modbus\_set\_slave(3)

===================

NAME

----

modbus\_set\_slave - set slave number in the context

SYNOPSIS

--------

\*int modbus\_set\_slave(modbus\_t \*'ctx', int 'slave');\*

DESCRIPTION

-----------

The \*modbus\_set\_slave()\* function shall set the slave number in the libmodbus

context.

The behavior depends of network and the role of the device:

\*RTU\*::

Define the slave ID of the remote device to talk in master mode or set the

internal slave ID in slave mode. According to the protocol, a Modbus device must

only accept message holding its slave number or the special broadcast number.

\*TCP\*::

The slave number is only required in TCP if the message must reach a device on a

serial network. Some not compliant devices or software (such as modpoll) uses

the slave ID as unit identifier, that's incorrect (cf page 23 of Modbus

Messaging Implementation Guide v1.0b) but without the slave value, the faulty

remote device or software drops the requests! The special value

`MODBUS\_TCP\_SLAVE` (0xFF) can be used in TCP mode to restore the default value.

The broadcast address is `MODBUS\_BROADCAST\_ADDRESS`. This special value must be

use when you want all Modbus devices of the network receive the request.

RETURN VALUE

------------

The function shall return 0 if successful. Otherwise it shall return -1 and set

errno to one of the values defined below.

ERRORS

------

\*EINVAL\*::

The slave number is invalid.

EXAMPLE

-------

[source,c]

-------------------

modbus\_t \*ctx;

ctx = modbus\_new\_rtu("/dev/ttyUSB0", 115200, 'N', 8, 1);

if (ctx == NULL) {

fprintf(stderr, "Unable to create the libmodbus context\n");

return -1;

}

rc = modbus\_set\_slave(ctx, YOUR\_DEVICE\_ID);

if (rc == -1) {

fprintf(stderr, "Invalid slave ID\n");

modbus\_free(ctx);

return -1;

}

if (modbus\_connect(ctx) == -1) {

fprintf(stderr, "Connection failed: %s\n", modbus\_strerror(errno));

modbus\_free(ctx);

return -1;

}

-------------------

AUTHORS

-------

The libmodbus documentation was written by Stéphane Raimbault

<stephane.raimbault@gmail.com>