Sequenz muss ausgewählt werden.**
2. **Es muss eine Anforderung zum Aktualisieren der vordefinierten Sequenz und Geschwindigkeit gestellt werden.**

#### **Auswahl der vordefinierten Sequenz:**

Die Sequenz wird anhand des Holding Register für Anforderungen 0x26 gewählt.

Es kann unter 8 vordefinierten Sequenzen ausgewählt werden. Von 0x1 bis 0x8 (von 1 bis 8).

Beispiel: Ändern der Sequenz zu 8.

Gesendete Nachricht: 30 10 00 26 00 01 02 00 08 F9 01

Dabei gilt:

30 ist die Adresse des Slave.

10 ist die verwendete Funktion. Write multiple registers.

00 26 ist die Adresse des ersten Holding Register, das festgelegt wird.

00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.

02 ist die Anzahl der Datenbyte, die gesendet werden.

00 08 ist die Sequenz, die gesendet wird (die Sequenz 8).

F9 01 ist die CRC.

#### **Anfordern der Aktualisierung der vordefinierten Sequenz und Geschwindigkeit:**

Eine Aktualisierung der vordefinierten Sequenz und Geschwindigkeit kann anhand des Bit-Adressmodus des Holding Register für Anforderungen 0x21 angefordert werden, indem Coil 0x214 auf 1 gesetzt wird.

Gesendete Nachricht: 30 0F 02 14 00 01 01 01 1C 6E

Dabei gilt:

30 ist die Adresse des Slave.

0F ist die verwendete Funktion. Write multiple coils.

02 14 ist die Adresse des ersten Coils, der festgelegt wird.

00 01 ist die Anzahl der festzulegenden Coils, in diesem Fall 1.

01 ist die Anzahl der Datenbyte, die gesendet werden.

01 ist die Anforderung zur Aktualisierung.

1C 6E ist die CRC.

Eine Aktualisierung der vordefinierten Sequenz und Geschwindigkeit kann auch angefordert werden, indem das Holding Register 0x21 auf 0x10 gesetzt wird.

---

### 8.1.8. ÄNDERN DER GESCHWINDIGKEIT

Zum Ändern der Geschwindigkeit für Lumiplus über Modbus sind zwei Schritte erforderlich.

1. **Die vordefinierte Geschwindigkeit muss ausgewählt werden.**
2. **Es muss eine Anforderung zum Aktualisieren der vordefinierten Sequenz und Geschwindigkeit gestellt werden.**

#### **Auswahl der vordefinierten Geschwindigkeit:**

Die Geschwindigkeit wird anhand des Holding Register für Anforderungen (0x27) gewählt.

Es kann unter 8 vordefinierten Geschwindigkeiten ausgewählt werden. Von 0x1 bis 0x8 (von 1 bis 8).

Beispiel: Ändern der Geschwindigkeit zu 8.

Gesendete Nachricht: 30 10 00 27 00 01 02 00 08 F8 D0

Dabei gilt:

30 ist die Adresse des Slave.

10 ist die verwendete Funktion. Write multiple registers.

00 27 ist die Adresse des ersten Holding Register, das festgelegt wird.

00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.

02 ist die Anzahl der Datenbyte, die gesendet werden.

00 08 ist die gesendete Geschwindigkeit (Geschwindigkeit 8).

F8 D0 ist die CRC.

#### **Anfordern der Aktualisierung der vordefinierten Sequenz und Geschwindigkeit:**

Eine Aktualisierung der vordefinierten Sequenz und Geschwindigkeit kann anhand des Bit-Adressmodus des Holding Register für Anforderungen 0x21 angefordert werden. Die einfachste Möglichkeit besteht darin, Coil 0x214 auf 1 festzulegen.

Gesendete Nachricht: 30 0F 02 14 00 01 01 01 1C 6E

Dabei gilt:

30 ist die Adresse des Slave.

0F ist die verwendete Funktion. Write multiple coils.

02 14 ist die Adresse des ersten Coils, der festgelegt wird.

