MANUAL

Space



FLEXLOG

Version: 1.6.0

Contents

1	**Introduction**		**5**
2	**General Information**		**6**
	2.1	Use of documentation	6
	2.2	Liability and guarantee	6
	2.3	Contact	6
	2.4	Release history	6
3	**Safety instructions**		**8**
	3.1	Used symbols and notice	8
	3.2	Qualification of personnel	9
	3.3	Intended use	9
	3.4	Functional safety techniques	10
4	**Product description**		**11**
	4.1	Use and Application	11
	4.2	Technical data and dimensions	11
	4.3	LED display	11
	4.4	Connections	13
5	**Transportation, storage and disposal**		**14**
	5.1	Transportation	14
	5.2	Storage	14
	5.3	Disposal	14
6	**Installation**		**16**
	6.1	Mechanical assembly	16
	6.2	Connections	17
	6.2.1	Digital I/O	17
	6.2.2	Ethernet	18
	6.2.3	Sensors	18
	6.2.4	Sensorplug 2	19
	6.2.5	Power supply	19
	6.3	Electrical installation	20

0 Contents

7	**Commissioning and settings**		**22**
	7.1	Web interface	22
	7.1.1	Status	24
	7.1.2	Product information	24
	7.1.3	System	25
	7.1.4	WiFi	25
	7.1.5	Sensors	26
	7.1.6	Neighborhood discovery	26
	7.1.7	Identification of the module in the field	26
	7.1.8	Manual mode	27
	7.2	Configuration	27
	7.2.1	General	27
	7.2.2	Network settings	28
	7.2.3	SSL certificate	29
	7.2.4	Sensors	29
	7.2.5	Temperature	30
	7.3	Lua scripts	30
	7.3.1	System scripts	30
	7.3.2	User-defined scripts	38
	7.4	Protocols	39
	7.4.1	REST API	39
	7.4.2	NTP	40
	7.4.3	MQTT	41
	7.4.4	Keep Alive	41
	7.5	User	41
	7.6	CFS+	41
	7.7	Reset to factory settings and restart	43
8	**Update firmware version**		**44**
	8.1	Determine current version	44
	8.2	Updating	44
9	**Repair, maintenance and cleaning**		**45**
	9.1	Replacement	45
	9.2	Error handling	45
	9.3	Cleaning	45

0 Contents

10	**Troubleshooting**		**46**
	10.1	Error table	46
	10.1.1	Overtemperature	46

1 Introduction

The **SPA01 module** is referred to below as **SPA01**, **Space** or **Module**.

The Wifi-capable Space control module with integrated antenna enables wireless networking of individual I/O field components. The compact design, classified with protection class IP54, enables use in industrial environments and guarantees simple commissioning.

2 General Information

2.1 Use of documentation

The documentation is an integral part of the product and contains important information on operation and service. It is intended for all persons who carry out assembly, installation, commissioning and service work on the product.

2.2 Liability and guarantee

Defect liability

Compliance with the documentation is a prerequisite for trouble-free operation and the fulfillment of any warranty claims. After receiving the product, check it for damage. In the event of material damage, notify the post office or forwarding agent and inform the supplier.

Disclaimer

Observance of this documentation and the documentation for the connected devices is a basic prerequisite for safe operation and for achieving the specified product properties and performance characteristics. For personal injury, property damage or financial loss resulting arising from non-compliance with the operating instructions, flexlog GmbH accepts no liability. Liability for material defects is excluded in such excluded in such cases.

2.3 Contact

flexlog GmbH
Phone: +49 721 7540350
Mail: support@flexlog.de

2.4 Release history

Version	Revision	Responsible
1.6.0	A	C. Hoge

2 General Information

Copyright note

©2024 - flexlog GmbH. All rights reserved.

3 Safety instructions

Safety is paramount when using and installing our products. This section provides important information and guidelines to ensure safe handling of the product. By following thesesafety instructions, accidents and damage can be avoided.

3.1 Used symbols and notice

The following symbols and notices in accordance with DIN 82079-1 are used in this documentation. They warn of possible personal injury or damage to property or provide information on work steps.

Symbol	Meaning
	DANGER - Immediate danger - Death or serious injury
	WARNING - Potential danger - Death or serious injury
	CAUTION - Potential danger - moderate or minor injuries
	ATTENTION - Possible material or product damage
	NOTICE - Facts to be observed in the current step of the instructions

The security advices devided into different categories to provide specific and relevant information for different aspects of the use and handling of the product. In this chapter you will information on the qualification of personnel, the intended use of the product and the functional safety techniques. Specific information on transportation, storage, assembly, electrical installation, commissioning, operation and disposal of the product is provided in the corresponding chapters. Each of these categories contains important information that you must observe to ensure safety throughout the entire life cycle of the product.

3.2 Qualification of personnel

Unqualified personnel are often unable to assess dangers correctly and are therefore exposed to greater risks.



Only authorise qualified personnel to carry out the tasks described in these operating instructions.



The operator is responsible for ensuring that the personnel know and comply with the local regulations and rules for safe working.

This manual is aimed at the following target groups:

Operating personnel

The operating personnel are trained in handling and cleaning the Space and adhere to the safety regulations.

Service staff

The service personnel have completed technical training or training from the manufacturer and carry out maintenance and repair work.

Qualified electrican

A qualified electrician has technical training and, based on their expertise and experience as well as their knowledge of the relevant regulations to carry out work on electrical systems professionally. They are able to independently recognise dangers and prevent personal injury and property damage caused by electrical voltage.

All work on the electrical equipment may only be carried out by a qualified electrician.

3.3 Intended use

The Space Module from flexlog GmbH may only be used in industrial environments for industrial purposes within the the defined performance limits specified in the technical data.

3.4 Functional safety techniques

Unless expressly stated otherwise in the documentation, the product is not intended to fulfil safety functions on its own. It may only be operated in conjunction with higher-level safety systems in order to ensure the required functional safety.

