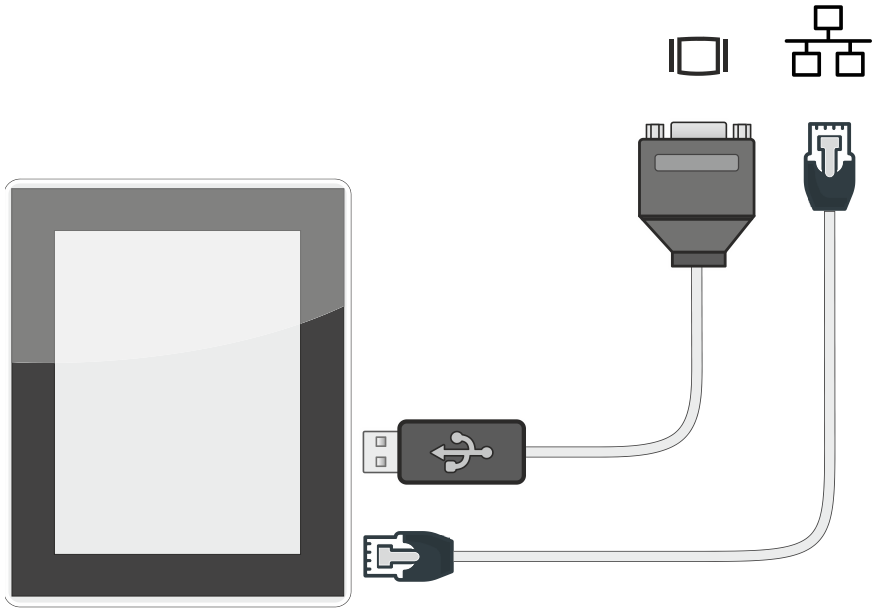
INTERFACES VACUU·SELECT

LADS OPC UA
Modbus TCP
RS-232



Instructions for use

Original instructions
Keep for further use!

This manual is only to be used and distributed in its complete and original form. It is strictly the user's responsibility to carefully check the validity of this manual with respect to the product.

Manufacturer:

VACUUBRAND GMBH + CO KG
Alfred-Zippe-Str. 4
97877 Wertheim
GERMANY

Phone:
Head office +49 9342 808-0
Sales +49 9342 808-5550
Service +49 9342 808-5660

Fax: +49 9342 808-5555
Email: info@vacuubrand.com
Web: www.vacuubrand.com

*Thank you for purchasing this product from **VACUUBRAND GMBH + CO KG**. You have chosen a modern and technically high quality product.*

TABLE OF CONTENTS

1 Introduction 5
1.1 Remote control and interfaces . . . 5
1.2 VACUU·SELECT supported software version . . . 6
1.3 Abbreviations . . . 6
1.4 Target group description . . . 6

2 Applications 9
2.1 Select application . . . 9
2.2 Determining the application ID . . . 9
2.3 List of applications . . . 10
2.4 List of process steps . . . 11

3 Network / IP settings 13
3.1 Connection examples . . . 13
3.2 Settings on the controller . . . 14

4 LADS OPC UA 15
4.1 General information . . . 15
4.2 Settings on the controller . . . 15
4.2.1 LADS OPC UA default settings . . . 15
4.2.2 Remote control settings . . . 16
4.3 Connection and model information . . . 17
4.3.1 Endpoints and sessions . . . 17
4.3.2 Namespaces . . . 17
4.3.3 Stability rules . . . 17
4.3.4 Important BrowseNames . . . 17
4.3.5 Entry points via DeviceSet . . . 18
4.4 Locking and access rules . . . 18
4.4.1 Lock methods . . . 18
4.4.2 Lock status . . . 19
4.5 Executing Templates . . . 19
4.5.1 Available templates (ProgramTemplateSet) . . . 19
4.5.2 Execution methods . . . 19
4.5.3 Vacuum control via FunctionSet . . . 20
4.6 Template management . . . 20
4.6.1 Template portability . . . 20
4.6.2 Template methods . . . 20
4.7 Parametrization . . . 21
4.7.1 Available parameters (SupportedPropertiesSet) . . . 21
4.8 Results . . . 21
4.8.1 CSV file in the ResultFileSet . . . 22
4.9 Peripherals in the FunctionSet . . . 22
4.10 Application example . . . 22

5 Modbus TCP 23
5.1 Settings on the controller . . . 23
5.1.1 Modbus TCP default settings . . . 23
5.1.2 Settings Modbus remote control . . . 23

5.2 Modbus communication 25
5.2.1 Supported function codes 25
5.2.2 Defined data types 25
5.2.3 Representation/setting of pressure values 26
5.2.4 Representation/setting of special values 27
5.2.5 Process parameter deactivation 27
5.2.6 Start/end remote control 28
5.2.7 VACUU·SELECT lock screen 28
5.2.8 Communication examples 29
5.3 Modbus Register Mapping 31
5.3.1 Common Model 31
5.3.2 Control Model 32
5.3.3 Process Control Model 34
5.3.4 Process Step Control Model 35
5.3.5 Service Model 36
5.3.6 SYNCHRO pumping units 36
5.4 Application example 37

6 RS-232 serial port 39

6.1 RS-232 communication 39
6.1.1 Connection example 39
6.1.2 Plug allocation (RS-232) 39
6.1.3 Settings on the controller 40
6.1.4 Representation/setting of special values 42
6.1.5 Process parameter deactivation 42
6.1.6 Start/end remote control 42
6.1.7 VACUU·SELECT lock screen 43
6.2 Interface commands 44
6.3 List of frequently used commands 45
6.3.1 Read commands 45
6.3.2 Write commands 45
6.4 List of all commands 46
6.4.1 Read commands 46
6.4.2 Write commands 49
6.5 Application example 51

7 Index 53

1 Introduction

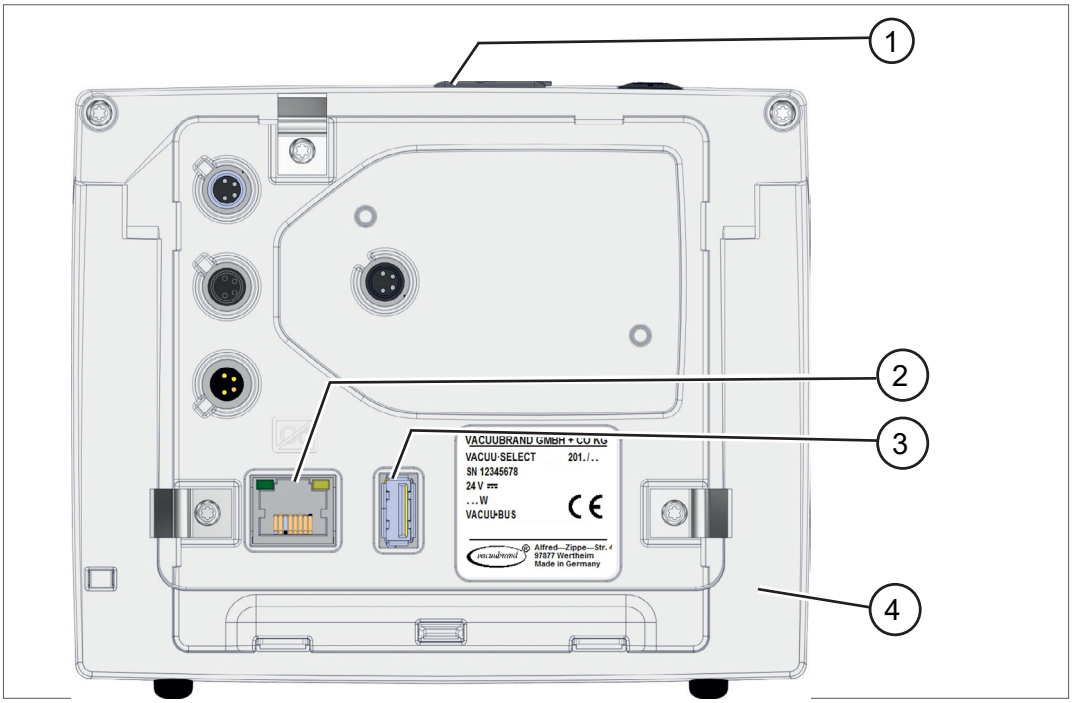
This document describes the general aspects of the communications protocols for LADS OPC UA, Modbus TCP, and the use of the serial port RS-232 with the VACUU·SELECT® vacuum controller.

1.1 Remote control and interfaces

As of software version V1.04 / V1.00 of the VACUU·SELECT®, communication is supported via RS-232 as well as Modbus TCP. As of software version V2.00 / V2.00 of the VACUU·SELECT®, LADS OPC UA is supported. This enables you to remotely monitor and control the controller from a central location, for example directly with a PC or via PLC in a process control system with a host computer.

VACUU·SELECT interfaces

Figure VACUU·SELECT interfaces



1	USB port, type A
2	RJ45 socket – LAN connection (Ethernet)
3	USB port, type A
4	Rear view of ***VACUU·SELECT®*** vacuum controller

IMPORTANT! The data link to the controller is not encrypted. The operator's IT must implement suitable measures to ensure network security.

1.2 VACUU·SELECT supported software version

Supported software version

Version	LADS OPC UA	Modbus TCP	RS-232
V1.00 / V1.00	---	---	---
V1.01 / V1.00	---	---	---
V1.02 / V1.00	---	---	---
V1.03 / V1.00	---	---	---
V1.04 / V1.00 *... or higher*	---	x	x
V2.00 / V2.00 *... or higher*	x	x	x

All information on the following pages relate to the current software version. To make use of remote access you require software version V1.04 / V1.00 or above.

⇨ If required, carry out a software update on the controller.

1.3 Abbreviations

Abbreviations

h	Hours
ID	Identification number for applications or process steps
LADS	Laboratory and Analytical Device Standard
min	Minutes
NA	No access
NaN	Not a number
OPC UA	Open Platform Communications Unified Architecture
RO	Read only
RW	Read / write
s	Seconds
TCP	Transmission Control Protocol, Ethernet
VS-C	Vacuum sensor for rough vacuum range = atmospheric pressure –1 mbar (atmospheric pressure –0.75 Torr); e.g., VACUU·SELECT sensor, VACUU·VIEW, VSK 3000
VS-P	Vacuum sensor for fine vacuum range = 1 mbar–0.001 mbar (0.75 Torr–0.00075 Torr); e.g., VACUU·VIEW extended, VSP 3000

1.4 Target group description

IMPORTANT!

It is the operator's responsibility to apportion tasks according to the areas of competence and relevant qualifications listed in the ***Responsibility matrix***.

Personnel qualification

Meaning Personnel qualification

Laboratory technician	Laboratory staff with professional qualification in the use of laboratory equipment and knowledge of applications and processes, as well as possible effects of remote access.
Electrician	Person with professional qualification in electrics.
IT expert	System technician in charge of network engineering, network administration and measures for network security and maintenance on the operator's side.

