NAME

modbus_write_and_read_registers − write and read many registers in a single transaction

SYNOPSIS

int modbus_write_and_read_registers(modbus_t **ctx*, int *write_addr*, int *write_nb*, const uint16_t **src*, int *read_addr*, int *read_nb*, const uint16_t **dest*);

DESCRIPTION

The *modbus_write_and_read_registers()* function shall write the content of the *write_nb* holding registers from the array *src* to the address *write_addr* of the remote device then shall read the content of the *read_nb* holding registers to the address *read_addr* of the remote device. The result of reading is stored in *dest* array as word values (16 bits).

You must take care to allocate enough memory to store the results in *dest* (at least *nb* * sizeof(uint16_t)).

The function uses the Modbus function code 0x17 (write/read registers).

RETURN VALUE

The *modbus_write_and_read_registers()* function shall return the number of read registers if successful. Otherwise it shall return −1 and set errno.

ERRORS

EMBMDATA

Too many registers requested, Too many registers to write

SEE ALSO

modbus_read_registers(3) **modbus_write_register**(3) **modbus_write_registers**(3)

AUTHORS

The libmodbus documentation was written by Stéphane Raimbault <**stephane.raimbault@gmail.com**[1]>

NOTES

1. stephane DOT raimbault AT gmail DOT com
   mailto:stephane DOT raimbault AT gmail DOT com