4 Product description

4.1 Use and Application

The Space module is used to integrate control and conveyor technology areas into a WLAN network. Here Space can function both as a client of an existing WLAN network or as an access point for other WLAN participants. Network-bound participants or areas can be connected to Space via an Ethernet interface. The additional connections for digital inputs/outputs and two relays allow further applications such as querying or controlling decentrally connected switches and lights. The module can also communicate via protocols such as MQTT, NTP, REST and FlexNet. It is protected against dust and splash water in accordance with IP54.

Detailed configurations can be made via a user-friendly web interface. With its CFS+ configuration tool, flexlog GmbH also offers a convenient way of updating the modules and reading out all relevant data. CFS+ also offers the option of exporting the current configuration of a module as a zip file and importing it, provided the file is unchanged, to another module with the same product variant.

4.2 Technical data and dimensions

Power supply	
Voltage	24 V DC
Tolerance input voltage	±10%
Power consumption (excl. peripherals)	max. 1 A
Ethernet	
Speed	10/100 Mbits/s full-duplex
Standard	IEEE 802.11n
Wi-Fi	2.4/ 5 GHz
Security standard	WPA2

4.3 LED display

Various system statuses can be displayed using the integrated LED strip.

Category	Description
Dimensions with socket (mm)	131 ×24×70
Dimensions with socket (mm)	117 ×24×70
Weight (g)	140
Housing material	NORYL N190 black
Installation	Retaining clip in the side profile or screw connection
Protection class	IP54
Temperature (In Use)	0 °C bis 45 °C
Temperature (Storage)	−20 °C bis 60 °C
Air humidity	10% bis 95% (without condensation)
Processor	Cortex-A72, 4x 1.5 GHz
Operating system	Linux
Protocols	HTTP, TCP/IP, IPv4, IPv6, REST ful-API, FlexNet

Table 4.1: General product properties

Operating status	Brightness	Visualisation	Flashing status (on/off in ms)
Normal operation	dark	on	-
Update mode	bright	slow flashing	3000 / 3000
General failure	bright	fast flashing	100 / 100
Boot mode	-	off	-
Module identification	bright	pulsed flashing	500 / 500

Table 4.2: LED status

The module also has LEDs as standard for the power supply, Ethernet ports, Wi-Fi status and sensor and digital I/O connections, which signal activity and trigger status respectively.

The positions and meaning of the LEDs are visible and described in the section electrical installation.

4.4 Connections

	Quantity
Power supply (M8 socket 4-pin, A-coded)	1x
Sensor 1 (M8 socket 4-pin, A-coded)	1x
Sensor 2 (M8 socket 4-pin, A-coded, with relay contacts)	1x
Ethernet (M8 socket 4-pin, D-coded)	1x
I/O Port (M8 socket 8-pin, A-coded)	1x

Table 4.3: Overview of the connections of the Space modules

5 Transportation, storage and disposal

5.1 Transportation

Please observe the following instructions when transporting the product:

- Check the delivery for any transport damage immediately upon receipt. Inform the transport company immediately of any damage found. If the product is damaged, it must not be assembled, installed or put into operation.
- Heavy impacts should be avoided during transport. If necessary, use suitable and sufficiently dimensioned means of transport.
- After transport, each Space should be checked for visible damage. Take photos of damaged parts and, if necessary, inform the carrier or flexlog immediately in order to secure possible claims for compensation.

.

5.2 Storage



Damage due to improper storage
Module can become unusable due to improper storage

- Ensure correct storage of the Space.

Always store the module in as dry and clean an environment as possible. Observe the permissible conditions.

5.3 Disposal

Dispose of the product and all its parts in accordance with their materials and national regulations. If possible, recycle the product or contact a specialised disposal company. If possible, separate the product into the following material categories:

- Iron, steel or cast iron
- Stainless steel
- Magnets
- Aluminium
- Copper
- Elektronic components
- Plastics

Disposal in accordance with WEEE Directive 2012/19/EU: In principle, the operator is responsible for the proper and environmentally friendly disposal of the products. The implementation of the WEEE Directive 2012/19/EU in national laws must be observed.

6 Installation



The operator is responsible for ensuring that the personnel know and comply with the local regulations and rules for safe working.



Avoid dropping or improper handling of the module to prevent internal damage!

6.1 Mechanical assembly



Improper handling when installing the module can lead to damage or a shortened service life of it.



- Avoid dropping or improper handling of the module to prevent internal damage.
- Check each Space for visible damage before assembly.

The module must be de-energised for installation. The module is clicked into the corresponding holder.

There are two types of holders that can be used to mount the module. The C-holder consists of retaining clips in the side profile of the conveyor section and enables the module to be simply clicked into place.



Figure 6.1: C-holder for click mounting

The S-bracket is screwed to the desired mounting location using M5 screws and then allows the module to be clicked into place.

Dismounting

To remove a module, pull it out of the holder and then remove it.

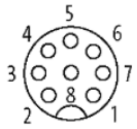
6.2 Connections

The power supply is fed through directly to the connections, which is why an appropriate back-up fuse must be installed upstream of the Space input.

6.2.1 Digital I/O

In addition to the sensors, a connection for digital inputs and outputs is provided. The outputs can each be loaded with up to 0.5 A; the outputs can be loaded with up to 3 A simultaneously. The connection is designed as an 8-pin M8 plug, A-coded.

The assignment of the I/O connection can be seen in the following illustration:



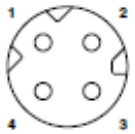
Pin / color	Signal
1 / white	Input/Output
2 / brown	24 V
3 / green	Input/Output
4 / yellow	Input/Output
5 / gray	Input/Output
6 / pink	Input/Output
7 / blue	GND
8 / red	Input/Output

Table 6.1: Contact assignment Digital I/O (female)

6.2.2 Ethernet

The Ethernet interfaces of the module can be used for control, networking and configuration. The interfaces are designed as a 4-pin M8 socket, D-coded.

The assignment can be seen in the following illustration:



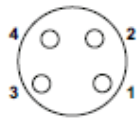
Pin / color	Signal
1 / yellow	TX data+
2 / white	RX data+
3 / orange	TX data-
4 / blue	RX data-

Table 6.2: Contact assignment ethernet (female)

6.2.3 Sensors

The module provides a connection for a sensor. The connection is designed as a 4-pin M8 socket, A-coded. The outputs can each be loaded with up to 0.5 A; the outputs can be loaded with up to 3 A simultaneously.

