

Basic Network Control of the SCF Drive using ModBus™ Communications

Follow these Quick Start instructions to use ModBus Communications for basic network control of an SCF drive. The ModBus Communication module is available as an option for the SCF drive and supports ModBus RTU. For more information refer to the SCF ModBus Reference Guide (RG-SFMOD) and "Getting Started with ModBus RTU" (AN0025). To download this product manual or application note visit the AC Tech Technical Library at <http://www.lenze-actech.com>.

For Further Reference

Modbus-IDA: <http://www.modbus-IDA.org>
AC Tech Library: <http://www.lenze-actech.com>

Quick Start Instructions:

1. Set Parameter 14 (P14) to 01 to enable serial communication.
2. Set P43 to the desired network address that the Modbus master will poll. Valid Modbus addresses are 1-247.
TIP - Avoid using address 1. Most Modbus devices ship with a default address of 1. Duplicate addressing on a Modbus network is not allowed. It can lead to conflicts when replacing and commissioning nodes. To avoid this, do not set the slave address to 1.
3. Set the Modbus master to 9600 baud. No other baud rates are supported by the SCF drive.
4. The SCF series drive has the provision for a watchdog timer to monitor network communications to the drive. The timer is hard coded to a value of 10 seconds. If the drive is under network control and the master does not communicate with the SCF drive for longer than the 10 second timeout period, the drive will STOP. The timer can either be enabled or disabled as outlined in the next step:
5. The drive needs to have its network data formatting set the same as the Modbus master.
 - a. If the Modbus master is set to use 8 data bits, no parity and two stop bits and you want the watchdog timer on the SCF drive ENABLED, set P15 to 02.
 - b. If the Modbus master is set to use 8 data bits, no parity and two stop bits and you want the watchdog timer on the SCF drive DISABLED, set P15 to 03.
 - c. If the Modbus master is set to use 8 data bits, even parity and one stop bit and you want the watchdog timer on the SCF drive ENABLED, set P15 to 04.
 - d. If the Modbus master is set to use 8 data bits, even parity and one stop bit and you want the watchdog timer on the SCF drive DISABLED, set P15 to 05.
 - e. If the Modbus master is set to use 8 data bits, odd parity and one stop bit and you want the watchdog timer on the SCF drive ENABLED, set P15 to 06.
 - f. If the Modbus master is set to use 8 data bits, odd parity and one stop bit and you want the watchdog timer on the SCF drive DISABLED, set P15 to 07.

Drive Control:

1. Start/Stop Control: Please be advised that while the drive is under network control the local STOP circuit is always enabled. Input 1 needs to be asserted in order for the drive to start. If you will be starting/stopping the drive solely via network control, simply jumper the TB1 input to TB2.
2. Use either Modbus function code 16 with a length of 1 or Modbus function code 06 to perform any writes to the drive.
3. Unlocking the Drive: The first write necessary to the drive to perform any function (start, change speed, change a parameter, etc) needs to be an unlock.

To only control the drive over the network you can write a value of 0 to the drive's Modbus register 40049 (AC Tech register 48).

To both control the drive and alter any programming parameters then write the drive's programming password to Modbus register 40049. The default password for the SCF drive is 225.

You should only need to send the unlock command once after power up. As long as the communications do not timeout you should not need to write another unlock to the drive before writing any other function.

- Setting the Drive to Manual Mode: In order for the drive to respond to speed commands written to the keypad speed register the drive must be put into manual mode. To do this write a value of 200H to Modbus register 40002 (the drive's control register). This step is not necessary if there are no other speed references setup on any TB-13x terminal.

Basic Drive Commands:

The following are the basic drive commands. ONLY ONE OF THESE CAN BE DONE AT A TIME:

- To STOP the drive using the stop method programmed in P04, write a value of 0004hex to Modbus register 40002 (AC Tech register 1).
- To Start the drive write a value of 0008hex to Modbus register 40002.
- To Set Reverse direction write a value of 0040hex to Modbus register 40002. P17 must be set to 2 in order for this command to be accepted.
- To Set Forward direction (the drive powers up with forward direction already selected) write a value of 0080hex to Modbus register 40002.
- If you want the network to control the speed of the drive, set P05 (Standard Speed Source) to 01 (Keypad) and write the speed to the Keypad Speed Command Register, Modbus register 40041 (AC Tech register 40). Speed is written in 0.1Hz (so 412 would be 41.2 Hz). In this mode the drive's initial speed reference on power up will be the last speed written to the drive.

Basic Drive Status:

AC Tech register 24 is a 6 word entity containing the drive's status information. To read the entire status block use Modbus function code 3 with a length of 6 to read starting at Modbus register number 40025. The low byte of the third word in this block of data contains the operational status. If this is the only data you want you can use Modbus function code 3 with a length of 1 to read register 40027.

The value of that low byte of data corresponds to the following operational states:

Operational Status (byte D3L or Register #26 DL)

0	FAULT LOCKOUT
1	FAULT
2	START PENDING
3	STOP
4	DC BRAKE
5	RUN AT 0Hz
6	RUN
7	ACCEL
8	DECEL
9	CURRENT LIMIT
10	DECEL OVERRIDE
11	LOWER TRANSISTORS SWITCHING ON

Basic Drive Network Programming:

The programming parameters of the SCF drive may be altered by a Modbus master. To do so simply write the desired value to the appropriate Modbus register. The translation is as follows:

$$\text{Modbus register number} = \text{SCF parameter number} + 51.$$

As an example if you wanted to change the acceleration time (P19) of the SCF drive write the time desired into Modbus register 40070. Note that time is written in 0.1 seconds (so 200 would be 20.0 sec).