00 01 ist die Anzahl der festzulegenden Coils, in diesem Fall 1.

01 ist die Anzahl der Datenbyte, die gesendet werden.

01 ist die Anforderung zur Aktualisierung.

1C 6E ist die CRC.

Eine Aktualisierung der vordefinierten Sequenz und Geschwindigkeit kann auch angefordert werden, indem das Holding Register 0x21 auf 0x10 gesetzt wird.

---

### 8.1.9. RUHEZEIT

Lumiplus verfügt über einen Ruhemodus, der konfiguriert werden kann. Nach der konfigurierten Zeit schaltet Lumiplus alle Leuchten aus. Der Benutzer kann eine der 8 konfigurierten Ruhezeiten auswählen. Die Auswahl einer davon abweichenden Zeit ist nicht möglich. Wenn 0 festgelegt ist, ist der Ruhemodus nicht aktiviert.

Es kann unter 8 vordefinierten Zeiten ausgewählt werden. Von 0x1 bis 0x8 (von 1 bis 8):

0	Kein Ruhemodus.
1	5 Minuten.
2	15 Minuten.
3	30 Minuten.
4	60 Minuten.
5	90 Minuten.
6	120 Minuten.
7	240 Minuten.
8	480 Minuten.

Beispiel, Ändern der Ruhezeit zu 1 (5 Minuten):

Gesendete Nachricht: 30 10 00 28 00 01 02 00 01 38 29

Dabei gilt:

- 30 ist die Adresse des Slave.
- 10 ist die verwendete Funktion. Write multiple registers.
- 00 28 ist die Adresse des ersten Holding Register, das festgelegt wird.
- 00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.
- 02 ist die Anzahl der Datenbyte, die gesendet werden.
- 00 01 ist die gesendete Zeit (Zeit 1, 5 Minuten).
- 38 29 ist die CRC.

---

### 8.1.10. ÜBERPRÜFUNG DES REGISTERS DER SELBSTHALTENDEN ALARME

Es kann überprüft werden, welche Alarme aktiviert wurden, seit die Stromversorgung zum letzten Mal unterbrochen oder der Alarm zurückgesetzt wurde. Es ist nur der Watchdog-Alarm implementiert. Um ihn zu überprüfen, muss das Holding Register 0x20 gelesen werden. Bit 0 bis 14 werden nicht verwendet, können aber gelesen werden. Bit 15 ist der Watchdog-Alarm. Der Wert 1 bedeutet, dass der Alarm aktiviert ist.

Wenn der Strom ausfällt, werden alle Alarme auf 0 gesetzt.

Beispiel: Lesen von selbsthaltenden Alarmen mit Holding Registers, wenn der Watchdog-Alarm aktiviert wurde

Empfangene Nachricht: 30 03 02 80 00 A4 40

Dabei gilt:

- 30 ist die Adresse des Slave.
- 03 ist die verwendete Funktion. Read Holding Registers.
- 02 ist die Anzahl der Datenbyte, die gelesen werden.
- 80 ist das höherwertige Byte gelesener Daten; es gibt einen Watchdog-Fehler an.
- 00 ist das niederwertige Byte der gelesenen Daten; es gibt an, dass im niederwertigen Byte keine Alarme vorliegen.
- A4 40 ist die CRC.

Selbsthaltende Alarme können auch im Bit-Modus mit Coils geprüft werden. Die Bit-Interpretation dieses Fehlers wird unten gezeigt. Die Coils 0x200 bis 0x20E werden nicht verwendet, können aber ohne Einschränkung gelesen werden:

Coil 0x20F      MODBUS Watchdog

Beispiel: Lesen von selbsthaltenden Alarmen mit Coils, wenn der Watchdog-Alarm aktiviert wurde:

Die empfangene Nachricht lautet: 30 01 01 01 9E B4

Dabei gilt:

- 30 ist die Adresse des Slave.
- 01 ist die verwendete Funktion. Read Coils.
- 01 ist die Anzahl der Datenbyte, die gelesen werden.
- 01 bedeutet, dass der Alarm aktiviert ist.
- 9E B4 ist die CRC.