The assignment can be seen in the following illustration:



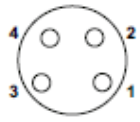
Pin / color	Signal
1 / brown	24 V
2 / white	Input/Output 2
3 / blue	GND
4 / black	Input/Output 1

Table 6.3: Contact assignment sensors (female)

6.2.4 Sensorplug 2

The module provides a connection for a total of two relays. The connection is designed as a 4-pin M8 socket, A-coded. The connection may be loaded with a maximum of 4 A. The switchable voltage is limited to 24 V.

The assignment can be seen in the following illustration:



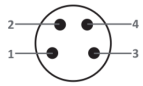
Pin / color	Signal
1 / brown	relay1_connection1
2 / white	relay1_connection2
3 / blue	relay2_connection1
4 / black	relay2_connection2

Table 6.4: Contact assignment relay sensor plug (female)

6.2.5 Power supply

The module is supplied via a 4-pin M8 plug, A-coded. The supply consists of a pair of wires for the logic supply. The power supply is limited to 4 A and must be fused externally.

The assignment can be seen in the following illustration:

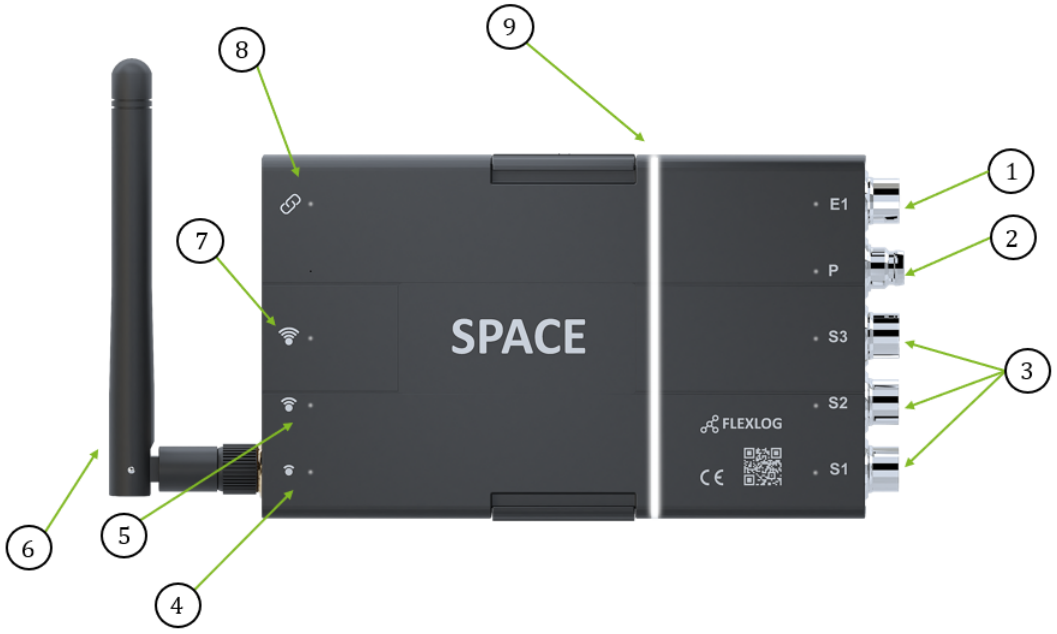


Pin / color	Signal
1 / brown	V-logic (24 V)
2 / white	-
3 / blue	GND-logic
4 / black	-

Table 6.5: Contact assignment power supply (male)

6.3 Electrical installation

Make sure that all the necessary covers are properly in place after the electrical installation. The protective measures and protective devices must comply with the applicable regulations (e.g. EN 60204-1 or EN 61800-5-1).



	Copmponents
1	Ethernet connection
2	Power supply
3	Sensor connection
4	Poor connection quality
5	Medium connection quality
6	WiFi antenna
7	Good connection quality
8	Unoccupied
9	LED Stripe

LEDs to indicate the connection quality:

- If the connection quality is good, all three connection LEDs light up.
- If the connection quality is medium, the bottom two connection LEDs light up.
- If the connection quality is poor, only the bottom connection LED lights up.

7 Commissioning and settings



Risk of injury from electrical voltage.
Moderate or minor injuries.

- Do not touch any power connections, connected cables or terminals when the power is on.



Risk of burns from hot surfaces.
Risk of injury from surface temperatures of the product above 60°C.

- Do not touch the module during operation.
- Allow the module to cool sufficiently before touching it.



Risk of property damage!
Risk of damage to the module due to disconnection during operation.

- Switch off the module before disconnecting it.

7.1 Web interface

The web interface offers the possibility to read out the motor data as well as the connected digital inputs and outputs and can be accessed at the IPv4 and IPv6 address of the module. To do this, click on the corresponding link on the module page in CFS+ or enter the IP address directly in the browser. The default IPv4 adress ex factory is ***192.168.10.1***.
The user system allows access to the web interface as a guest user, who only has read access to the status area, and as an administrator with read and write access to all areas. The default password for the administrator is admin. When the page is called up, the user is greeted with a login dialogue, if not already logged in, which can be closed to continue as a guest and log in later if necessary.
The web interface is divided into the following areas:

- Status 

- Settings 
- Lua scripts 
- Protocols 
- Users 

In addition, the information page can be accessed via the information button at the bottom left.
You can log out using the logout button .

Status

General information about the module and the directly connected neighbouring modules can be viewed on the status page, and the current status of the sensors and digital inputs can be viewed. In addition it is possible to control the digital outputs in manual mode. A more detailed description of this page can be found in the Status section of this manual.

Settings

In the area **Settings** can be used to make changes to configure the module. You can find more detailed information in the section Settings of this manual.

Lua scripts

In the area **Lua scripts** Pre-defined scripts can be parameterised and activated. Furthermore, freely programmable scripts can be created. For more detailed information, please refer to the section Lua scripts of this manual.

Protocols

The protocols with which the module can communicate can be managed in the **Protocols** area. For more detailed information, please refer to the section communication protocols of this manual.