Responsibility matrix

Responsibility matrix and areas of competence

Activity	Laboratory technician	Electrician	IT expert
Interface connection at controller	x	x	–
Connection to PLC	–	x	–
Connection to PC	x	x	x
Network data synchronization	x	–	x
Network configuration (systems integration of controller)	–	–	x
Implementation of IT specific measures for network security*	–	–	x
Remedy of network problems	–	–	x
Controller software update	x	–	–
Reconfiguration** following software update or loading factory settings	x	x	x
Data import/export. e.g., established applications	x	–	–
Data logger download	x	–	–
Troubleshooting	x	–	x
Operation	x	–	–
Advanced operation	x	–	–
Error report	x	x	x
Remedy	x	–	(x)
Remote control, responsibility for process security during remote access	x	–	–

** Data transfers with Modbus TCP or RS-232 are not encrypted.*

*** Network or interface settings*

2 Applications

2.1 Select application

Select application via ID

Applications in the vacuum controller are labeled with an application identification number (application ID), also referred to as *Process Application ID*.

Using these defined application IDs, the required application can be selected via remote control, LADS OPC UA, Modbus TCP or RS-232.

If the host computer issues a demand for applications that are not technically supported by the device, the transmitter receives a corresponding error message. This message can be processed into an appropriate message in the host computer.

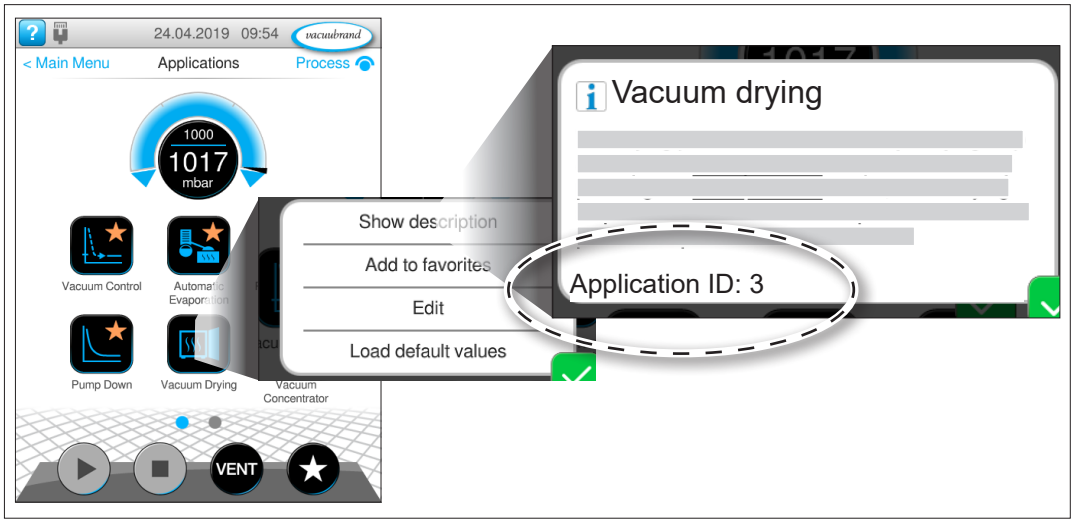
The same functions can be displayed via the interfaces as can locally on the controller.

2.2 Determining the application ID

VACUUBRAND standard applications are in the 0–99 ID range. When you create your own applications – e.g., a new application in the editor – or import an existing application they are automatically assigned a free ID from 100 upwards. Conversely, this means that the ID of an own application must be known for it to be available for selection via remote control.

Main menu / Applications / Context menu: Show description

→ Example
Show application ID on the display



2.3 List of applications

Default process application ID

Application IDs
Standard applications



ID	Standard application	2-point VS-C	2-point VS-P	VARIO VS-C	VARIO VS-P
0	Pump down	x	x	x	x
1	Automatic evaporation	---	---	x	x
2	Application example 1 (automatic evaporation)	---	---	x	x
3	Vacuum drying	x	x	x	x
4	Pump down and hold	x	x	x	x
5	Filtration	x	---	x	---
6	Vacuum control	x	x	x	x
7	Turbo backing pump	---	---	x	x
8	Vacuum concentrator	x	---	x	---
9	Gel drying	x	---	x	---
10	Freeze drying	---	x	---	x
11	Schlenk line	---	x	---	x
12	VACUU·LAN	x*	x*	x	x
13	Boiling point recognition	x	x	---	---
14	Application example 1 (boiling point detection)	x	x	---	---
...	...				

** Only with VMS-B module*

The standard applications in the controller are the same on every controller and are counted from application ID 0. Depending on the component configuration certain applications may not be available (see columns in the table).

My process application ID

→ Example
Application IDs
Own applications



ID	Standard application	2-point VS-C	2-point VS-P	VARIO VS-C	VARIO VS-P
100	Application example: Pump down 60 %	x	x	x	x
101	Application example: Pump down and hold 5 min	x	x	x	x
...	...				

Your own applications created in the controller editor are automatically counted from ID number 100 → see also chapter: *2.2 Determining the application ID*.

2.4 List of process steps

Process steps in the vacuum controller are labeled with a process step identification number, also referred to as the *Process Step ID*.

All standard applications and specifically created applications use process steps from a pre-defined set of steps. Each process step consists of elementary and optional parameters. Elementary parameters must mandatorily be assigned a value, while optional parameters can be deactivated selectively.

Process step ID

Process step IDs

ID	Designation	Pressure setpoint	Set speed / maximum speed	Hysteresis / start pressure	Minimum / maximum	Duration / run-on time
0	Pump down	---	x	---	(x)	(x)
1	Vacuum control	x	x	x	(x)	(x)
2	Ramp	x	---	x	---	x
3	Vent	x	---	---	---	(x)
4	Hold vacuum	x	x	x	(x)	(x)
5	Automatic boiling point function	---	---	---	(x)	(x)
6	Boiling point recognition	---	---	---	---	(x)
7	Loop	---	---	---	(x)	(x)
8	Turbo backing pump	---	---	---	---	(x)
9	VACUU·LAN	x	---	x	---	x

x = Elementary parameter

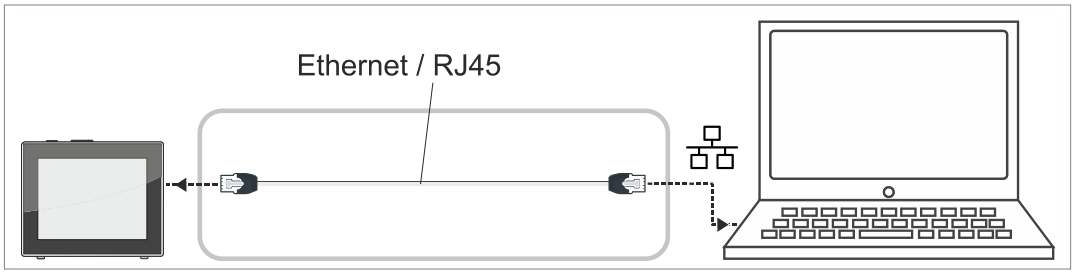
(x) = Optional parameter

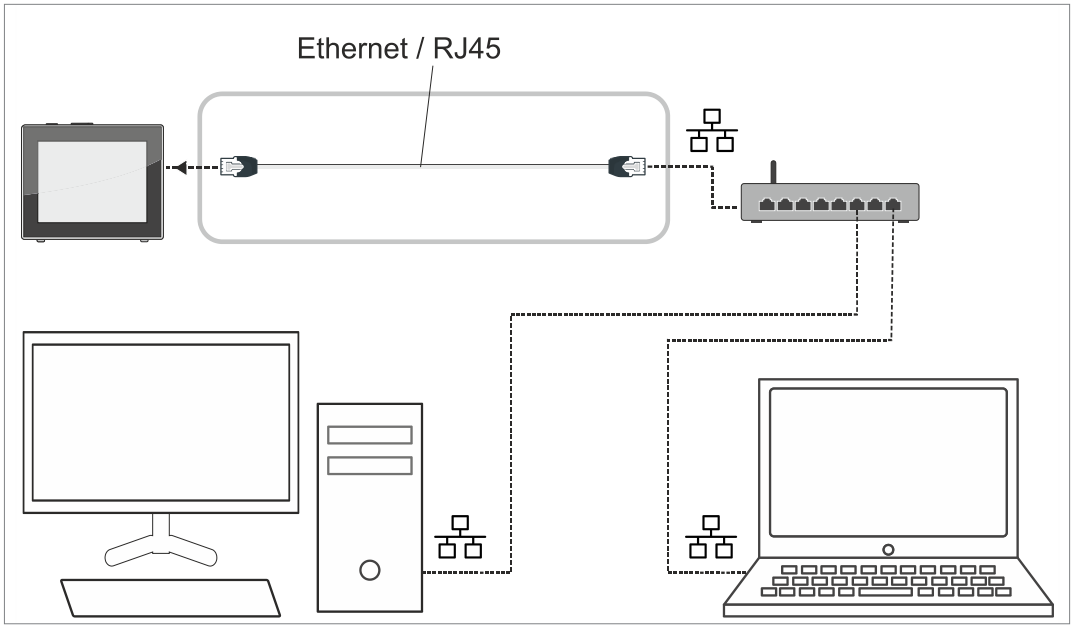
3 Network / IP settings

For remote control via Modbus TCP or LADS OPC UA, use the Ethernet connection RJ45 on the back of the controller. Connection examples for RS-232: → *see chapter 6.1.1 Connection example*

3.1 Connection examples

→ Examples
Ethernet connection





3.2 Settings on the controller

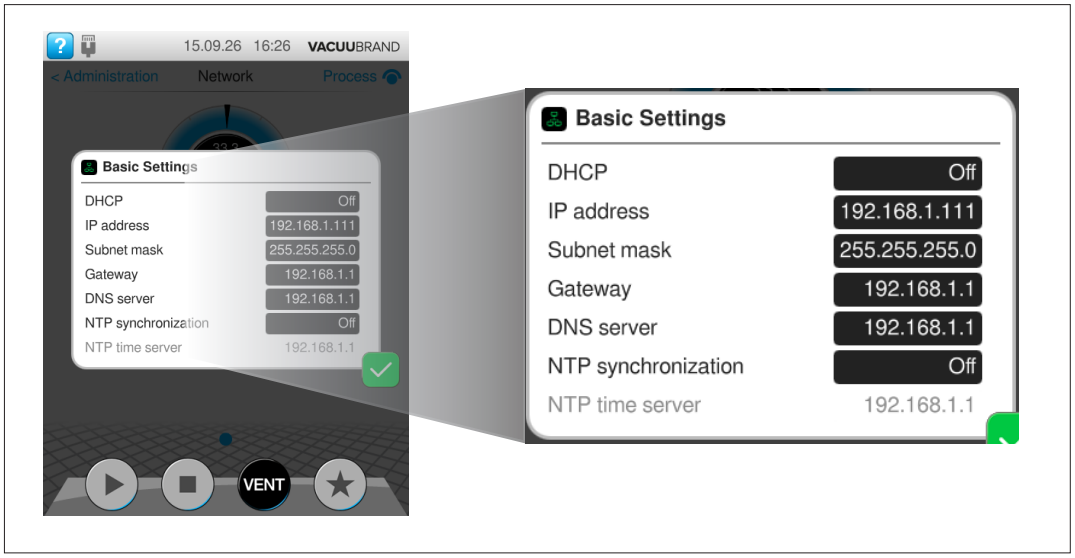
General network/IP settings

General network/IP settings

In order to use Modbus TCP or LADS OPC UA on the VACUU·SELECT, various basic settings must be carried out first.

