

How to use a serial terminal

- I. Identify the device for the serial communication.....2
 - I.1. Linux users..... 2
 - I.2. Windows users..... 2
- II. Tera Term (Windows)..... 3
- III. PuTTY (Windows / Unix)..... 4
- IV. GTKTerm (Linux)..... 5
- V. Minicom (Console Linux)..... 6

The serial terminal is an essential tool in a developer’s life, just like the multimeter and the scope are essential to a hardware electronics engineer.

With a serial terminal one can send data from an embedded system to a computer in a simple way (only two wires). This is mainly used for log and debug purposes.

Here is how to use some of the most famous serial terminals.



Except where otherwise noted, this work is licensed under <https://creativecommons.org/licenses/by-nc-sa/4.0/>

I. IDENTIFY THE DEVICE FOR THE SERIAL COMMUNICATION

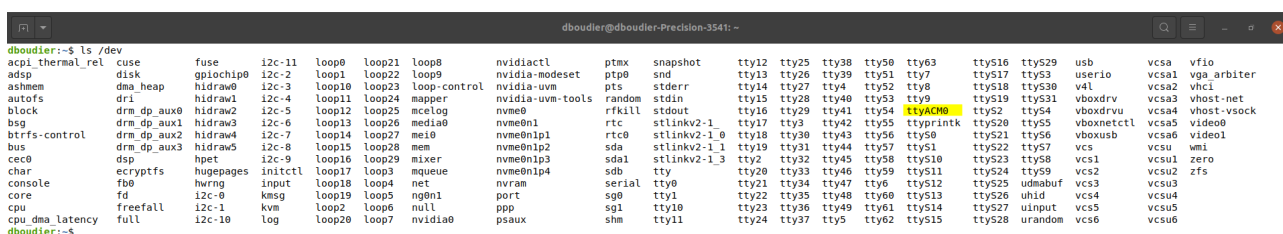
Before using a serial terminal, we must find out the name of the connected device.

I.1. Linux users

Start a Linux shell: `Ctrl+Alt+T`

First, before plugging in the device: `ls /dev`

After having plugged the device in: `ls /dev`



```

duboulier@dboudier-Precision-3541: ~
duboulier:~$ ls /dev
acpi_thermal_rel  cuse          fuse          i2c-11       loop0        loop21       loop8        nvidiactl    ptmx         snapshot     tty12        tty25        tty30        tty50        tty63        ttyS16       ttyS29       usb          vcsa        vfio
adsp             disk          gpiochip0    i2c-2        loop1        loop22       loop9        nvidia-modeset  ptp0         snd          tty13        tty26        tty39        tty51        tty7         ttyS17       ttyS3        userio      vcsa1      vga_arbiter
ashmem          dma_heap      hidraw0      i2c-3        loop10       loop23       loop-control  nvme0         pts          stderr       tty14        tty27        tty4         tty52        tty8         ttyS18       ttyS30       v4l         vcsa2      vhci
autofs          drm_dp_aux0   hidraw1      i2c-4        loop11       loop24       mapper       nvidia-vm-tools  random       stdin       tty15        tty28        tty40        tty53        tty9         ttyS19       ttyS31       vboxdrv    vcsa3      vhost-net
block           drm_dp_aux1   hidraw2      i2c-5        loop12       loop25       mcelog       nvme0         rkill        stdout       tty16        tty29        tty41        tty54        ttyACM0      ttyS2        ttyS4        vboxdrv    vcsa4      vhost-vsock
bsg             drm_dp_aux1   hidraw3      i2c-6        loop13       loop26       media0       nvme0         rtc          stlinkv2-1_  tty17        tty3        tty42        tty55        ttyprintk    ttyS20       ttyS5        vboxnetctl vcsa5      video0
btrfs-control   drm_dp_aux2   hidraw4      i2c-7        loop14       loop27       mei0         nvme0n1       rtc0         stlinkv2-1_0  tty18        tty30        tty43        tty56        tty50        ttyS21       ttyS6        vboxusb    vcsa6      videol
bus             drm_dp_aux3   hidraw5      i2c-8        loop15       loop28       mem          nvme0n1p2     sda          stlinkv2-1_1  tty19        tty31        tty44        tty57        tty51        ttyS22       ttyS7        vcs        vcsu       wmi
cec0            dsp           hpet         i2c-9        loop16       loop29       mixer        nvme0n1p3     sda1         stlinkv2-1_3  tty2         tty32        tty45        tty58        tty510       ttyS23       ttyS8        vcs1       vcsu1     zero
char            ecryptfs     hugepages    initctl      loop17       loop3        mqueue       nvme0n1p4     sdb          tty           tty20        tty33        tty46        tty59        tty511       ttyS24       ttyS9        vcs2       vcsu2     zfs
console         fb0           hwrng        input        loop18       loop4        net          nvram         serial        tty0          tty21        tty34        tty47        tty6         tty512       ttyS25       udmabuf    vcs3       vcsu3
core            fd            i2c-0        kmsg         loop19       loop5        ng0n1        port          sg0          tty1          tty22        tty35        tty48        tty60        tty513       ttyS26       uhid        vcs4       vcsu4
cpu             freefall     i2c-1        kvm          loop2        loop6        null         ppp           sgl          tty10         tty23        tty36        tty49        tty61        tty514       ttyS27       uinput     vcs5       vcsu5
cpu_dma_latency full          i2c-10       log          loop20       loop7        nvidia0      psaux         shm          tty11         tty24        tty37        tty5         tty62        tty515       ttyS28       urandom    vcs6       vcsu6
duboulier:~$