Information page

The information page provides general information about the flexlog GmbH and the used licences.

Users

The password of the logged-in user can be changed here. You can find more detailed information in the Users section of this manual.

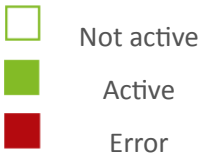
7.1.1 Status

Information about the module, the status of the network interfaces and the digital inputs and outputs, as well as the directly connected neighbouring module, is displayed here.
The status page is devided in eight areas:

- **Product information** shows general information about the product.
- **System** shows current system information.
- **WiFi** shows information on the current WiFi connection.
- **Ethernet** shows the currently set Ethernet network settings.
- **Sensors** shows the current state of the installed sensor inputs and I/Os.
- **Neighborhood discovery**shows information about directly connected neighbouring modules.
- **Module identification** enables module identification to be activated, as described in LED interface.
- **Manual mode** allows you to activate manual mode.

The status information of the module is used to better assess the current module state. It should be noted that changes in the module state are displayed with a delay.

The following coding applies to all status fields in these areas:



7.1.2 Product information

The following information can be found in the product information area:

- The **Product number** is used to clearly identify the hardware variant.
- The **Serial number** clearly identifies the module.
- The **Device name** is the name of the module. The default value is an empty string, but it can be configured by the user under Settings.
- The **HW version** provides information about the module's hardware version.

- The **SW version** provides together with the **SW buildvnumber**information about the software version of the module.
- The **MAC** shows the hardware address of the module's primary network connection.

For support requests, at least the **serial number** , the **SW version** and the **Build number** should be provided, but preferably all the product information listed here.

7.1.3 System

The system area shows the current system status. In addition, the operating time and temperature can be read out:

- **Module state** summarises all the error states of the module. This means that as soon as a component has an error, a module error is displayed here.
- **Sensor status** shows the current status of all sensors and digital I/Os .
- **Temperature** shows the current operating temperature of the module.
- **Overtemperature** indicates whether the temperature has risen above the selected threshold. This threshold can be configured under Settings or via a bus system.
- **Operation time** indicates the current operating time of the module since the last restart.

7.1.4 WiFi

Mode describes the current WiFi function. A distinction is made between deactivated, WiFi client and access point. Further information is provided depending on the selected mode.
WiFi-Client

- Connected describes whether a connection to the access point has been established.
- Connection quality describes the current quality of the connection to the access point as a percentage.
- SSID shows the name of the connected access point.
- IPv4 address shows the IPv4 address used.
- IPv4 subnet mask shows the subnet mask used.
- Default gateway shows the default gateway used.

Ethernet

- IPv4 address shows the IPv4 address used.
- IPv4 subnet mask shows the subnet mask used.
- Default gateway shows the default gateway used.

7.1.5 Sensors

The sensor area shows the current status of the sensors and relays as it is transmitted to the protocols used. Inversion at configuration level is also taken into account.

7.1.6 Neighborhood discovery

The Neighborhood discovery shows information about modules, which are connected directly to one of the Ethernet interfaces:

- **Port** indicates the module's Ethernet interface to which the neighbouring module is connected.
- **Serial number** uniquely identifies the neighbouring module.
- **Device name** is the name of the neighbouring module.
- **IPv6 address** displays the IPv6 address of the neighbouring module. Clicking on this will take you to the web interface of the neighbouring module.

7.1.7 Identification of the module in the field

Module identification allows individual modules to be localised and can be activated in two ways: module identification and reverse identification. Both can be activated or deactivated via CFS+.

- **Module identification:** When activated, the LED strip of the module flashes at a frequency of 1 Hz. In addition, the module is displayed in CFS+ with a flashing green symbol.
- **Reverse identification:** This function makes it possible to initiate module identification by activating a digital input accordingly. To do this, the digital input must receive a positive edge three times within two seconds, with the third activation remaining active for at least three seconds. It should be noted that reverse identification can lead to increased latency of the digital inputs.

Identification settings via the web interface

In the web interface, the respective identification method can be activated and deactivated using sliders in the status area.

7.1.8 Manual mode



After using the manual mode, normal operation of the module can no longer be guaranteed until the next restart.

Manual mode allows you to activate digital outputs on a **test basis**. While manual mode is active, the module's actuators cannot be controlled by any other component. Active manual mode is automatically deactivated after one hour.

7.2 Configuration

The modules are configured via Ethernet using the CFS+ configuration software developed by flexlog GmbH or the module's web interface. This makes it possible to configure the modules even without an operational system. In addition to configuration via the web interface, the module can be configured using Profinet or CFS+.

Settings

In the Settings tab, configurations can be made in the various system areas:

- General
- Network
- SSL Certification
- I/Os
- LEDs
- Temperature

7.2.1 General

In the General area, which is located in the tab under Settings  a device name can be assigned. The default value here is the MAC address. The corresponding restrictions are noted in the text below the input field. The device name can also be changed via CFS+.
In addition, a module restart and a factory reset can be instructed here. You can find more information on this in the chapter Reset factory settings and restart.

7.2.2 Network settings

Ethernet und WiFi IP Settings

An IPv4 address and an IPv6 address with a corresponding subnet mask or prefix can be assigned to the module here. A standard gateway can also be configured. As an alternative to the IPv4 addresses, DHCP can be activated for both the Ethernet and the WLAN interface. If DHCP is active, the interface is configured by the DHCP server used. The configuration option for IPv4 addresses is deactivated when DHCP is active. These settings can be configured for both the network connection and the WLAN connection (under Network →WLAN in the IP settings tab).



The IP addresses of the Ethernet interface and the WiFI interface must not be in the same subnet! This also applies to WiFi mode: Deactivated

WiFi

All the necessary WiFi settings are made in this area. By selecting the desired functionality (Disabled, Connect to existing network, Set up access point), the page adapts to the respective function.

Connect to an existing network

In this area, all the necessary settings can be made to connect the Space module to an existing network. The WPA2 PSK (pre-shared key) and Enterprise protocols are currently supported. Under Enterprise, it is possible to activate or deactivate the validation of the certificate when establishing a connection with the access point.