All relevant settings are summarized in the ***Network*** menu and can be found in the controller as follows: *Process screen / Main menu / Settings / Administration / Network*

Basic Settings menu screen



⇨ Synchronize the data with your required network settings.

⇨ Tap on one of the black input fields to adjust the data; consult your IT department if required.

The VACUU·SELECT can subsequently be reached via the set IP address. If an automatic configuration of the parameters via DHCP is selected, the assigned IP data can be viewed here. In addition, a cable connection to a network is shown by means of a symbol in the status bar (see illustration).

IMPORTANT! If the controller is reset to the factory settings, the data in this menu must be synchronized again.

4 LADS OPC UA

4.1 General information

The following sections describe how users can operate the VACUU·SELECT via the integrated OPC UA server and the underlying OPC UA Companion Specification LADS (OPC 30500-1, “Basics”, Release 1.0.1) “Laboratory & Analytical Device Standard”. Basic knowledge of OPC UA and the corresponding specifications is required.

The focus is on practical workflows (monitoring, program runs, result retrieval) and on integration rules that make client implementations robust. The target group of this documentation are system integrators (SCADA, MES, Edge, IIoT) and providers of laboratory IT (LIMS, ELN, workflow).



Behavior and available nodes/methods may vary with software updates. Users should treat the method signatures and node structures (browsing) provided by the server as the “single source of truth”.

Terminology

LADS terminology

The following terms are frequently used in the subsequent chapters:

Term	Description
Device	VACUU·SELECT vacuum controller (one device per server instance)
FunctionalUnit	Functional unit in the LADS model. Exactly one FunctionalUnit is available per VACUU·SELECT
FunctionSet	Individual functions of the functional unit, e.g., connected sensors and actuators
Lock	Exclusive write/control access via the locking functionality of OPC UA Device Integration (DI)
ProgramManager	LADS object for managing templates, the running program, and results
ResultSet	Collection of results created by program runs

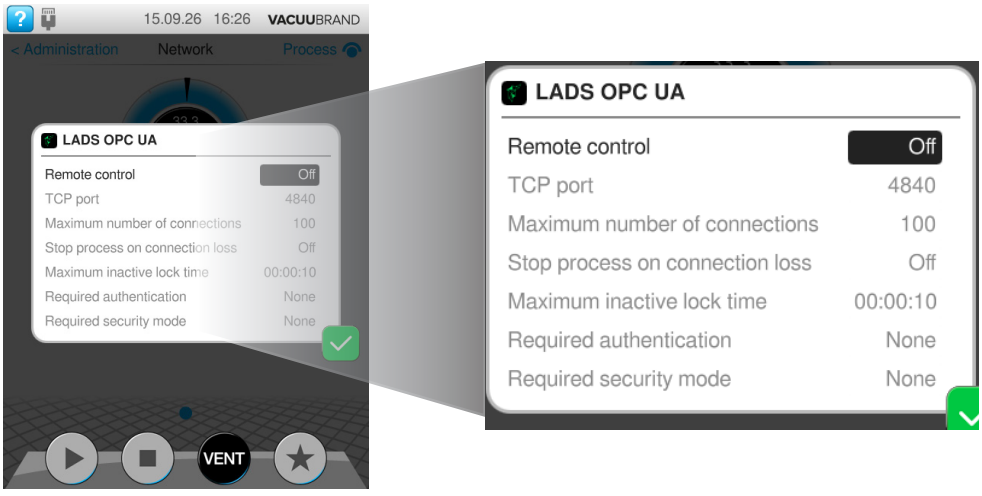
4.2 Settings on the controller

4.2.1 LADS OPC UA default settings

LADS OPC UA default settings

- Remote control (default: Off)
- TCP-Port (default: 4840)
- Maximum number of connections (default: 100)
- Process stop on connection loss (default: Off)
- Maximum inactive lock time (default: 30 s)
- Required authentication (default: None)
- Required security mode (default: None)

LADS OPC UA default settings



LADS OPC UA default settings

4.2.2 Remote control settings

For using LADS OPC UA, two basic modes are available: *Read-only access* or *Full access*. If, for example, only measured values and parameters are to be read from the controller for documentation purposes, but active remote control is not to be permitted, selecting the *Read-only access* is recommended.

By default, up to 100 parallel connections to the VACUU·SELECT are supported. This number can be restricted via the settings. If the configured *Maximum number of connections* has already been reached, any additional connection attempt will be rejected by the controller. A new user can connect to the VACUU·SELECT only after one of the existing sessions has been closed.

Using the option *Process stop on connection loss*, the behavior of the VACUU·SELECT in the event of an uncontrolled interruption (connection closed or timeout) of the controlling connection can be defined. If this option is enabled, the controller automatically stops a running process if the remote control is terminated unexpectedly.

Further information on remote control via LADS OPC UA,
→ *see chapter **4.3 Connection and model information***.

The server currently provides the interface without security modes and without authentication (Security Policy/Mode: None). For productive operation, it is strongly recommended to protect access by network segmentation (e.g., VLAN/firewall, only defined clients).

4.3 Connection and model information

4.3.1 Endpoints and sessions

The following information describes how to connect to the server instance:

Endpoints and sessions

- **Host**: either the configured IP address or the host name according to the following scheme: *vacuuselect-<serial number>*
- **Endpoint scheme**: *opc.tcp://<host>:<port>*
- **Port**: 4080, configurable in the menu,
  → *see chapter 4.2.1 LADS OPC UA default settings*
- **Max. sessions**: 100, configurable in the menu,
  → *see chapter 4.2.1 LADS OPC UA default settings*

4.3.2 Namespaces

The server uses the following namespaces:

Namespaces

- **UA**: OPC UA Core („NodeSet Zero“)
- **DI**: OPC UA Devices (OPC 10000-100)
- **AMB**: OPC UA Asset Management Basics (OPC 10000-110)
- **LADS**: OPC UA for Laboratory & Analytical Device Standard (OPC 30500-1)
- **Machinery**: OPC UA for Machinery (OPC 40001-1)
- **VacuuSelect**: VACUU·SELECT manufacturer-specific

4.3.3 Stability rules

The following basic behavior can be assumed for the VacuuSelect namespace:

Stability rules

- **BrowseNames**: stable (recommended for browse paths and filters)
- **DisplayNames**: may vary depending on the language set in the VACUU·SELECT (for display purposes only)
- **NodeIds**: stable, *except* for dynamically generated nodes (e.g., new *Result* objects in the *ResultSet)*

4.3.4 Important BrowseNames

BrowseNames

Object	BrowseName	Description
Device	VacuuSelectDevice	Central device instance
FunctionalUnit	VacuuSelectUnit	Unique FunctionalUnit
FunctionSet	FunctionSet	Contains all functions of the FunctionalUnit
Lock	Lock	Locking and remote control
ProgramManager	ProgramManager	Template management
ProgramTemplateSet	ProgramTemplateSet	Collection of all available templates
ActiveProgram	ActiveProgram	Progress/status
ResultSet	ResultSet	Collection of results (dynamic)

4.3.5 Entry points via DeviceSet

The LADS Companion Specification provides an entry point for devices via **DI DeviceSet**.

The recommended browse path is: *Objects.DeviceSet.VacuuSelectDevice*.

On the VACUU·SELECT, the **device instance** always uses the BrowseName *VacuuSelectDevice* and the corresponding DisplayName „VACUU·SELECT". Further meta information (e.g., serial number, hardware and software version, …) is available within the device instance.

4.4 Locking and access rules

Remote Control and write access

The VACUU·SELECT uses the **DI Locking** functionality to coordinate exclusive remote control and write access. **Reading is always allowed**, regardless of the current lock state.

If no other connection currently holds an active lock, method calls and write operations to variables (if otherwise permitted) are available. With **InitLock** the lock can be actively requested. Alternatively, the lock is assigned on the first method call or write operation to a variable. If another client already holds an active lock, **no method calls** and **no write operations** are possible. However, read access to the entire address space remains available.

On **inactivity**, the lock expires and is automatically released for other connections. A **Maximum Locking Interval** (Maximum inactive lock time) can be configured in the device menu. If the controlling client disconnects, the lock is released automatically.

If methods are called cyclically or variables are written, the lock is renewed automatically. For longer sequences, **RenewLock** can be called periodically to keep the lock active. It is recommended to hold a lock only as long as necessary and to call **ExitLock** upon completion.

4.4.1 Lock methods

Locking is part of the FunctionalUnit and is controlled via the following methods:

Lock methods

- **InitLock**: initialize lock (request exclusive access)
- **RenewLock**: renew lock (to prevent expiration)
- **ExitLock**: release lock early



The specific method signatures (InputArguments/OutputArguments) can be read by clients directly from the server at runtime (OPC UA Method Node → Property *InputArguments/OutputArguments*).

4.4.2 Lock status

The following read-only nodes are available for status queries:

Lock status

- **Locked**: boolean status indicating whether the unit is locked
- **LockingUser**: user identifier (if known to the controlling client)
- **LockingClient**: client identification
- **RemainingLockTime**: remaining time until automatic release

4.5 Executing Templates

Templates

This chapter describes executing an application via LADS. In the LADS context, applications as used in the VACUU·SELECT are referred to as *Templates* or *ProgramTemplates*. If the device/the FunctionalUnit is locked (→ *see chapter **4.4 Locking and access rules***), method calls and write operations are blocked, i.e., executing templates is not possible.

The templates available on the device can be accessed via the *ProgramTemplateSet*. This list may vary depending on the device configuration (e.g., depending on connected peripherals). To select a template, the unique *TemplateId* is always required.

4.5.1 Available templates (ProgramTemplateSet)

The available templates correspond to the applications available locally on the device. Their unique identification is performed via *TemplateIdX* (X is the sequential application ID assigned by VACUU·SELECT; → *see chapter **2.2 Determining the application ID***).

Templates can be transferred between devices (→ *see chapter **4.6 Template management***).

4.5.2 Execution methods

The following methods are available for executing templates:

Methods to execute templates

- **Start**: starts the FunctionalUnit with the currently selected template and optional parameters. On the VACUU·SELECT, properties at start are optional. Keys (QualifiedName) must match entries in the *SupportedPropertiesSet*.
- **StartProgram**: starts a program run based on a template. *ProgramTemplateId* is required (use *TemplateId17* for example). All other inputs are optional (properties, job/task IDs, samples). *DeviceProgramRunId* is returned by the device and is used to uniquely identify the later *Result*.
- **Stop**: ends the running process/program run. On the VACUU·SELECT, the *Result* is created at process end (Stop) (→ *see chapter **4.7 Parametrization***).

4.5.3 Vacuum control via FunctionSet

In addition, the FunctionalUnit of the VACUU·SELECT can also be started via its FunctionSet. To do so, vacuum control is represented as a **Pressure Controller** (LADS type definition AnalogControlFunctionType) and allows parameterization as well as execution of various methods. Status information and the current pressure value are also available.

Whenever one of the following methods is called, the device may first switch to the default application **Vacuum Control** (*Application ID 6* or *TemplateId6*):

- **Start**: starts the FunctionalUnit with the currently set parameters (e.g., *TargetValue*). Parameters can be changed during execution.
- **StartWithTargetValue**: identical to Start, but optionally accepts a new *TargetValue* and applies it immediately.