Die Alarme werden zurückgesetzt, indem das Holding Register 0x20 auf 0 gesetzt wird.

Beispiel, Zurücksetzen aller Alarme:

Gesendete Nachricht: 30 10 00 20 00 01 02 00 00 F8 A1

Dabei gilt:

- 30 ist die Adresse des Slave.
- 10 ist die verwendete Funktion. Write multiple registers.
- 00 20 ist die Adresse des ersten Holding Register, das festgelegt wird.
- 00 01 ist die Anzahl der festzulegenden Holding Registers, in diesem Fall 1.
- 02 ist die Anzahl der Datenbyte, die gesendet werden.
- 00 00 sind die gesendeten Daten. (0, um die Alarme auf 0 festzulegen).
- F8 A1 ist die CRC.

---

### 8.1.11. ÜBERPRÜFUNG DES REGISTERS DER SOFORTALARME

Wenn Fehler vorliegen, wird das niederwertigste Bit im Input Register für Status 0x00 auf 1 gesetzt.

Um detailliertere Informationen zum aufgetretenen Fehler zu erhalten, können diese beim Input Register 0x01 angefordert oder aus Digital Input 0x010 bis 0x01F abgelesen werden. Da in diesem Fall nur der Watchdog-Alarm implementiert ist, ist Digital Input 0x01F der einzige Alarm, der zu 1 wechseln kann. (Dies entspricht dem Watchdog-Alarm.)

Die Bit-Interpretation des Fehlerregisters Input Register 0x01 wird unten gezeigt. Bit 0 bis 14 werden nicht verwendet, können aber ohne Einschränkung gelesen werden. Bit 15 zeigt den Watchdog MODBUS-Fehler.

## 9. GRUNDLEGENDE MODBUS-RTU-REGISTERZUORDNUNG

Die Tabelle in diesem Abschnitt enthält unsere exklusive Original-Registerzuordnung mit dem Namen der Funktion und ihrer Adresse.

Neben der Registerzuordnung selbst ist es wichtig, die Beziehung zwischen Fehlern (Holding Register 0x20) und Sofortalarmen (Input Register 0x01) zu verstehen.

Es besteht auch eine direkte Beziehung zwischen Holding Register 0x20 und Input Register 0x01. Holding Register 0x20 gibt die Gesamtzahl der selbsthaltenden Alarme an, Input Register 0x01 den aktuellen Alarm. Diese Beziehung gilt auch für die Bit-zu-Bit-Beziehung zwischen Registern. Da nur der Watchdog-Alarm implementiert ist, stimmen selbsthaltende Alarme und Sofortalarme immer überein.

Um die Alarme zurückzusetzen, muss das Holding Register 0x20 zurückgesetzt werden, nicht das Input Register 0x01. Das Input Register 0x01 wird zurückgesetzt, wenn der aktuelle Fehleralarm verschwindet. Um alle Alarme zurückzusetzen, müssen Coil 0x200 bis 0x20F auf 0 gesetzt werden. Hinweis: Bei einer Stromunterbrechung werden alle selbsthaltenden Alarme zurückgesetzt.