Set up access point

In this area, all the necessary configurations can be made to use the SPACE module as an access point. The Open, WPA, WPA2 and WPA +WPA2 standards are currently supported.

DNS

Three different DNS servers can be entered here. If the DNS server is to be obtained via DHCP, DHCP must first be set for the corresponding interface (Ethernet or WLAN). If DHCP is activated, no DNS server can be entered manually.

Port forwarding

Both available WAN interfaces can be set under the Port forwarding setting. If the Ethernet interface is to be used as a WAN interface, the SPACE module must be connected to a network with Internet access using an Ethernet cable. Additional devices (e.g. an access point) can now be connected to the SPACE

module via the WiFi interface. These now also have Internet access via the SPACE module. If the WLAN interface is to be used as a WAN interface, the SPACE module must be connected as a WiFi client to a network with Internet access. Additional devices can now be connected via the Ethernet interface. These now also have Internet access via the SPACE module. The 1:1 Nat or port forwarding can only be used if a WAN interface has been selected. Using port forwarding rules, network traffic to a port of the primary IPv4 address of the set WAN interface can be forwarded to a defined port of an internal IPv4 address (target port). This makes it possible to make network resources from the internal network (defined port and defined transport protocol) publicly available. With port forwarding, both source and destination addresses as well as source and destination ports are translated.

1:1 Nat (Beta)

With the help of 1:1 NAT, it is possible to uniquely assign an external IPv4 address of the module (public source address on the WAN interface) to a private IPv4 destination address. This allows network resources of a host in the internal network to be made available under a public IPv4 address. When a new 1:1 NAT rule is added, the respective source address set on the WAN interface is reconfigured with a /32 subnet. However, the source address must not match an IPv4 address previously configured on the module. With 1:1 NAT, only the source and destination addresses of a network packet are translated; the source and destination ports of the transport layer are not changed.

HTTP-Reverse-Proxy (Beta)

HTTP reverse proxy rules allow incoming HTTP(S) requests on a specific network port to be forwarded to a target URL first and the responses from there to be sent back to the original client accordingly. A common use case for this is SSL termination. It is important to note that when defining HTTP reverse proxy rules on ports 22, 53, 80, 123, 443, 8080 or 20220, the module may no longer be fully functional! In addition, rules may have no effect if port forwarding has already been set up for the specified port.

7.2.3 SSL certificate

SSL certificates can be uploaded under this setting. If no SSL certificate is provided, the web server can only be accessed via HTTP (port 80). If an SSL certificate is provided, the web server can also be accessed via HTTPS (port 443). Requests to HTTP are redirected to HTTPS.

7.2.4 Sensors

Settings for the sensors can be made in the I/Os area. Settings for the sensors can be made here. Each pin of sensors S1 and S3 can be inverted and a debounce time can be specified in milliseconds.

Debouncing

The Debouncing Time is specified by the developer. This time describes the minimum time for which a signal must be present at an input for it to be detected and processed by the system. This is not the technically shortest possible time - a time suitable for the intended use should be selected.

7.2.5 Temperature

The excess temperature limit of the internal CPU can be set in the Temperature section. If this is reached or exceeded, the module enters an error state.
A possible configuration value is between 0°C and 100°C; however, a maximum operating temperature of 85°C is recommended.

7.3 Lua scripts

In the Lua scripts tab, the user can activate pre-written scripts. The module currently includes the script for a complete accumulating conveyor by default. The desired scripts can be activated using the slider *Activate*. Scripts can be disabled in the same way.

7.3.1 System scripts

The module currently contains four system scripts as standard.

7.3.1.1 Accumulation conveyor logic

This script enables interface communication via REST API with the autonomous accumulation conveyor of a Flexbox. To be able to use the script, the interface flexbox must be in the same network as the Space module.

Space configuration

The IPv4 address of the interface flexbox must be entered in the configuration.

Flexbox configuration

On the flexbox side, the IPv4 address of the space module must be entered as the predecessor/successor according to the direction of transport. Select flexnet as the protocol used.

REST endpoints provided

The following REST endpoints are provided for communication. As with the endpoints described under Protocols, the JSON format is also used for these endpoints:

General endpoints

Query the status of the autonomous system

GET

`/v1/Autonomous/Active`

Response Schema: application/json
Example (Response-Body):

GET/v1/Autonomous/Active

{
 "state": true
 }

Activation/deactivation of the autonomous system

Activating/deactivating the accumulation conveyor on one module is automatically passed on to all other modules of the autonomous accumulation conveyor system. After the accumulation conveyors have been activated, the accumulation sections are first given a free run. This means that if the stop sensor of a conveyor section is not triggered, the motor rotates in the conveying direction for a configurable time in order to localise any packages on the accumulation conveyor.

PUT

`/v1/Autonomous/Activate`

Request Body Schema: application/json
Parameter:

- state (Boolean)
  Activates/deactivates the autonomous system

Example (Request-Body):

PUT/v1/Autonomous/Activate

{
 "state": true
 }

Requirement:
The activator on the interface flexbox must be configured as *Auto*. No other activator may be used within the autonomous system.

Endpoints handover point

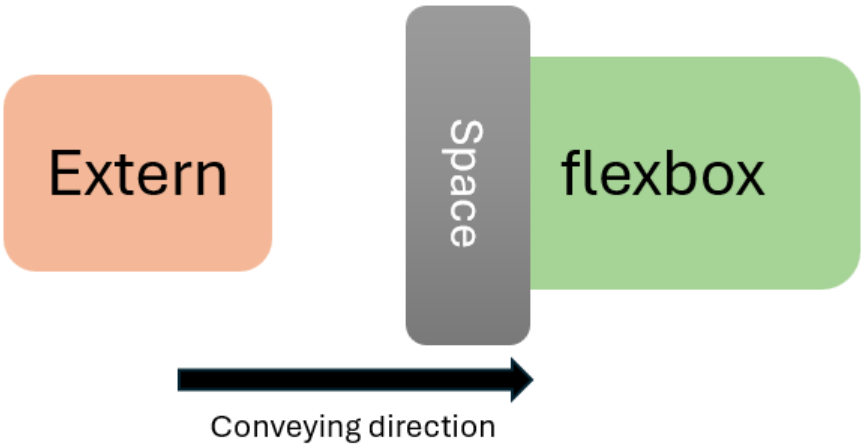


Figure 7.1: System script - Endpoints handover point

Request for authorisation to convey to autonomous system

GET

`/v1/Autonomous/ClearanceToPredecessor`

Response Schema: application/json
Example (Response-Body):