Stopping execution of the template via the **Stop method** is performed as described in chapter ***4.5.2 Execution methods***.

4.6 Template management

The VACUU·SELECT implements the LADS ExtendedProgramManager functionality. This allows *Program Templates* to be transferred from the client to the device (upload), from the device to the client (download), and to be deleted from the device (remove).

The template data is **proprietary** and cannot be meaningfully edited outside the device. The methods are therefore primarily intended for:

- Backup/restore of templates
- Transfer between devices (migration/fleet operation)
- Documentation on the customer side

4.6.1 Template portability

Template transfer

Templates can be transferred from device A to device B. For maximum compatibility, the **software versions of both controllers should be identical** at the time of transfer. In the event of incompatibilities, the device returns corresponding **StatusCodes**.

4.6.2 Template methods

The following methods are available for template management:

Template management

- **Upload**: the server creates an entry in the *ProgramTemplateSet* for each uploaded template. The *TemplateId* is assigned by the device and, on the VACUU·SELECT, follows the *TemplateIdX* scheme.
- **Download**: download a template by specifying the unique *TemplateId* or *Applikations-ID* (designation locally on the device).
  → *see chapter 2.2 Determining the application ID*
- **Remove**: delete a template from the device.

4.7 Parametrization

Applications/templates can be parameterized either by direct access to variables within the *FunctionSet* or via the collection of all parameters in the *SupportedPropertiesSet*. This list is static. However, depending on the device configuration (e.g., due to connected peripherals) or settings, some parameters may not be available.

The following **StatusCodes** are possible:

Available StatusCodes

- **Good**: valid parameter value that can be processed normally
- **Good_NoData**: parameter not available, default value assigned
- **Uncertain**: indeterminate parameter value, e.g., the pressure setpoint when venting to atmospheric pressure

For example, parameterizing a pump speed is only possible with VARIO pumps and is not available for other product variants.

Whenever parameters (properties) are used in method calls, the *BrowseNames* of the parameters listed below must be used.

4.7.1 Available parameters (SupportedPropertiesSet)

The following parameters are generally available:

Available parameters

- **Duration**: duration or delay time
- **Hysteresis**: hysteresis for two-point control or switch-on pressure
- **Limit**: minimum/maximum
- **Pressure**: pressure setpoint
- **Speed**: maximum speed or set speed for VARIO control

4.8 Results

Results and CSV files at end of process

The VACUU·SELECT dynamically creates a *Result* when a process ends (Stop). The end of the process can be triggered both via the OPC UA connection and locally via operation on the display. *Results* are generally read-only and are only available during the controller runtime, i.e., they are not persistent through a power-cycle of the VACUU·SELECT. Therefore, it is recommended to download *Results* (especially CSV files) automatically as soon as possible after the end of the process and archive them outside the device.

In the *ResultSet*, a *NodeVersion* is maintained and incremented when changes occur. Clients can use this for an efficient polling strategy:

- Read *NodeVersion* periodically
- In case *NodeVersion* increases: browse the *ResultSet* anew and process new entries

4.8.1 CSV file in the ResultFileSet

In the *ResultFileSet*, the associated **CSV file** is available for download (corresponding to the download that is also possible locally via USB flash drive). Access to the data is provided via the OPC UA **FileType**. Larger files are transferred in chunks (the client must support *Chunked Reads*).

4.9 Peripherals in the FunctionSet

Peripherals - Supported components

Within the *FunctionSet* of the *FunctionalUnit*, in addition to the *PressureController* that maps the basic functionality and its parameterization (→ *see chapter **4.5.3 Vacuum control via FunctionSet***), peripherals possibly connected to the VACUU·SELECT are also represented.

For each element of the *FunctionSet*, the *IsEnabled* variable indicates whether the respective component is available in the system. That means it is connected and enabled. The following components are currently supported:

- **Coolant Valve**: state of the coolant valve
- **Inlet Valve**: state of the inlet/suction line valve
- **Level Sensor**: state of the level sensor
- **Pressure Sensor**: state and measured value of the pressure sensor
- **Vent Valve**: state and control of the vent valve

4.10 Application example

The following sequence describes a simple application example for running a process, monitoring the progress, and finally downloading the log data.

1. Check lock status (*Lock.Locked*).
2. Call *Lock.InitLock* (if remote control is planned).
3. Select *ProgramTemplateId* (from *ProgramManager.ProgramTemplateSet*).
4. Call *FunctionalUnitState.StartProgram* and store *DeviceProgramRunId*.
5. Monitor progress via *ProgramManager.ActiveProgram*.
6. Call *FunctionalUnitState.Stop*.
7. Browse *ProgramManager.ResultSet* and load log data as CSV file.
8. *Lock.ExitLock*.

5 Modbus TCP

5.1 Settings on the controller

5.1.1 Modbus TCP default settings

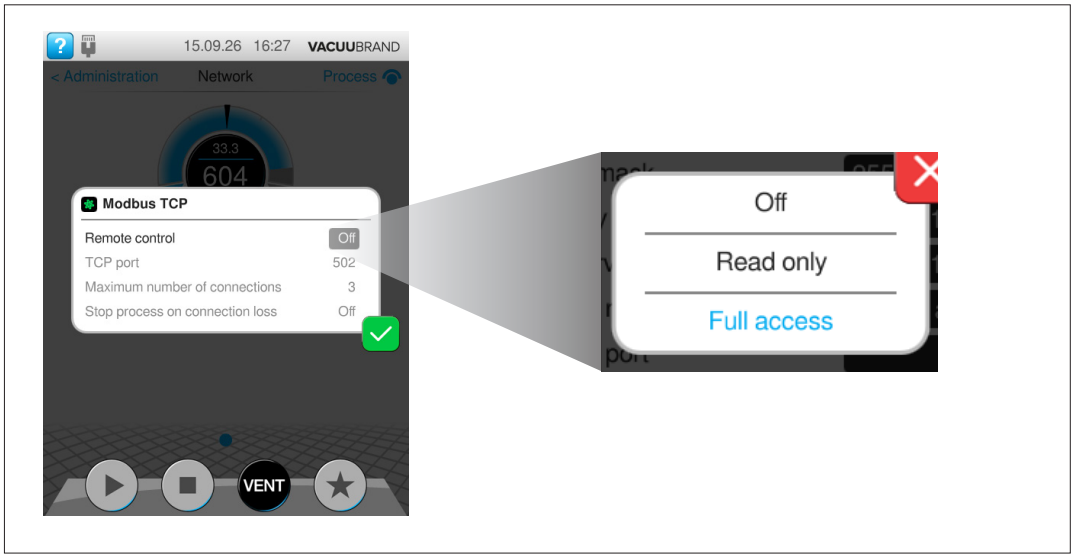
Modbus TCP default settings

The settings relevant for Modbus TCP can be found at the end of the ***Network*** context menu:

- Modbus remote control (default: Off)
- Modbus port (default: 502)
- Maximum number of connections (default: 3)
- Stop process on connection loss (default: Off)

5.1.2 Settings Modbus remote control

Settings Modbus remote control

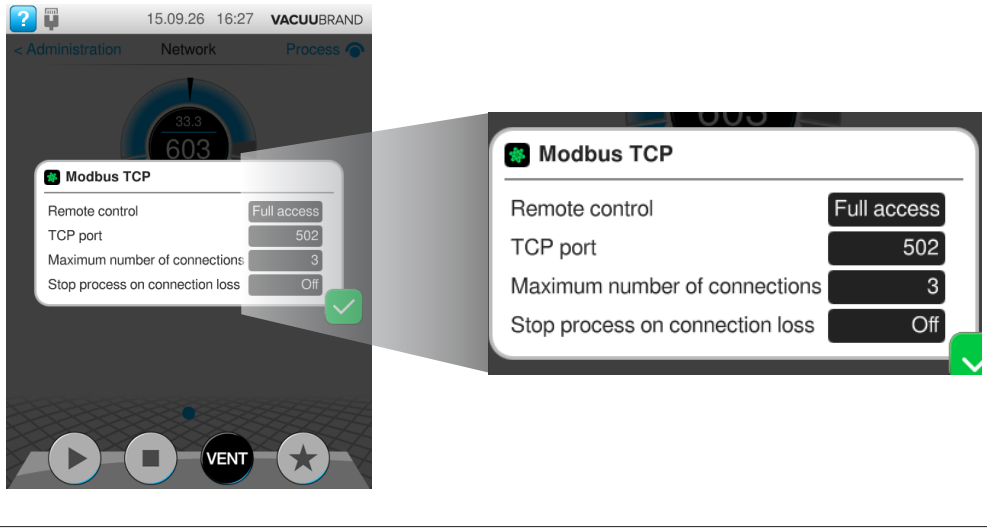


Two fundamental types of access are available when using Modbus TCP: ***Read only access*** or ***Full access***. If the aim for example is simply to read out measured values and parameters from the controller for report purposes, with no active remote control permissible, we recommend selecting the ***Read only*** setting.

Modbus remote control full access/read only

→ Example

Modbus remote control with full access



Up to three parallel TCP connections to the VACUU·SELECT are supported as standard. This number can be limited by means of a setting. If the previously set ***Maximum number of connections*** has been reached, each additional connection attempt is rejected by the controller. A new subscriber can only connect to the VACUU·SELECT when one of the existing connections has been closed.

The option for ***Stop process on connection loss*** allows the behavior of the VACUU·SELECT in the event of an uncontrolled loss of the controlling connection (connection closed or timeout) to be defined. If this option is activated, the controller automatically stops a running process when an uncontrolled interruption to remote control has occurred.

5.2 Modbus communication

5.2.1 Supported function codes

Supported function codes

Code	Description
03 (0x03)	**Read Holding Registers** For reading single or multiple successive register values.
06 (0x06)	**Write Single Holding Register** Exclusively for writing single register values (data types int16, uint16, enum16 or sf). If several registers are merged into a larger data type (e.g., int32, uint32 or string), function code 16 must be used for writing. Overview of supported data types → *see chapter 5.2.2 Defined data types*.
16 (0x10)	**Write Multiple Holding Registers** For writing multiple successive registers that are merged into a larger data type or which require consistent adjustment → *see chapter 5.2.3 Representation/setting of pressure values*.

5.2.2 Defined data types

The available Modbus registers described in chapter *5.3 Modbus Register Mapping* are based on the data types listed below. If certain functions are not supported by the controller, the associated Modbus registers are assigned the relevant NaN values.

Each individual 16-bit register is represented in little endian byte order. For 32-bit data types consisting of two 16-bit registers, the lower register address represents the least significant part of the entire data type.

Data types composed of more than one Modbus register must be written with function code 16 → *see chapter 5.2.1 Supported function codes*.