Name	Holding Registers	Input Registers	Coils	Eingänge	Beschreibung
<b>ID_Address</b>	0x00				Dieser Parameter ist die Adresse des LumiPlus, standardmäßig 0x30. Wenn in der Anlage mehr als ein Gerät mit dem Bus verbunden ist, muss die ID_Address des Geräts geändert werden, bevor ein anderes angeschlossen wird.
<b>COM_Setup</b>	0x01				Dieser Parameter bestimmt die Auswahl der Baud-Geschwindigkeit. Es kann unter sechs möglichen Konfigurationen ausgewählt werden: Werkseitige Konfiguration: 0. 9600, E81 Unterstützte Werte: 1. 19200, 8E1 2. 9600, 8N2 3. 19200, 8N2 4. 9600, 8N1 5. 19200, 8N1
<b>ID_Manufacturer_hi</b>	0x02				Dieser Parameter gibt das höherwertige Byte an, das den Herstellercode darstellt.
<b>ID_Manufacturer_lo</b>	0x03				Dieser Parameter gibt das niederwertige Byte an, das den Herstellercode darstellt.
<b>ID_Product_code_hi</b>	0x04				Dieser Parameter gibt das höherwertige Byte an, das den Produktcode darstellt.
<b>ID_Product_code_lo</b>	0x05				Dieser Parameter gibt das niederwertige Byte an, das den Produktcode darstellt.
<b>Reserved</b>	0x06				Reserviert.
<b>HW_Version</b>	0x07				Dieser Parameter gibt die Hardwareversion von LumiPlus an.
<b>SW_Version</b>	0x08				Dieser Parameter gibt die Softwareversion von LumiPlus an.
<b>MODEL_Serie_hi</b>	0x09				Dieser Parameter gibt das höherwertige Byte der Seriennummer an.
<b>MODEL_Serie_lo</b>	0x0A				Dieser Parameter gibt das niederwertige Byte der Seriennummer an.
<b>MODEL_Production_hi</b>	0x0B				Dieser Parameter gibt das höherwertige Byte der Produktcharge an.
<b>MODEL_Producton_low</b>	0x0C				Dieser Parameter gibt das niederwertige Byte der Produktcharge an.

Name	Holding Registers	Input Registers	Coils	Eingänge	Beschreibung
<b>watchdog_time</b>	0x10				Dieser Parameter gibt die Zeit in Sekunden an, die ab der letzten Kommunikation verstreichen kann, bevor der Watchdog aktiviert wird. Bei 0 wird er deaktiviert. Bereich: 0-65535.
<b>watchdog_config</b>	0x11				Dieser Parameter definiert, wie der Watchdog reagiert, wenn die Kommunikation ausfällt.
<b>Status, wenn der Watchdog-Wert überschritten wird</b>	niederwertiges Byte				Das <b>niederwertige Byte</b> wird nicht berücksichtigt. Es muss 0 betragen.
<b>Konfiguration des Betriebsmodus</b>	höherwertiges Byte				Das <b>höherwertige Byte</b> ist die Konfiguration des Betriebsmodus.  0 = definiert in der vordefinierten Farbe/Sequenz des Watchdogs. 1 = Neukonfiguration des Moduls.
<b>Vordefinierte Farbe/Sequenz des WDT</b>	0x14				Dieser Parameter definiert die Farbe oder Sequenz, die bei Aktivieren des Watchdogs aktiviert wird. 0xFFFF behält die Farbe/Sequenz wie vor dem Alarmstatus bei.
<b>Farbe/Sequenz</b>	niederwertiges Byte				Das <b>niederwertige Byte</b> ist die Farbe, wenn die Geschwindigkeit 0 beträgt, bzw. die Sequenz, wenn die Geschwindigkeit einen anderen Wert als 0 hat.
<b>Geschwindigkeit</b>	höherwertiges Byte				Das <b>höherwertige Byte</b> : Geschwindigkeit: 0 = feste Farbe; 1 bis 8 = Sequenz.
<b>Vordefinierte Farbe/Sequenz für Start</b>	0x15				Dieser Parameter definiert die Farbe oder Sequenz des LumiPlus beim Start. 0xFFFF behält die Farbe/Sequenz wie vor dem letzten Ausschalten bei.
<b>Farbe/Sequenz</b>	niederwertiges Byte				Das <b>niederwertige Byte</b> ist die Farbe, wenn die Geschwindigkeit 0 beträgt, bzw. die Sequenz, wenn die Geschwindigkeit einen anderen Wert als 0 hat.
<b>Geschwindigkeit</b>	höherwertiges Byte				Das <b>höherwertige Byte</b> : Geschwindigkeit: 0 = feste Farbe; 1 bis 8 = Sequenz.

