

**NAME**

`modbus_new_rtu` – create a libmodbus context for RTU

**SYNOPSIS**

```
modbus_t *modbus_new_rtu(const char *device, int baud, char parity, int data_bit, int stop_bit);
```

**DESCRIPTION**

The `modbus_new_rtu()` function shall allocate and initialize a `modbus_t` structure to communicate in RTU mode on a serial line.

The `device` argument specifies the name of the serial port handled by the OS, eg. `/dev/ttyS0` or `/dev/ttyUSB0`. On Windows, it's necessary to prepend COM name with `\.\.` for COM number greater than 9, eg. `\.\.\COM10`. See [http://msdn.microsoft.com/en-us/library/aa365247\(v=vs.85\).aspx](http://msdn.microsoft.com/en-us/library/aa365247(v=vs.85).aspx) for details

The `baud` argument specifies the baud rate of the communication, eg. 9600, 19200, 57600, 115200, etc.

The `parity` argument can have one of the following values

- `N` for none
- `E` for even
- `O` for odd

The `data_bits` argument specifies the number of bits of data, the allowed values are 5, 6, 7 and 8.

The `stop_bits` argument specifies the bits of stop, the allowed values are 1 and 2.

**RETURN VALUE**

The `modbus_new_rtu()` function shall return a pointer to a `modbus_t` structure if successful. Otherwise it shall return `NULL` and set `errno` to one of the values defined below.

**ERRORS****EINVAL**

An invalid argument was given.

**EXAMPLE**

```
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;
}
```

**SEE ALSO**

`modbus_new_tcp(3)` `modbus_free(3)`

**AUTHORS**

The libmodbus documentation was written by Stéphane Raimbault  
[stephane.raimbault@gmail.com<sup>\[1\]</sup>](mailto:stephane.raimbault@gmail.com)

**NOTES**

1. stephane DOT raimbault AT gmail DOT com  
[mailto:stephane DOT raimbault AT gmail DOT com](mailto:stephane.DOTraimbault@ gmail.COM)