Modbus data types and NaN values

Type	Description	NaN Value
int16	Signed integer value (16 bit)	0x8000
int32	Signed integer value (32 bit)	0x80000000
uint16	Unsigned integer value (16 bit)	0xFFFF
uint32	Unsigned integer value (32 bit)	0xFFFFFFFF
float32	Floating-point, simple accuracy, IEEE-754 (32 bit)	0xFFFFFFFF
enum16	Number code (16 bit). The breakdown of the possible codes can be found directly under the designation of the according Modbus register.	0xFFFF
string	String (multiple of 2 bytes). A zero-terminating value or a value of fixed length.	0x0000
p	Representation of pressure values as a floating-point number or by means of integers; for more information → *see chapter 5.2.3 Representation/setting of pressure values*.	*see 5.2.3*

5.2.3 Representation/setting of pressure values

Representation/setting of pressure values

Pressure values, such as set and actual pressures, can be represented either with floating point arithmetic or with integer values → *see Modbus register #40812*. In the data models → *see chapter 5.3 Modbus Register Mapping*, a pressure value always occupies three Modbus registers, which are assigned differently depending on the selected representation. It is therefore only possible to change pressure values with function code 16 → *see chapter 5.2.1 Supported function codes*.

Integer representation (factory setting)

Integer representation (factory setting)

The values are represented with a mantissa (uint32) and a signed exponent (int16). A pressure value is obtained with the following calculation formula:

Pressure value = mantissa * $10^{exponent}$

Examples: 123 * 10^{-3} = 0.123 mbar
500 * 10^{0} = 500 mbar

Register	Assignment	Data type
0 **1**	Mantissa	uint32
2	Exponent	int16

Floating-point representation

Floating-point representation

The value is represented as float32. Only two of the three registers are used in this version. Read and write commands must be executed accordingly on the first two registers.

Register	Assignment	Data type
0 **1**	Pressure value	float32
2	---	---

5.2.4 Representation/setting of special values

Representation/setting of special values

As well as the value ranges and NaN values described in chapter ***5.2.2 Defined data types***, the following special values are additionally defined below for parameters of data type p (= pressure values), provided they are supported by the relevant process step → see chapter ***2.4 List of process steps***.

AUTO setting for parameter *Hysteresis*

Register	Integer value	Floating-point value
0 **1**	0xFFFFFFFE	0xC0000000
2	0x0000	0x8000

ATM setting for parameter *Pressure setpoint*

Register	Integer value	Floating-point value
0 **1**	0xFFFFFFFD	0xC0400000
2	0x0000	0x8000

5.2.5 Process parameter deactivation

Process parameter deactivation

Provided the respective parameter of a process step supports deactivation, e.g., the parameter *Duration* in the process step *Pump down*, deactivation can be achieved by writing the register value 0.

If a pressure parameter is to be deactivated, e.g., parameter *Minimum* in the process step *Pump down*, it may be necessary to write several registers → *see chapter **5.2.3 Representation/setting of pressure values***.

A deactivated parameter subsequently has no influence on the controller or the process sequence → *see chapter **2.4 List of process steps***.

5.2.6 Start/end remote control

Start/end remote control

If settings or process parameters are to be changed on the VACUU·SELECT, remote control must be started first. Although several parallel connections to the device are possible in principle, only one connection can exclusively take on remote control of the device and thereby also block local operation.

Provided no remote control is active yet – including via RS-232 – remote control can be established via Modbus register #40802 → *see chapter* ***5.3.2 Control Model***. For this, various modes are available which either completely block local operation, or at the least permit the connection to be terminated by activating the ON/OFF button on site. It is also possible to specify which screen is displayed in the locked state (*Process screen* or *Pressure graph*).

5.2.7 VACUU·SELECT lock screen

During remote operation via Modbus TCP the display of the VACUU·SELECT is shown locked.

Locked screen

→ Example
Locked controller



When remote control is active, the device display always jumps to the *Process screen*. Alternatively, the display can also jump to the *Pressure graph* screen.

The locked screen is shown as long as remote control is active.

5.2.8 Communication examples

The following tables show examples of Modbus TCP read and write sequences on the VACUU·SELECT.

Read

In the *Read* example, three registers are read from base register #40912 using function code 03.

Client Request ADU						
MBAP Header				PDU		
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Data	
					Starting Address	Qty. of Registers
0x 00 00	0x 00 00	0x 00 06	0x 01	0x 03	0x 9F D0	0x 00 03

Server Response ADU						
MBAP Header				PDU		
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Data	
					Byte Count	Register Values
0x 00 00	0x 00 00	0x 00 09	0x 01	0x 03	0x 06	0x 00 00 44 78 80 00

Write (single register)

In the *Write (single register)* example, register #40802 is assigned a new value using function code 06.

Client Request ADU						
MBAP Header				PDU		
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Data	
					Register Address	Register Value
0x 00 00	0x 00 00	0x 00 06	0x 01	0x 06	0x 9F 62	0x 00 01

Server Response ADU						
MBAP Header				PDU		
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Data	
					Register Address	Register Value
0x 00 00	0x 00 00	0x 00 06	0x 01	0x 06	0x 9F 62	0x 00 01

Write (multiple registers)

The *Write (multiple registers)* example shows a pressure setpoint being set using function code 16 (Modbus registers #41104 to #41106) → *see chapter* ***5.2.1 Supported function codes***.

Client Request ADU								
MBAP Header				PDU				
					Data			
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Starting Address	Qty. of Registers	Byte Count	Register Values
0x 00 00	0x 00 00	0x 00 0D	0x 01	0x 10	0x A0 90	0x 00 03	0x 06	0x 01 4D 00 00 FF FF

Server Response ADU						
MBAP Header				PDU		
					Data	
Transaction ID	Protocol ID	Length	Unit ID*	Function Code	Starting Address	Qty. of Registers
0x 00 00	0x 00 00	0x 00 06	0x 01	0x 10	0x A0 90	0x 00 03

) The **Unit ID of the request ADU is reused in the response ADU..*

5.3 Modbus Register Mapping

5.3.1 Common Model

Address[1)]	Size	Name	Data Type	Description	Access
40000	4	VACUUBUS_ID	string(8)	VACUU·BUS Modbus identifier. Uniquely identifies this as a VACUU·BUS Modbus Register Map. Fixed value = “VACUUBUS” (0x5641435555425553)	RO
40004	1	VACUUBUS_MID	uint16	VACUU·BUS Modbus Model Block identifier. Uniquely identifies this as a VACUU·BUS Modbus Common Model Block. Fixed value = 0x0001	RO
40005	1	VACUUBUS_Length	uint16	Length of block in 16 Bit registers. Value = 18	RO
40006	1	Protocol Version	uint16	VACUU·BUS Modbus Protocol Version	RO
40007	1	Device Address	uint16	Unique device address or Modbus Unit ID	RO
40008	1	Manufacturer ID	enum16	Device manufacturer ID. Possible values: 1 = VACUUBRAND GMBH + CO KG	RO
40009	1	Product ID	enum16	Product ID. Possible values: 1 = VACUU·SELECT	RO
40010	10	Serial Number	string(20)	Device serial number	RO
40020	1	Software Version #1	uint16	Device software version The device‘s software represented as an according integer value. Examples: V1.00 = 0x0064 V2.34 = 0x00EA	RO
40021	1	Hardware Version #1	uint16	Device hardware version The MSByte points to a character of the alphabet, representing the board layout revision. The LSByte represents the assembly version. Examples: A.01 = 0x0101 D.12 = 0x040C	RO
40022	1	Software Version #2	uint16	See description of 40020	RO
40023	1	Hardware Version #2	uint16	See description of 40021	RO

1) Register address values with reference to base 0 (protocol address).

5.3.2 Control Model

Address[1)]	Size	Name	Data Type	Description	Access
40800	1	VACUUBUS_MID	uint16	VACUU·BUS Modbus Model Block identifier. Uniquely identifies this as a Control Model Block. Value = 0x0009.	RO
40801	1	VACUUBUS_Length	uint16	Length of block in 16 Bit registers. Value = 11	RO
40802	1	Remote Control Mode	enum16	Remote control mode. Possible values: 0 = Remote off 1 = Remote on, Process screen A[2)] 2 = Remote on, Process screen A[3)] 3 = Remote on, Chart view screen A[2)] 4 = Remote on, Chart view screen A[3)] 5 = Remote on, Process screen B[2) 4)] 6 = Remote on, Process screen B[3) 4)] 7 = Remote on, Chart view screen B[2) 4)] 8 = Remote on, Chart view screen B[3) 4)]	RW
40803	2	Operating Status	uint32	Information on the system‘s current operating status. This bitfield contains the following data: Bit 0 = Sensor overpressure (warning) Bit 1 = Sensor underrange (warning) Bit 2 = Sensor failure Bit 3 = Liquid level sensor triggered Bit 4 = Inlet valve failure Bit 5 = Vent valve failure Bit 6 = Water valve failure Bit 7 = Pump/VMS-B failure Bit 8 = VARIO pump failure Bit 9 = Digital I/O module failure Bit 10 = Analog I/O module failure Bit 11 = EK Peltronic failure Bit 12 = Waiting period liquid level sensor active Bit 13:31 = Reserved for future use By writing zero, all pending failures/warnings will be acknowledged	RW
40805	1	Pressure Unit	enum16	Pressure unit. Possible values: 0 = mbar 1 = Torr 2 = hPa	RW
40806	1	Autostart Mode	enum16	Autostart mode. Possible values: 0 = Autostart disabled 1 = Autostart enabled	RW
40807	1	Vent Valve in Vacuum Control Mode	enum16	Vent valve in vacuum control mode. Possible values: 0 = Vent valve disabled 1 = Vent valve enabled on setpoint change	RW
40808	2	Delay Time of Coolant Valves	uint32	Delay time of coolant valves [s]	RW

40810	2	Delay Time of Liquid Level Sensors	uint32	Delay time of liquid level sensors [s]	RW
40812	1	Data Type of Pressure Values	enum16	Pressure value data type → see chapter *5.2.3 Representation/setting of pressure values* Possible values: 0 = Integer (mantissa + exponent) 1 = Floating point	RW

1) Register address values with reference to base 0 (protocol address).

2) Device locked

3) Unlocking by ON/OFF button possible

4) Only available on SYNCHRO pumping units

5.3.3 Process Control Model

Address[1)]	Size	Name	Data Type	Description	Access
40900	1	VACUUBUS_MID	uint16	VACUU·BUS Modbus Model Block identifier. Uniquely identifies this as a Process Control Model Block. Fixed value = 0x000A (Primary) or 0x000B (Secondary)	RO
40901	1	VACUUBUS_Length	uint16	Length of block in 16 Bit registers. Value = 13	RO
40902	1	Process Application ID	uint16	Selected application‘s ID → *see chapter: **2.2 Determining the application ID***	RW
40903	1	Process Run Mode	enum16	Process run mode. Possible values: 0 = STOP 1 = START	RW
40904	1	Control Vent Valve	enum16	Control the process‘ vent valve(s). Possible values: 0 = Close 1 = Open 2 = Vent to atmospheric pressure and close automatically	RW
40905	1	Temporary Vent Valve in Vacuum Control Mode	enum16	Vent valve in vacuum control mode. This temporarily overwrites the basic settings until process stop (see 40807). Possible values: 0 = Vent valve disabled 1 = Vent valve enabled on setpoint change 2 = Vent valve enabled for vacuum control	RW
40906	1	Current Process Step	uint16	Currently active process step [1..n]. Jumping to next process step by writing the next process step‘s index	RW
40907	1	Number Of Process Steps	uint16	Number of steps in the selected process	RO
40908	1	Process Step Jump Enable	enum16	Possibility to jump to the next process step. Possible values: 0 = Jumping disabled 1 = Jumping enabled	RO
40909	2	Process Time Elapsed	uint32	Process time elapsed [s]	RO
40911	1	Process Vacuum Type	enum16	Process vacuum type. Possible values: 0 = Rough vacuum 1 = Fine vacuum	RO
40912	3	Sensor Value	p	Actual pressure [mbar/Torr/hPa] → *see chapter: **5.2.3 Representation/setting of pressure values***	RO

40915	1	Process State Information	uint16	Information on process status. This bitfield contains the following data: Bit 0 = Pump running (motor speed > 0) Bit 1 = Inlet valve open Bit 2 = Cooling valve open Bit 3 = Vent valve open Bit 4:7 = Reserved for future use Bit 8 = Pump down / actual pressure > set pressure / boiling point detection Bit 9 = actual pressure = set pressure / VACUU·LAN delay / boiling point tracking Bit 10 = actual pressure < set pressure / VACUU·LAN monitoring / auto termination Bit 11:15 = Reserved for future use	RO

1) Register address values with reference to base 0 (protocol address).