```

The device name should appear on the second display, but not on the first one.

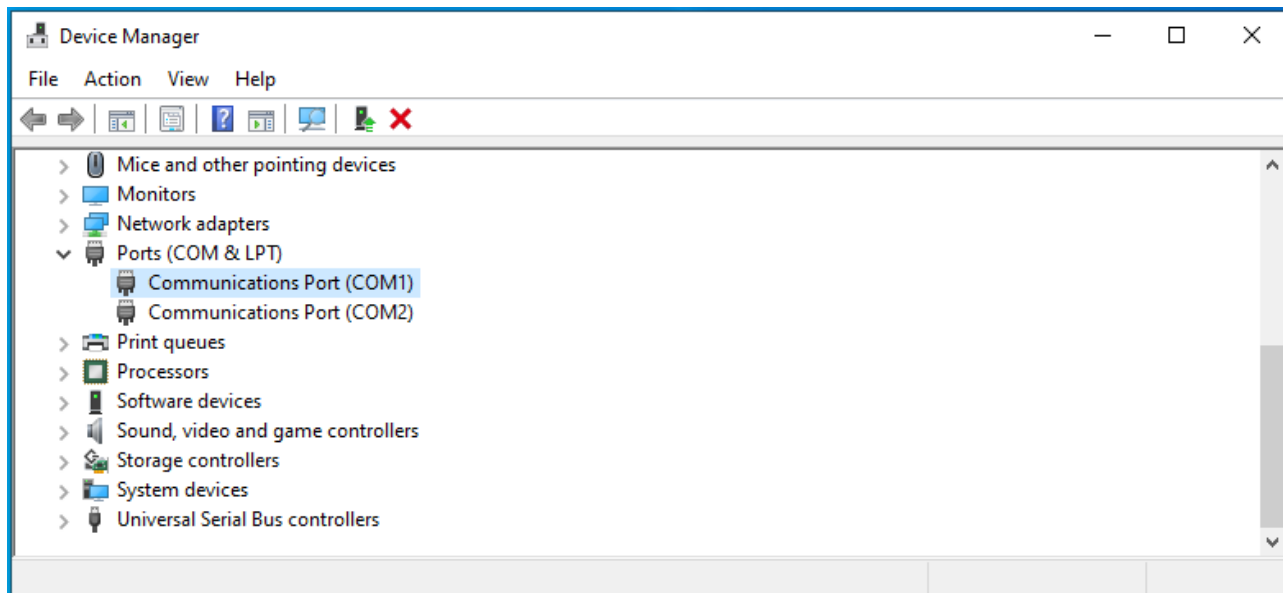
The full device name is usually `dev/ttyACM0` or `/dev/ttyUSB0`.