GET/v1/Autonomous/ClearanceToPredecessor
{ "state": true }

(true: *The following may be conveyed*, false: *The following must not be fed in*)

Setting the conveying authorisation to autonomous system

PUT

`/v1/Autonomous/RequestFromPredecessor`

Request Body Schema: application/json *Parameter:*

- state (Boolean)
  (true: *External would like to promote*, false: *External would not like to promote*)

Example (Request-Body):

GET/v1/Autonomous/Activate
{ "state": true }

→If the status of both RequestToSuccessor and CleranceToPredecessor is*true*, the transfer process is started automatically.

Endpoints take-over point

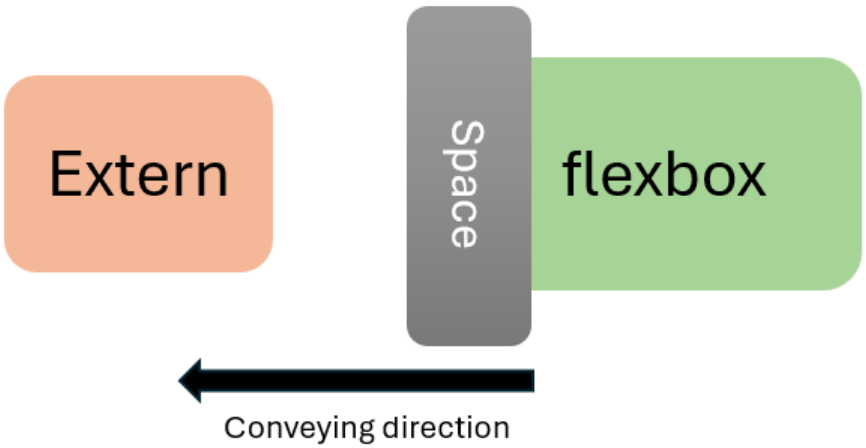


Figure 7.2: System script - Endpoints take-over point

Funding enquiry to external system

GET

`/v1/Autonomous/RequestToPredecessor`

Response Schema: application/json
Example (Response-Body):

GET/v1/Autonomous/ClearanceToPredecessor

```
{
"state": true
}
```

(true: *Autonomous promoter would like to promote*, false: *Autonomous promoter would not like to promote*)

Conveyance authorisation from external system

PUT

`/v1/Autonomous/ClearanceFromSuccessor`

Request Body Schema: application/json
Parameter:

- state (Boolean)
  (true: *The following may be conveyed*, false: *It must not be conveyed*)

Example (Response-Body):

GET/v1/Autonomous/Activate

```
{
"state": true
}
```

→If the status of both RequestToSuccessor and CleranceToPredecessor is*true*, the transfer process is started automatically.

Bidirectional handover/take-over point

The bidirectional handover/transfer point is currently not offered as a standard script. If you are interested, please contact info@flexlog.de. When using the bidirectional accumulation conveyor, the conveying direction can be reversed via the REST API. The setting of the conveying direction on one module is automatically passed on to all other modules of the autonomous accumulation conveyor system. Once the conveying direction has been reversed, the accumulation sections are first given a free run. This means that if the stop sensor of a conveyor section is not triggered, the motor turns in the conveying direction for a configurable time in order to localise any packages on the accumulation conveyor. To utilise this feature, each accumulation conveyor requires both a start and a stop photoelectric sensor. The following REST endpoints are provided for controlling the bidirectional accumulation

conveyor:

Query the conveying direction of the autonomous system

GET

`/v1/Autonomous/ReverseMode`

Request Body Schema: application/json *Example (Response-Body):*

GET/v1/Autonomous/ReverseMode


```
{
"direction": "input",
"id": "Autonomous/ReverseMode",
"result": "ok",
"state": true
}
```


(true: *Autonomous accumulation conveyor is the handover point*, false: *Autonomous accumulation conveyor is the take-over point*)

Setting the conveying direction of the autonomous system

PUT

`/v1/Autonomous/SetReverseMode`

Request Body Schema: application/json *Parameter:*

- state (Boolean)
  (true: *Autonomous accumulation conveyor becomes handover point*, false: *Autonomous accumulation conveyor becomes take-over point*)

Example (Response-Body):

GET/v1/Autonomous/SetReverseMode


```
{
"state": true
}
```

7.3.1.2 EmptyConveyorCounter

The EmptyConveyorCounter is currently not offered as a standard script. If you are interested, please contact info@flexlog.de. This script makes it possible to record the total number of free accumulation conveyor spaces within a conveyor line (maximum 10 flexboxes). The number of free accumulation conveyor spaces is provided via the REST API.

Space configuration

The IPv4 addresses of the flexboxes to be monitored must be entered in the configuration.

flexbox configuration

On the Flexbox side, the IPv4 address of the Space module must be entered in the corresponding field of the congestion conveyor script used.

REST endpoints provided

c

Query occupancy of the autonomous accumulation conveyor line

GET

`/v1/Autonomous/ConveyorCounter`

Request Body Schema: application/json *Example (Response-Body):*

GET/v1/Autonomous/ConveyorCounter


```
{
"direction": "input",
"id": "Autonomous/ConveyorCounter",
"result": "ok",
"state": 2
}
```


(state: *Number of free accumulation conveyor segments on the autonomous accumulation conveyor line.*)

7.3.1.3 ConveyorBlocker

The ConveyorBlocker is currently not offered as a standard script. If you are interested, please contact info@flexlog.de. This script makes it possible to manually block the accumulation conveyor positions

of an autonomous accumulation conveyor. This means that transport of the blocked accumulation conveyor location is blocked until it is released again. The accumulation conveyor locations can be blocked separately via two REST-PUT endpoints.