5.3.4 Process Step Control Model

Address[1)]	Size	Name	Data Type	Description	Access
41100	1	VACUUBUS_MID	uint16	VACUU·BUS Modbus Model Block identifier. Uniquely identifies this as a Process Step Control Model Block. Fixed value = 0x000C (Primary) or 0x000D (Secondary).	RO
41101	1	VACUUBUS_Length	uint16	Length of block in 16 Bit registers. Value = 14	RO
41102	1	Process Step Selector	uint16	Select a specific process step to set/get parameters. Possible values: 0 = Active process step 1..n = Specific process step	RW
41103	1	Process Step ID	enum16	Unique process step ID. Possible values → *see chapter: **2.4 List of process steps***	RO
41104	3	Set-pressure Value	p	Set-pressure value [mbar/Torr/hPa]	RW
41107	1	Set-speed Value	uint16	Set-speed value or maximum speed VARIO pumps [%]	RW
41108	2	Duration	uint32	Duration of the process step or delay time VACUU·LAN [s]	RW
41110	3	Hysteresis Value	p	Hysteresis value or switch-on pressure [mbar/Torr/hPa]	RW
41113	3	Minimum/Maximum Value	p	Minimum/Maximum value [mbar/Torr/hPa]	RW

1) Register address values with reference to base 0 (protocol address).

5.3.5 Service Model

Address[1)]	Size	Name	Data Type	Description	Access
41300	1	VACUUBUS_MID	uint16	VACUU·BUS Modbus Model Block identifier. Uniquely identifies this as a Service Model Block. Fixed value = 0x000E.	RO
41301	1	VACUUBUS_Length	uint16	Length of block in 16 Bit registers. Value = 11	RO
41302	2	Controller Operating Time	uint32	Controller Operating time [min]	RO
41304	2	VARIO Pump Operating Time	uint32	VARIO pump Operating time [min]	RO
41306	1	VARIO Pump Service Monitoring Enable	enum16	VARIO pump service assistant enable mode. Possible values: 0 = Service assistant disabled 1 = Service assistant enabled	RO
41307	2	VARIO Pump Last Service Time	uint32	VARIO pump last time serviced [min]	RO
41309	1	VARIO Pump Service Interval	uint16	VARIO pump service interval [h]	RO
41310	1	VARIO Pump Service Threshold	uint16	VARIO pump service threshold [%]	RO

1) Register address values with reference to base 0 (protocol address).

5.3.6 SYNCHRO pumping units

For pumping units that can control two processes, the Process Control Model and the Process Step Control Model are available separately for process A and B, respectively. The assignment of the individual registers is described in ***5.3.2 Control Model on page 32*** and ***5.3.3 Process Control Model on page 34***, respectively. The base addresses are defined as follows:

Register area	Process A	Process B
Process Control Model	from 40900	from 41000
Process Step Control Model	from 41100	from 41200

5.4 Application example

Vacuum control

The following sequence shows vacuum control to a pressure setpoint of 12.3 mbar, by way of example.

Activate remote control

Address	Value	Description
40802	1	Remote Control Mode Activate remote control, device locally locked, jump to *Process screen*
	or	
40802	2	Remote Control Mode Activate remote control, device can be locally unlocked, jump to *Process screen*
	or	
40802	3	Remote Control Mode Activate remote control, device locally locked, jump to *Pressure graph*

Select "vacuum control" application (ID=6)

Address	Value	Description
40902	6	Process Application ID

Specify pressure setpoint

Address	Value	Description
41104	123	Set-pressure Value (mantissa)
41106	-1	Set-pressure Value (exponent)

Start process

Address	Value	Description
40903	1	Process Run Mode

Stop process

Address	Value	Description
40903	0	Process Run Mode

Deactivate remote control

Address	Value	Description
40802	0	Remote Control Mode

While the process is running, the following Modbus registers can for example serve to monitor the process flow:

Address		Description
40909		Process Time Elapsed [s]
40912		Sensor Value (mantissa)
40914		Sensor Value (exponent)

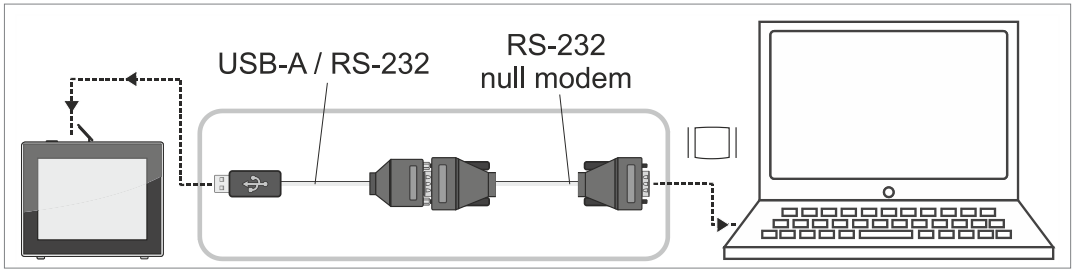
6 RS-232 serial port

Point-to-point communication to a serial device can be established from the controller via a USB adapter. In order to use the functions via RS-232, a USB adapter/RS-232 and null modem cable must be connected to one of the USB ports of the controller.

6.1 RS-232 communication

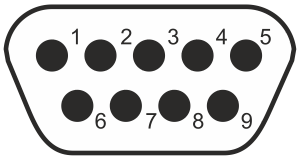
6.1.1 Connection example

→ Example RS-232 connection



Adapter cable, USB to RS-232, 1 m	20637838
RS-232C null modem cable, 2x socket Sub-D 9-pin, 1.5 m	20637837

6.1.2 Plug allocation (RS-232)



Sub-D 9-pin

PIN	Designation	Function
1	DCD (Data Carrier Detect)	Signal received detector
2	RXD (Receive Data)	Received data
3	TXD (Transmit Data)	Transmission data
4	DTR (Data Terminal Ready)	Terminal device ready for operation
5	GND (Ground)	System ground
6	DSR (Data Set Ready)	Standby
7	RTS (Request To Send)	Switch on transmitting part
8	CTS (Clear To Send)	Ready to send
9	RI (Ring Indicator)	Incoming call

→ see also data sheet included in the adapter cable delivery

6.1.3 Settings on the controller

General communication settings

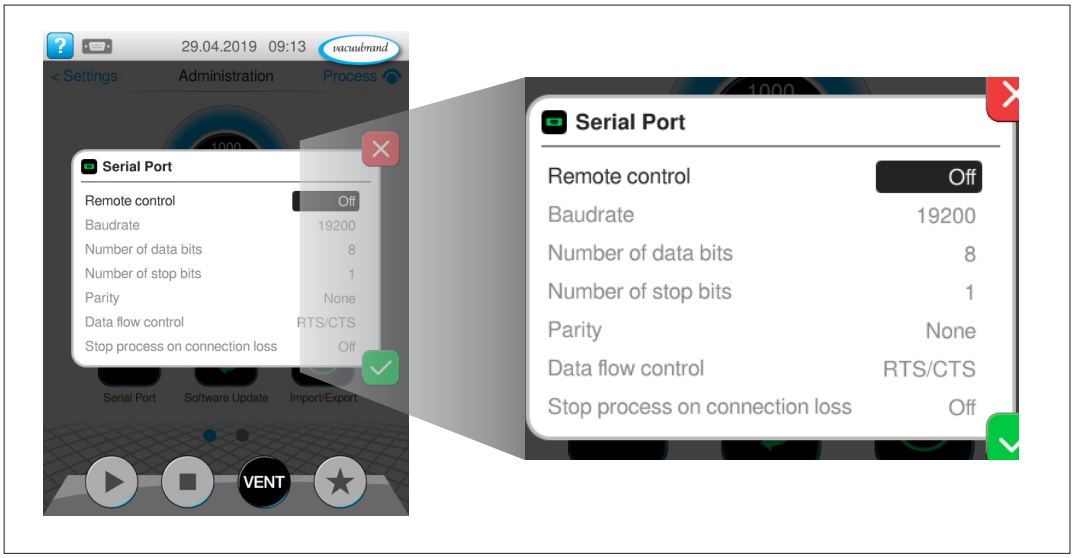
General communication settings

In order to use RS-232 on the VACUU·SELECT, various basic settings must be carried out first.

All relevant settings are summarized in the ***Serial port*** context menu and can be found in the controller as follows: *Process screen / Main menu / Settings / Administration / Serial port*

Serial port context menu screen

→ Example
Serial port context menu

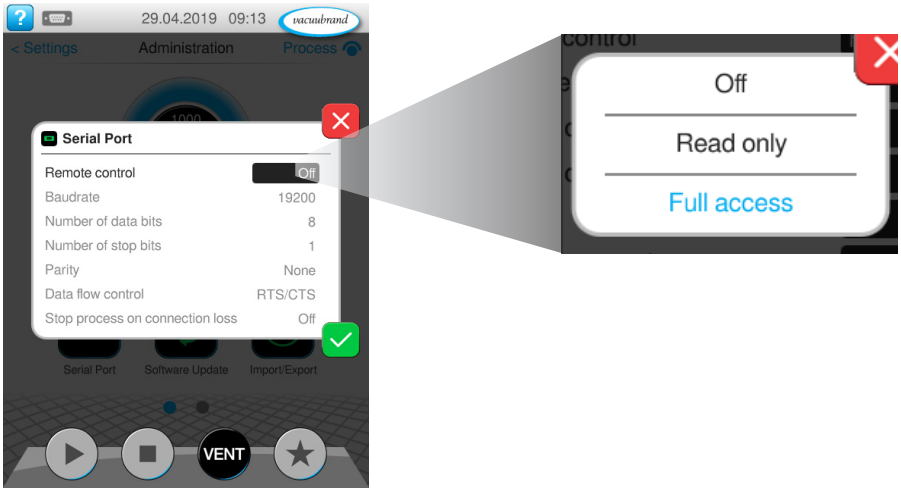


RS-232 default settings

- Remote control (default: Off)
- Baud rate (default: 19200)
- Number of data bits (default: 8)
- Number of stop bits (default: 1)
- Parity (default: None)
- Data flow control (default: RTS/CTS)
- Stop process on connection loss (default: Off)