Name	Holding Registers	Input Registers	Coils	Eingänge	Beschreibung
<b>Status</b>	0x00				Dieser Parameter gibt an, in welchem Status sich LumiPlus befindet und ob ein Alarm vorliegt.
<b>Alarm ON (eingeschaltet)</b>	Niederwertiges Byte .bit 0 .bit 1 .bit 2...7			0x00 0x01	Das niederwertige Byte gibt den Alarm an und ob eine Leuchte eingeschaltet ist. Es wurde ein Alarm aktiviert (falls 1). Es wurde eine Leuchte eingeschaltet (falls 1). Werden nicht verwendet.
<b>0: Status Start</b> <b>1: Status Watchdog</b> <b>2: Aktualisierungsanforderung für vordefinierte Farbe</b> <b>3: Aktualisierungsanforderung für Sequenz und Geschwindigkeit</b>		Höherwertiges Byte			Das <b>höherwertige Byte</b> gibt den aktuellen Status an.
<b>Selbsthaltender Alarm</b>  Watchdog	0x20  .bit 1...14 .bit 15		0x20F		Dieser Parameter gibt an, welcher Alarm aktiviert wurde.  Werden nicht verwendet.
<b>Sofortalarm</b> Watchdog		0x01 .bit 1...14 .bit 15		0x01F	Dieser Parameter gibt die Sofortalarme an. Werden nicht verwendet.
<b>Aktuatoren 0x00</b> Anforderung ON (aktiviert) Anforderung OFF (deaktiviert)  Aktualisierungsanforderung für vordefinierte Farbe	0x21 .bit 0 .bit 1 .bit 2 .bit 3 .bit 4 .bit 5...15		0x210 0x211 0x212 0x213 0x214		Diese Parameter sind die möglichen Anforderungen.  Nicht reserviert.  Werden nicht verwendet.

Name	Holding Registers	Input Registers	Coils	Eingänge	Beschreibung
<b>Vordefinierte Farbe</b>	0x25				Dieser Parameter bestimmt die Lichtfarbe. Nach Auswahl ist eine Aktualisierungsanforderung für die Farbe erforderlich, um die Farbänderung zu sehen. Die Farben 1 - 12 stehen zur Verfügung.
<b>Vordefinierte Sequenz</b>	0x26				Dieser Parameter bestimmt die Lichtsequenz. Nach Auswahl ist eine Aktualisierungsanforderung für die Sequenz erforderlich, um die Sequenzänderung zu sehen. Die Sequenzen 1 - 8 stehen zur Verfügung.
<b>Sequenzgeschwindigkeit</b>	0x28				Dieser Parameter bestimmt die Geschwindigkeit der Lichtvariation. Nach Auswahl ist eine Aktualisierungsanforderung für die Sequenz und Geschwindigkeit erforderlich, um die Geschwindigkeitsänderung zu sehen. Die Geschwindigkeiten 1 - 8 stehen zur Verfügung.
<b>Ruhezeit</b>	0x28				Dieser Parameter bestimmt die Zeit, die bis zum Ausschalten der Leuchte vergeht. Es ist keine Aktualisierung der Anforderung erforderlich. Die Zeit bis zum Ausschalten reicht von 1 bis 8.

## 10. PRODUKTVERSION

Handbuch V1.0: Alle Informationen in diesem Handbuch beschreiben das Verhalten der Hardwareversion 1 und der Softwareversion 1.