Space configuration

The IPv4 address of the blockable flexbox must be entered in the configuration.

flexbox configuration

On the flexbox side, the IPv4 address of the space module must be entered in the corresponding field (*Blockable by*) of the congestion conveyor script used.

REST endpoints provided

The following REST endpoints are provided for communication. As with the endpoints described under Protocols, the JSON format is also used for these endpoints:
Blocking of an accumulation conveyor space

PUT

`/v1/Autonomous/BlockConveyor_1`

PUT

`/v1/Autonomous/BlockConveyor_2`

Request Body Schema: application/json
Parameter:

- state (Boolean)
  (true: *Block accumulation conveyor location*, false: *enable accumulation conveyor location*)

Example (Response-Body):

GET/v1/Autonomous/BlockConveyor_1
{ "state": true }

7.3.1.4 TrafficLightController

The TrafficLightController is currently not offered as a standard script. If you are interested, please contact info@flexlog.de.

7.3.2 User-defined scripts

In this section, personal Lua scripts can be configured, programmed and activated. This area is currently in a beta version and can be tested. A free script can be created via *New category*. A category can contain several scripts. The category can be named at category level.
Configuration parameters can also be added, deleted or edited. A configuration parameter consists of the following fields:

- **Variable name:** This name is used to use the parameter in the Lua scripts.
- **Displayname:** This name is displayed to the user when the category is locked. It should best describe the variable for a user of the scripts.
- **Type:** Specifies the type of variable. The types *String, Boolean, Integer, Float,IPv4 Address* and *Timer* are currently offered.
- **Value:** The desired value of the parameter can be entered here. Allowed values for the parameter can be specified via *Extended*. The user of the script can choose between these values during configuration.

The entire category is locked by activating the *Lock* slider. The subordinate scripts can no longer be viewed and edited. The configuration parameters are limited to the values display name, type and value. Only the value can be changed. If permitted parameters have been defined in advance for a configuration parameter, only these values are offered as a selection. All parameters marked as configurable can now be configured by the user. Non-configurable parameters are hidden, but can still be used in the scripts. Any number of scripts per category can be created in the script area. Free scripts can be programmed in Lua here.
Pre-defined configuration parameters can be used in the scripts via their *variable name*. A library is provided so that the module's hardware can be used as objects within the code. All the options provided tools of this library are described in Appendix: Flexlog Library.



After a script has been activated, it is no longer possible to control or configure the module via a bus system.

Advanced Mode

This mode is enabled by default. User scripts are automatically executed without the need for a main script.

Expert Mode

It is possible to change the mode in a drop-down menu in the upper right corner. In expert mode, the main script is visible and editable by the user.



After switching to expert mode, it is not possible to change the category back to advanced mode.

7.4 Protocols

7.4.1 REST API

This section describes the use of the REST API provided. All messages use the JSON format. By default, PUT and GET endpoints are provided for each IO port. In addition, the relays can be controlled via PUT requests. The endpoints respond with standard responses. In the following, the IP of the desired interface must be used for ***Modul_IP***.

GET

The states of all digital I/Os can be queried via HTTP GET under the base URL.
Base URL: **http://*Modul_IP*/v1**
In addition, the status of individual DIs can be explicitly queried by appending the corresponding name to the base URL.
Example GET: **http://*Modul_IP*/v1/S1_Pin4**
The corresponding names are: S1_Pin2, S1_Pin4, S3_Pin1, S3_Pin3, S3_Pin4, S3_Pin5, S3_Pin6, S3_Pin8.
The answers always consist of the following keys:

- **direction** specifies the direction of the signal on the hardware side. A distinction is currently made between input_output and output. An output and an input_output can be written via PUT, an input_output can also transmit its current status via a GET request.
- **id** describes the name of the requested component.
- **result** provides feedback on any problems with the I/O port. If the function is correct, the value of this key is *ok*. If this is not the case, the possible source of the error can be analysed via the status page.
- **state** contains the current process value of the pin. Logically *true* or *false* is supplied.

Example response to GET request


```
{
"direction": "input_output",
 "id": "S1_Pin4",
 "result": "ok",
 "state": true
}
```

PUT

All digital I/O and the relays can be controlled via PUT request. The request must contain the key *state* with a logical value [true/false]. An example of this would be **http://*Modul_IP*/v1/S3_Pin5** with the body:

Example body for PUT request

```
{
 "state": true
}
```

The answers always consist of the following keys

- **id** describes the name of the requested component.
- **result** provides feedback on any problems with the I/O port. If the function is correct, the value of this key is *ok*. If this is not the case, the possible source of the error can be analysed via the status page.
- **state** contains the current process value of the pin. Logically *true* or *false* is supplied.

Example response to PUT request

```
{
"id": "S3_Pin5",
"result": "ok",
"state": true
}
```

Additional endpoints

In addition to the hardware-related endpoints, Space offers a range of endpoints that are responsible for communication between external areas and the internal Lua scripts (example: congestion conveyor). These endpoints are described separately in Lua scripts and in relation to their application purpose.

7.4.2 NTP

The desired NTP server can be entered and saved. By accepting and synchronising, the entered server is used as the NTP server and synchronisation is carried out. The last synchronisation is displayed in the lower section next to the system time.

7.4.3 MQTT

MQTT (Message Queuing Telemetry Transport) is a lightweight protocol that was specially developed for efficient communication between devices in the Internet of Things (IoT). It enables the real-time exchange of messages via a central broker. With a simple publish-subscribe model and predefined topics, MQTT offers a resource-saving and scalable solution for networking devices in distributed systems. Space currently supports MQTT as a beta version.

- **Broker:** The host address and port of the MQTT broker can be set here.
- **Configuration:** The ClientID of the module can be set here. If 'Use device name as ClientID' is activated and the module does not have a device name, the MAC address is used.
- **Topics:** Existing topics can be activated or edited and new topics can be created here. Information on creating MQTT topics can be found under the MQTT documentation tab. The example topic set by the factory can also be used as a help.

7.4.4 Keep Alive

This setting can be used to enter an IPv4 address to which a ping packet is to be sent at a specific time interval.