RS-232 remote control settings

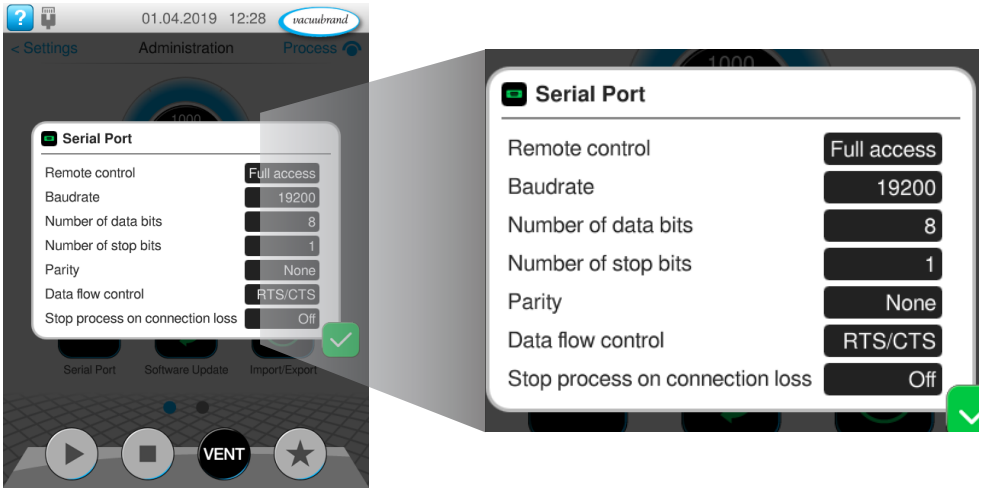
→ Example
Serial port settings



Two fundamental types of access are available when using RS-232: ***Read only*** access or ***Full access***. If the aim for example is simply to read out measured values and parameters from the controller for report purposes, with no active remote control permissible, we recommend selecting the ***Read only*** setting.

Serial port full access/read only

→ Example
Serial port with full access



⇨ Synchronize the data with your required communication settings.
⇨ Tap on one of the black input fields to adjust the data.

IMPORTANT!

If the controller is reset to the factory settings, the data in this menu must be synchronized again.

6.1.4 Representation/setting of special values

Representation/setting of special values

In addition to decimal values, various special values are also defined and can be used with certain interface commands (e.g., **OUT_SP_1**) or process parameters:

- Parameter hysteresis ***AUTO***
- Parameter pressure setpoint ***ATM***

6.1.5 Process parameter deactivation

Process parameter deactivation

Provided the respective parameter of a process step supports deactivation → *see chapter 2.4 List of process steps*, deactivation can be achieved by writing the value 0. A deactivated parameter subsequently has no influence on the controller and/ or the process sequence. (e.g., the parameter *Duration* in the process step *Pump down*) → *see chapter 2.4 List of process steps*.

6.1.6 Start/end remote control

Start/end remote control

If settings or process parameters are to be changed on the VACUU·SELECT, remote control must be started first. Provided no remote control is active yet (not via Modbus TCP either), remote control can be established via the interface command **REMOTE**.

For this, various modes are available which either completely block local operation, or at the least permit the connection to be terminated by activating the ON/OFF button on site. It is also possible to specify which screen is displayed in the locked state (*Process screen* or *Pressure graph*).

6.1.7 VACUU·SELECT lock screen

During remote operation via serial port, the display of the VACUU·SELECT is shown locked.

Locked screen

→ Example
Locked controller



When remote control is active, the device display always jumps to the *Process screen*. Alternatively, the display can also jump to the *Pressure graph* screen.

The locked screen is shown as long as remote control is active.

6.2 Interface commands

Interface commands

All interface commands are written in capitals throughout and end with the respective control characters **<CR>** (Carriage Return), **<LF>** (Line Feed) or both combined **<CR><LF>**. Value entries or parameters are separated from the command by a space character and can flexibly be shortened to the relevant digits (e.g., 5, 05, 005, 0005 are identical for pressure specifications). The response format is dictated by the selected communication mode.

The command set of the VACUU·SELECT is based on NAMUR recommendations and is essentially compatible with the precursor models CVC 2000 and CVC 3000. Some commands of the CVC 2000 and CVC 3000 are no longer supported by the VACUU·SELECT or are no longer required. These are the groups of commands **IN_SP_P** and **OUT_SP_P** (for programs) as well as the command **STORE** for saving settings. In return, the command set of the VACUU·SELECT was extended by some new commands, e.g. to pick up the concept of applications.

Basically, three different communication modes are available (see command **CVC**), which dictate the interpretation of value entries and the response format of the individual commands. In this respect, the baudrate

factory settings match the factory settings of the CVC 3000 and are compatible with the command set of the CVC 2000. Control programs for the CVC 2000 or CVC 3000 can therefore communicate with the VACUU·SELECT without any changes. For full functionality we recommend switching to the VACUU·SELECT communication mode.

Interface commands that don't exist in the CVC 2000 or CVC 3000 command sets (e.g., **IN_APP** or **OUT_APP**) will also work without explicit changeover of the communication mode.

Write commands are only possible when remote control is active → *see chapter 6.1.6 Start/end remote control* and do not provide a response if the factory settings are in place. Whether a command was correctly implemented can be ascertained via a separate read command (see command **IN_ERR**). Use the **ECHO** command to activate an automatic response. If a command was correctly processed, a corresponding reply is delivered in return. The write commands **REMOTE**, **ECHO** and **CVC** are always possible and don't require active remote control.

IMPORTANT!

All settings adjusted by means of write commands are permanently saved by the controller.

To ensure reliable device operation, a pause time of at least 100 milliseconds must be observed between two consecutive commands. Individual write commands, such as selecting an application for example, require a longer pause time for the command to be fully implemented.

6.3 List of frequently used commands

6.3.1 Read commands

Command	Response by selected communication mode			Description
	CVC 2000	CVC 3000	VACUU·SELECT	
IN_PV_1[1)]	XXXX mbar/hPa/Torr	XXXX.X mbar/hPa/Torr		Current sensor pressure value
	X.XXEXX mbar/hPa/Torr (on VS-P sensors)			

6.3.2 Write commands

Command	Parameter	Description
OUT_SP_1y[1),3),4)]	XXXX or XXXX.X X.XXEXX (on VS-P sensors)	Change the set-pressure of process step y without use the vent valve for vacuum control
OUT_SP_Xy[1),3),4)]		Change the set-pressure of process step y with one-time ventilation to the new set pressure, if necessary
OUT_SP_2y[3)]	XX.X XXX	Change maximum pump speed of process step y. Values given in Hz (Communication mode CVC 2000) or % (CVC 3000 or VACUU·SELECT)
REMOTE	0	Exit remote control
	1	Enable remote control. Local operation disabled and process screen active.
	1y	Optional parameter y: 0: Go to process screen 1: Go to chart view screen
	2	Enable remote control. Local operation disabled, but exit is possible by ON/OFF button.
	2y	Optional parameter y: 0: Go to process screen 1: Go to chart view screen
START	*no parameters* or 1	Start process
STOP	*no parameters* or 0	Stop process and acknowledge failures/warnings
	1	Stop process
ECHO	0	Disable Echo
	1	Enable Echo. Write commands will be responded
CVC	2	Communication mode CVC 2000
	3	Communication mode CVC 3000
	4	Communication mode VACUU·SELECT

1) Values according to the device's pressure unit presetting.

2) Parameter y is optional. If y is not specified, the parameter value from the current process step of the selected application is used. If no application was started, the parameter from the first process step is read.

3) Parameter y is optional. If y is not specified, the parameter value is applied to the current process step of the selected application.

4) The basic settings regarding the use of the vent valve become temporary (until the end of the process) overwritten.

5) Any running process is stopped when the command is executed.

6.4 List of all commands

6.4.1 Read commands

Command	Response by selected communication mode			Description
	CVC 2000	CVC 3000	VACUU·SELECT	
IN_PV_1[1)]	XXXX mbar/hPa/Torr	XXXX.X mbar/hPa/Torr		Current sensor pressure value
	X.XXEXX mbar/hPa/Torr (on VS-P sensors)			
IN_PV_Sy[1)]	XXXX.X mbar/hPa/Torr (X.XXEXX on VS-P sensors)			Current pressure value of sensor y Order of numbering as shown in chart view screen
IN_PV_2	XX.X Hz	XXX %		Current pump speed
IN_PV_3	XX:XX h:m		XX:XX:XX h:m:s	Process time elapsed
IN_PV_31	XX:XX:XX			Process time elapsed
	XXX.XX:XX:XX			Process time elapsed including number of days, as soon as runtime exceeds 24 hours
IN_PV_X	XXXX.X XXXX.X … mbar/hPa/Torr (X.XXEXX on VS-P sensors)			Pressure values of all connected sensors
IN_PV_T	XXXXdXXh			VACUU·SELECT operating time in days and hours.
IN_CFG	yXXXX			Selected application 0: VACUU·LAN 1: Pump down 2: Vacuum control 3: Boiling point automatic/detection 4: Custom application or additional standard application
	XyXXX			Coolant valve connected (0/1)
	XXyXX			Vent valve connected (0/1) 0: not connected 1: connected
	XXXyX			Automatic end reached (0/1)
	XXXXy			Remote control active (0/1)
		yXXXXXXXXXXXXXXX		Selected application 0: VACUU·LAN 1: Pump down 2: Vacuum control 3: Boiling point automatic/detection 4: Custom application or additional standard application

Command	Response by selected communication mode			Description
	CVC 2000	CVC 3000	VACUU·SELECT	
IN_CFG		XyXXXXXXXXXXXXX		0..D: Language (hexadecimal) 0: German 1: English 2: French 3: Italian 4: Spanish 5: Turkish 6: Korean 7: Chinese 8: Portuguese 9: Russian A: Polish B: Dutch C: Japanese D: Suomi
		XXyXXXXXXXXXXXX		Pressure unit: 0: mbar 1: Torr 2: hPa
		XXXyXXXXXXXXXXX		Autostart off/on (0/1)
		XXXXyXXXXXXXXXX		Beep on failure/warning (0/1)
		XXXXXyXXXXXXXXX		VARIO pump connected (0/1)
		XXXXXXyXXXXXXXX		VMS-B module connected (0/1)
		XXXXXXXyXXXXXXX		Inlet valve connected (0/1)
		XXXXXXXXyXXXXXX		Coolant valve connected (0/1)
		XXXXXXXXXyXXXXX		Vent valve connected (0/1)
		XXXXXXXXXXyXXXX		Digital I/O module connected as fault indicator interface (0/1)
		XXXXXXXXXXXyXXX		Level sensor connected (0/1)
		XXXXXXXXXXXXyXX		Digital I/O module connected as remote module interface (0/1)
		XXXXXXXXXXXXXyX		Number of the selected sensor Order of numbering as shown on chart view screen
		XXXXXXXXXXXXXXy		Number of connected sensors
		XXXXXXXXXXXXXXX		Remote control active (0/1)
IN_ERR	yXXX			Failure on VARIO pump, inlet valve or VMS-B module (0/1)
	XyXX			Sensor overpressure or underrange warning (0/1)
	XXyX			Sensor failure (0/1)
	XXXy			Last command failed (0/1)
		yXXXXXXXX		VARIO pump failure (0/1)
		XyXXXXXXX		Inlet valve failure (0/1)
		XXyXXXXXX		Coolant valve failure (0/1)
		XXXyXXXXX		Vent valve failure (0/1)

