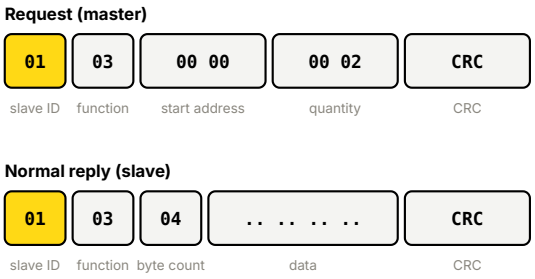
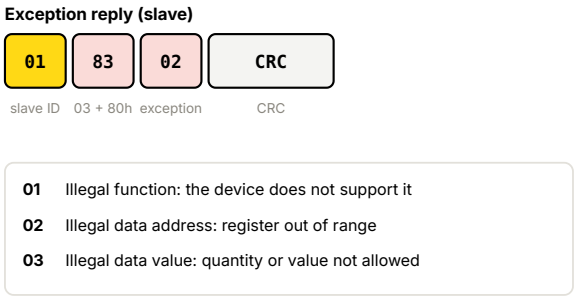
Modbus RTU timeout? 6 checks before blaming the wiring

When a Modbus RTU master gets no reply, the wiring may be fine. Check addressing, serial settings, the single master rule, timing and exception codes first, then the RS-485 physical layer. Valid for any PLC acting as master and any slave device.

What a healthy exchange looks like



Read holding registers (function 03): the slave answers with the same ID and function code, the data and a CRC. No answer at all means the request never arrived or was not accepted.



Exception reply: function code plus 80h and an exception code. The link works: the problem is the request, not the cable.

A timeout is not a diagnosis

The master only knows that no valid reply arrived in time. Wrong ID, wrong settings, a collision, a short timeout or a physical fault all look the same from the master side.

One change at a time

Test with one slave and one known register. Change one setting, test, and note the result before changing the next one.

Six checks, in this order

Check	What to do	Good result
1. Address	Poll the slave ID set in the device. Slave IDs go from 1 to 247; 0 is broadcast and gets no reply.	Each device answers only to its own ID, no duplicates on the bus.
2. Serial settings	Compare baud rate, parity, data bits and stop bits of master and every slave, from each manual.	Exactly the same settings on all devices.
3. One master	Check that no SCADA, HMI or configuration tool is also polling the bus.	Only one device sends requests.
4. Timing	Increase the response timeout and test again. The device manual may state its response time.	The reply arrives before the timeout.
5. Exception code	Read the exception code and check the function code and the register address (0-based in the frame).	A normal reply with the requested data.
6. Physical layer	A/B wires, 120 Ω termination at both ends, bias at one point. See the RS-485 Know-How.	The bus passes the RS-485 checks.

What the symptoms tell you

Symptom	Possible causes	Next step
No slave answers	Serial settings, master port not enabled or in the wrong mode, A/B swapped, no termination or bias.	Checks 2 and 6.
One slave does not answer, the others do	Wrong or duplicate ID, different settings on that device, its wiring.	Checks 1 and 2, then its connection.
Exception 01	The device does not support that function code.	Use a function listed in its manual.
Exception 02	Register out of range, often an offset of one between manual and frame.	Check 5: try address −1.
Exception 03	Quantity or value not accepted by the device.	Read fewer registers; check the allowed range.
Timeouts from time to time	Timeout too short, a second master, polling too fast, noise.	Checks 3 and 4, then 6.
Reply arrives but values are wrong	Register offset of one, or 32-bit values with a different word order.	Compare with a known value from the manual.

From the specification

- Slave addresses 1 to 247; 0 is broadcast (no reply); 248 to 255 are reserved.
- RTU frames are separated by a silence of at least 3.5 character times.
- A read holding registers request (03) can ask for up to 125 registers.

Register numbering

Many manuals use the 1-based convention: 40001 is the first holding register, 30001 the first input register. In the frame, the first address is 0.

Other manuals already give 0-based addresses. If in doubt, read a register with a known value.

If it has never worked

- Test one slave alone with a PC, a USB to RS-485 adapter and a Modbus master tool.
- Swap A and B at one device, following the + and − marks of each manual.
- Check that the PLC port is enabled and in RS-485 mode. This depends on your PLC range.

Serial defaults

The Modbus serial line specification sets 19200 bps and even parity as default, and 2 stop bits when no parity is used.

Many devices ship with other settings, such as 9600 8N1. What matters is that all devices match.

References: Modbus Application Protocol Specification V1.1b3 and Modbus over Serial Line Specification and Implementation Guide V1.02 (Modbus Organization). Device manuals take precedence over this general guide.