## 11. FEHLERBEHEBUNGSLEITFADEN

Problem	Lösung
<p>Der Master erhält keine Antwort auf die Anforderungen.</p>	<ul style="list-style-type: none"> <li>a) Überprüfen Sie die Kabelpolarität A,B.</li> <li>b) Überprüfen Sie die Erdungsverbindung zwischen Master und Slave.</li> <li>c) Prüfen Sie bei einem langen Buskabel die Abschlusswiderstände entsprechend Kapitel 4 des Handbuchs.</li> <li>d) Vergewissern Sie sich, dass die Baudrate und die Bit-Einstellungen für Master und Slaves gleich sind.</li> <li>e) Vergewissern Sie sich, dass der Master die CRC entsprechend dem Modbus-Standard beim Senden einschließt. Prüfen Sie die Berechnung der CRC.</li> <li>f) Überprüfen Sie, dass die angeforderte Funktion vom Slave unterstützt wird.</li> <li>g) Prüfen Sie, ob die Slave-Adresse diejenige ist, die Sie in der Anforderung verwenden. Wenn Sie sich nicht sicher sind, isolieren Sie den Slave (verbinden Sie ihn direkt mit dem Master) und führen Sie einen „Erkennungs“-Prozess im Master aus. Wenn der Master diese Art Prozess nicht automatisch durchführt, überprüfen Sie die Slave-Adressen manuell von 1 bis 255 mit den verschiedenen Baudraten und Bit-Konfigurationen, die vom Slave unterstützt werden.</li> </ul>
<p>Der Slave führt keine Anforderung aus.</p>	<ul style="list-style-type: none"> <li>a) Überprüfen Sie die Kabelpolarität A,B.</li> <li>b) Überprüfen Sie die Erdungsverbindung zwischen Master und Slave.</li> <li>c) Prüfen Sie bei einem langen Buskabel die Abschlusswiderstände entsprechend Kapitel 4 des Handbuchs.</li> <li>d) Vergewissern Sie sich, dass die Baudrate und die Bit-Einstellungen für Master und Slaves gleich sind.</li> <li>e) Vergewissern Sie sich, dass der Master die CRC entsprechend dem Modbus-Standard beim Senden einschließt. Prüfen Sie die Berechnung der CRC.</li> <li>f) Überprüfen Sie, dass die angeforderte Funktion vom Slave unterstützt wird.</li> <li>g) Prüfen Sie, ob die Slave-Adresse diejenige ist, die Sie in der</li> </ul>

	<p>Anforderung verwenden. Wenn Sie sich nicht sicher sind, isolieren Sie den Slave (verbinden Sie ihn direkt mit dem Master) und führen Sie einen „Erkennungs“-Prozess im Master aus. Wenn der Master diese Art Prozess nicht automatisch durchführt, überprüfen Sie die Slave-Adressen manuell von 1 bis 255 mit den verschiedenen Baudraten und Bit-Konfigurationen, die vom Slave unterstützt werden.</p>
<p>Der Slave führt einige Anforderungen nicht aus oder die „Nachrichtenfehlerrate“ ist zu hoch.</p>	<p>a) Vergewissern Sie sich, dass Sie ein gedrilltes und abgeschirmtes Kabel gemäß EN50170 verwenden (besonders mit langen Buskabeln oder in Umgebungen mit hoher elektromagnetischer Kontamination, z. B. in der Nähe von Wechselrichtern oder Schaltnetzteilen).</p> <p>b) Überprüfen Sie, ob die Kabelabschirmung bei Master und Slaves mit der Erdung verbunden ist.</p> <p>c) Prüfen Sie die Abschlusswiderstände entsprechend Kapitel 4 dieses Handbuchs.</p> <p>d) Konfigurieren Sie eine niedrigere Baudrate in Master und Slaves.</p>
<p>Unerwartete WatchDog-Fehler</p>	<p>a) Überprüfen Sie, ob die „Nachrichtenfehlerrate“ des Bus zu hoch ist.</p> <p>b) Überprüfen Sie, ob der Bus mit Nachrichten überlastet ist. In diesem Fall könnte die Zeit zwischen zwei Nachrichten an den gleichen Slave größer als die im Slave konfigurierte WatchDog-Zeit sein. Erhöhen Sie die WatchDog-Zeit im Slave oder die Baudrate des Bus (Master und Slave).</p>

Handbuch V2.0: Alle Informationen in diesem Handbuch beschreiben das Verhalten der Hardwareversion 1 und der Softwareversion 1.