Command	Response by selected communication mode			Description
	CVC 2000	CVC 3000	VACUU·SELECT	
IN_ERR		XXXXyXXXX		Sensor overpressure or underrange warning (0/1)
		XXXXXyXXX		Sensor failure (0/1)
		XXXXXXyXX		External failure on digital I/O module operating as fault indicator interface (0/1)
		XXXXXXXyX		Level sensor limit reached on collection flask (0/1)
		XXXXXXXXy		Last command failed (0/1)
IN_SP_1y[1)2)]	XXXX mbar/hPa/Torr		XXXX.X mbar/hPa/Torr	Set-pressure value at process step y
	X.XXEXX mbar/hPa/Torr (on VS-P sensors)			
IN_SP_2y[1)2)]	XX.X Hz	XXX %		Maximum pump speed of process step y
IN_SP_3y[1)2)]	XXXX mbar/hPa/Torr		XXXX.X mbar/hPa/Torr	Switch-on pressure for VACUU·LAN or hysteresis for vacuum control of process step y
	X.XXEXX mbar/hPa/Torr (on VS-P sensors)			
IN_SP_4y[1)2)]	XX:XX h:m		XX:XX:XX h:m:s	Delay time for VACUU·LAN of process step y
IN_SP_5y[1)2)]	XXXX mbar/hPa/Torr		XXXX.X mbar/hPa/Torr	Maximum (e.g. control or hold vacuum) or minimum value (e.g. pump down) at process step y
	X.XXEXX mbar/hPa/Torr (on VS-P sensors)			
IN_SP_6y[1)2)]	XX:XX h:m		XX:XX:XX h:m:s	Set duration of process step y
IN_APP	X			Selected application's ID → *see chapter: **2.2 Determining the application ID***
IN_PROCESS	X			Selected process (A/B) Switchover only possible on SYNCHRO pumping units.
IN_STEP	X			Currently active process step of the selected application: 0: Process stopped 1..n: Process started
IN_VER	VACUU-SELECT VX.XX / VX.XX			Software version
IN_STAT			yXXXXX	Pump running (0/1)
			XyXXXX	Inlet valve open (0/1)
	yXXX		XXyXXX	Cooling valve open (0/1)
	XyXX		XXXyXX	Vent valve open (0/1)
	XX0y		XXXX0y	VACUU·LAN 0: Inactive 1: Pump down until set pressure 2: Delay time active 3: Monitoring active
	XX1y		XXXX1y	Pump down 0: Inactive 1: Active

Command	Response by selected communication mode			Description
	CVC 2000	CVC 3000	VACUU·SELECT	
	XX2y		XXXX2y	Vacuum control 0: Inactive 1: Actual pressure > Set pressure 2: Actual pressure = Set pressure 3: Actual pressure < Set pressure
	XX3y		XXXX3y	Boiling point detection/Automatic mode 0: Inactive 1: Boiling point detection 2: Boiling tracking 3: Auto termination
	XX4y		XXXX4y	Custom application or additional standard application 0: Inactive 1: Active

6.4.2 Write commands

Command	Parameter	Description
OUT_MODE	0	VACUU·LAN
	1	Pump down
	2	Vacuum control
	3	Boiling point automatic/detection
OUT_APP	0..99	Select a default application. For the complete list of default applications → *see chapter: 2.3 List of applications*
	100..n	Select custom application. Get the application ID from its description in the controller
OUT_PROCESS	A	Select process A
	B	Select process B (only available on SYNCHRO pumping units)
OUT_STEP	*no parameters*	Go to the application's next process step. Only available, when jumping to the next step is enabled
OUT_CFG	yXXX	0..D: Select language (hexadecimal) See list from command „IN_CFG"
	XyXX	Select pressure unit See list from command „IN_CFG"
	XXyX	Switch off/on autostart (0/1)
	XXXy	Switch off/on beep on failure/warning (0/1)
OUT_SP_1y[1),3),4)]	XXXX or XXXX.X X.XXEXX (on VS-P sensors)	Change the set-pressure of process step y without use the vent valve for vacuum control
OUT_SP_Xy[1),3),4)]		Change the set-pressure of process step y with one-time ventilation to the new set pressure, if necessary
OUT_SP_Vy[1),3),4)]		Change the set-pressure of process step y and permanently use the vent valve for vacuum control
OUT_SP_2y[3)]	XX.X XXX	Change maximum pump speed of process step y. Values given in Hz (Communication mode CVC 2000) or % (CVC 3000 or VACUU·SELECT)
OUT_SP_3y[1),3)]	XXXX or XXXX.X X.XXEXX (on VS-P sensors)	Change switch-on pressure for VACUU·LAN or hysteresis for vacuum control of process step y
OUT_SP_4y[3)]	XX:XX or XX:XX:XX	Change delay time for VACUU·LAN of process step y

Command	Parameter	Description
OUT_SP_5y[1),3)]	XXXX or XXXX.X X.XXEXX (on VS-P sensors)	Change maximum (e.g. control or hold vacuum) or minimum value (e.g. pump down) of process step y
OUT_SP_6y[3)]	XX:XX or XX:XX:XX	Change set duration of process step y
OUT_SENSOR	1..8	Select active sensor for control. Order of numbering as shown in chart view screen.
OUT_VENT[4)]	0	Close vent valve
	1	Open vent valve
	2	Vent to atmospheric pressure (ATM) or max. 1060 mbar
REMOTE	0	Exit remote control
	1	Enable remote control. Local operation disabled and process screen active.
	1y	Optional parameter y: 0: Go to process screen 1: Go to chart view screen
REMOTE	2	Enable remote control. Local operation disabled, but exit is possible by ON/OFF button.
	2y	Optional parameter y: 0: Go to process screen 1: Go to chart view screen
START	*no parameters* or 1	Start process
STOP	*no parameters* or 0	Stop process and acknowledge failures/warnings
	1	Stop process
ECHO	0	Disable Echo
	1	Enable Echo. Write commands will be responded
CVC	2	Communication mode CVC 2000
	3	Communication mode CVC 3000
	4	Communication mode VACUU·SELECT

1) Details subject to device default setting for pressure unit.

2) Parameter y is optional. If y is not indicated the parameter value is selected by the current process step of the running application. If no application was started, the first process step is read from.

3) Parameter y is optional. If y is not indicated the parameter value is applied to the current process step of the selected application.

4) The basic settings regarding use of the venting valve are temporarily overwritten (until the end of the process currently running).

5) Any process that may be running is stopped when the command is implemented.

6.5 Application example

Vacuum control

The following sequence shows vacuum control to a pressure setpoint of 12.3 mbar, by way of example.

Input the basic settings

Command	Response	Description
ECHO 1<CR>	1<CR><LF>	Activate Echo mode
CVC 4<CR>	4<CR><LF>	VACUU·SELECT communication mode

Activate remote control

Command	Response	Description
REMOTE 1<CR>	1<CR><LF>	Activate remote control, device locally locked, jump to *Process screen*
	or	
REMOTE 2<CR>	2<CR><LF>	Activate remote control, device can be locally unlocked, jump to *Process screen*
	or	
REMOTE 11<CR>	11<CR><LF>	Activate remote control, device locally locked, jump to *Pressure graph*

Select "vacuum control" application (ID=6)

Command	Response	Description
OUT_APP 6<CR>	6<CR><LF>	Select application

Specify pressure setpoint

Command	Response	Description
OUT_SP_1 12.3<CR>	0012.3<CR><LF>	Adjust pressure setpoint to 12.3 mbar

Start process

Command	Response	Description
START<CR>	1<CR><LF>	Start process

Stop process

Command	Response	Description
STOP<CR>	0<CR><LF>	Stop process

Deactivate remote control

Command	Response	Description
REMOTE 0<CR>	0<CR><LF>	Deactivate remote control

While the process is running, the following Modbus registers can, for example, serve to monitor the process flow:

Command	Response	Description
IN_PV_1<CR>	0123.4 mbar<CR><LF>	Actual pressure of sensor
IN_PV_3<CR>	00:12:34 h:m:s<CR><LF>	Process runtime

7 Index

A
Abbreviations 6
Application example, Modbus TCP 37
Application example, RS-232 51
Application IDs Standard applications 10
Available parameters 21
Available StatusCodes 21

C
Commonly used commands, RS-232 45
Common model Modbus register mappings ... 31
Communication examples 29
Context menu, serial port 40
Control model 32

D
Device locked 30, 33

E
Endpoints and sessions 17
Ethernet 5
Ethernet connection 13
Example – read 29
Example – write 29
Example – write multiple 30

F
Floating-point representation 26

G
General communication settings 40

I
Integer representation 26
Interface commands 44

L
LADS BrowseNames 17
LADS Namespaces 17
LADS OPC UA default settings 15, 16
LADS OPC UA TCP 15
LADS Supported components 22
LADS Template management 20
LADS Templates 19
LADS Template transfer 20
LADS terminology 15
Lock methods 18
Lock status 19

M
Methods to execute templates 19
Modbus data types 25
Modbus register mapping 31
Modbus remote control full access 24
Modbus TCP default settings 23

N
NaN values 25
Network/IP settings 14
Null modem cable 39

P
Personnel qualification 7
Plug allocation (RS-232) 39
Process flow 52
Process parameter deactivation 27, 42
Process step ID 11
Process step identification number 11
Process step IDs 11

R
Read commands, RS-232 46
Representation/setting of pressure values 26
Representation/setting of special values .. 27, 42
Responsibility matrix 7
Results and CSV files 21
RS-232 connection 39

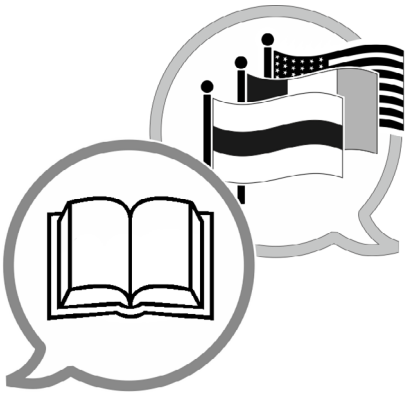
S
Select application via ID 9
Serial port settings 41
Serial port with full access 41
Settings Modbus remote control 23
Show application ID on the display 9
Stability rules 17
Start/end remote control 28, 42
Supported function codes 25
Supported software version 6
SYNCHRO pumping units 36

U
Unlocking by ON/OFF 33
USB adapter/RS-232 39

V
VACUUBRAND standard applications 9
View of locked controller 28, 43

W
Write commands, RS-232 49
Write commands; RS-232 45

VACUUBRAND®



Manufacturer:

VACUUBRAND GMBH + CO KG
Alfred-Zippe-Str. 4
97877 Wertheim
GERMANY

Phone:
Head office +49 9342 808-0
Sales +49 9342 808-5550
Service +49 9342 808-5660

Fax: +49 9342 808-5555
Email: info@vacuubrand.com
Web: www.vacuubrand.com

Version: 20901392_EN_Vsel-Modbus+RS232_LADS_V2_170926

– TDD1 – © VACUUBRAND GMBH + CO KG – 2026