7.5 User

Here you will find a page for the logged-in user where you can change the password.

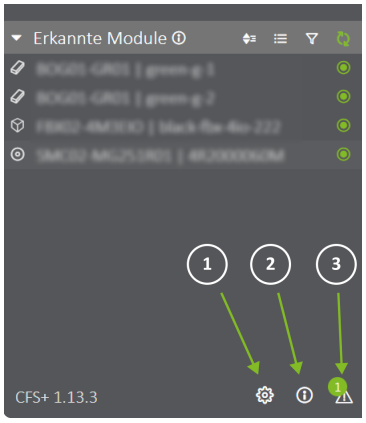
7.6 CFS+

CFS+ is the software developed by flexlog GmbH for module management. It can be used to access the web interfaces of the modules and to configure the modules. It also provides an overview of all modules available on the network and can combine different modules into projects. The CFS+ can be used to update the firmware and monitor the modules.

Overview

After starting CFS+, various programme settings can be made and the modules accessed.

1	**Settings**	Switch tooltips on and off, manage firmware via product number
2	**Info**	Information on the CFS+ version and licences
3	**Status bar**	Status informations about processes in CFS+



The language settings can be found in the top right corner next to the flexlog logo.

Projects

The **Projects** section can be used to combine several moduls into one project. Existing projects can be filtered, sorted and customised in the view. Projects can also be imported.

Further information on the project area can be accessed via the information symbol .

Settings

	Show project configuration/show module configuration	This can be used to switch between project and module configuration.
	Sort by	This is where you can set the sorting of the displayed projects.
	Display	This can be used to select which project data is displayed.
	Filter	This can be used to filter for specific modules.
	Actions	Here you can create new projects or import existing ones.

Discovered modules

All modules in the network can be managed in the **Discovered modules** area.

Further information about the area can be accessed via the information symbol .

Settings

Sort by	This is where you can set the sorting of the displayed modules.
Display	The information displayed can be selected here.
Filter	This can be used to filter for specific modules..
Deactivate module discovery	This is where you can activate and deactivate the automatic module detection.

7.7 Reset to factory settings and restart

The module can be reset to the factory settings and restarted via the CFS+ or the web interface.

Via CFS+

Right click on a module in the **Discovered modules** area to open a menu. You can choose between a complete reset to factory settings and a reset that retains the IP addresses. You can select to restart the module in the same menu.

Via web interface

the function can be accessed via the web interface by clocking on the **Einstellungen, Allgemein** tab. In the Factory settings section, you can choose between a complete reset to factory settings and a partial reset that retains the IP addresses. The module can be restarted using the button.

8 Update firmware version

8.1 Determine current version

The current software version can be found on the module overview page in CFS+. This is described as **SW Version**.

8.2 Updating

The CFS+ module's overview page has a **Verfügbare Firmware** section that lists all currently available firmware versions. The required firmware version can be supplied to the module in three ways:

- The firmware in question is installed directly on the module via **Update** (internet connection required).
- The firmware is made available offline on the device used to access CFS+ via **Download**.
- The firmware can be exported as a .cfp file via **Export**.

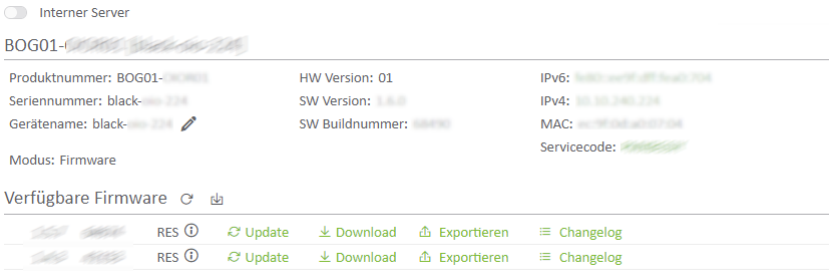


Figure 8.1: firmware update options

9 Repair, maintenance and cleaning

9.1 Replacement

If a Space is defective, it must be replaced. The module must be de-energised before it can be removed. The new module is installed as described under Installation.

9.2 Error handling

Possible faults are described in the chapter Troubleshooting. If a fault cannot be remedied by the solution described there or a fault not described occurs, the module is taken out of service and protected against renewed, inadvertent start-up.
In the event of errors that cannot be rectified or are not described, please contact flexlog GmbH.



When contacting customer support, you should always provide at least the

- serial number,
- SW-version
- and build number

9.3 Cleaning



Damage due to improper cleaning!
The module can be damaged by improper cleaning.

- Do not immerse the module in liquids!
- Do not use any cleaning agents!

If the module becomes dirty, it can be cleaned as follows:

- Vacuuming dust and dirt.
- If a thorough cleaning is required, disconnect the module from the power supply, remove it and clean it with a damp cloth.

10 Troubleshooting

10.1 Error table

Error pattern	Description of the fault	Potential cause	Solving
Red display in the system area at overtemperature	overtemperature	missing ventilation	increase temperature threshold
		too high ambient temperature	switch off system until temperature has dropped
			create ventilation options
Power LED of the Space lights up not	logic supply missing	logic supply defective	check logic power supply unit
		Faulty cabling	Check cabling
		Defective module	replace module if necessary
Collective error in the interface is red	Collective error module	Other error present	Check error displays
Digital I/O or sensor is red in the interface	Error in digital I/O or sensors	Overload of a connected sensor	Remove overload and wait
		Overload digital I/O connection	
		fuse blown	replace module
-	Collective error when temperature thresholds exceeded values	see overtemperature	see overtemperature

10.1.1 Overtemperature

In the event of an overtemperature, this is displayed via the web interface in the System area. The overtemperature status field is red and the current temperature can be seen above it.

List of Tables

4.1 General product properties 12
4.2 LED status 12
4.3 Overview of the connections of the Space modules 13

6.1 Contact assignment Digital I/O (female) 18
6.2 Contact assignment ethernet (female) 18
6.3 Contact assignment sensors (female) 19
6.4 Contact assignment relay sensor plug (female) 19
6.5 Contact assignment power supply (male) 19