I.2. Windows users

Open the **Gestionnaire de périphériques / Device Manager**.

Observe the devices in **Ports (COM & LPT)**.

Unplug and plug the device to make sure of its name.



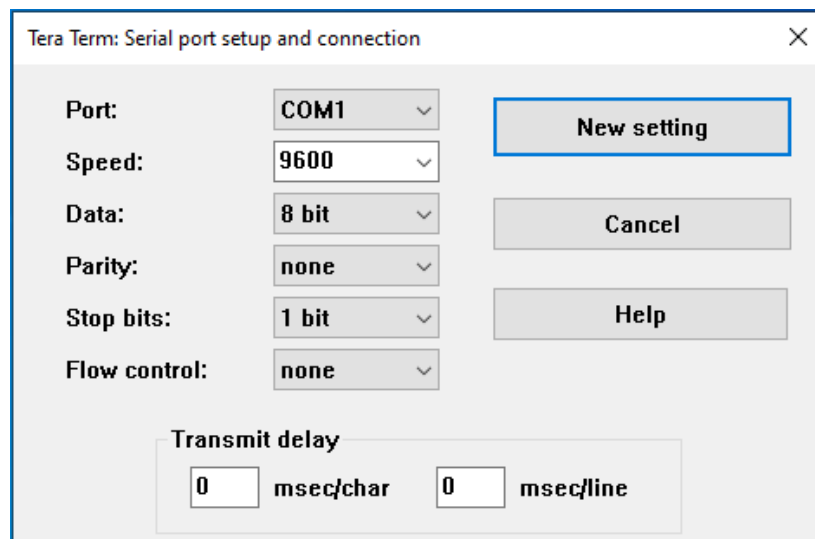
II. TERA TERM (WINDOWS)

Launch Tera Term

At Tera Term's startup, select the device's com port.

Serial communication parameters

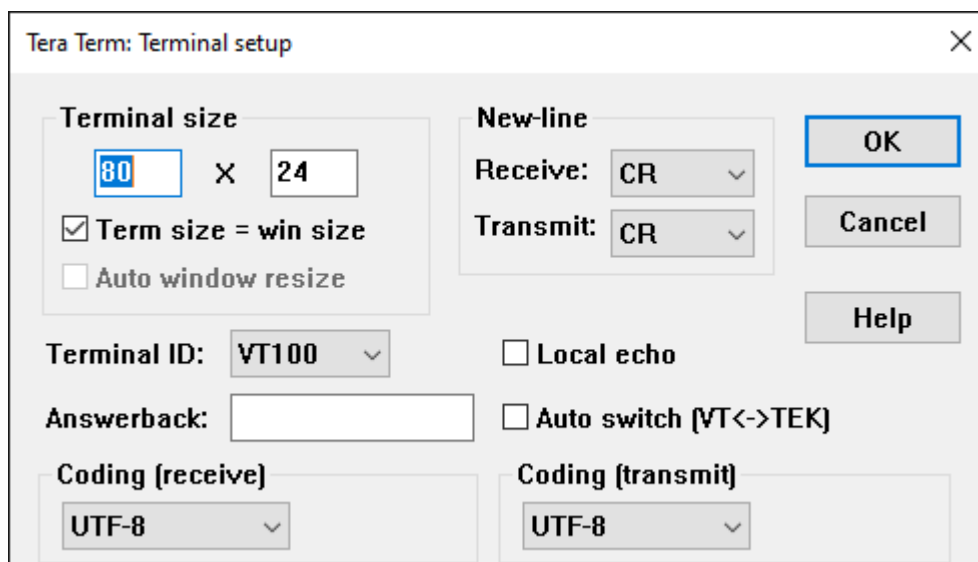
Setup → Serial port...



If you observe line feed or carriage return problems

Setup → Terminal...

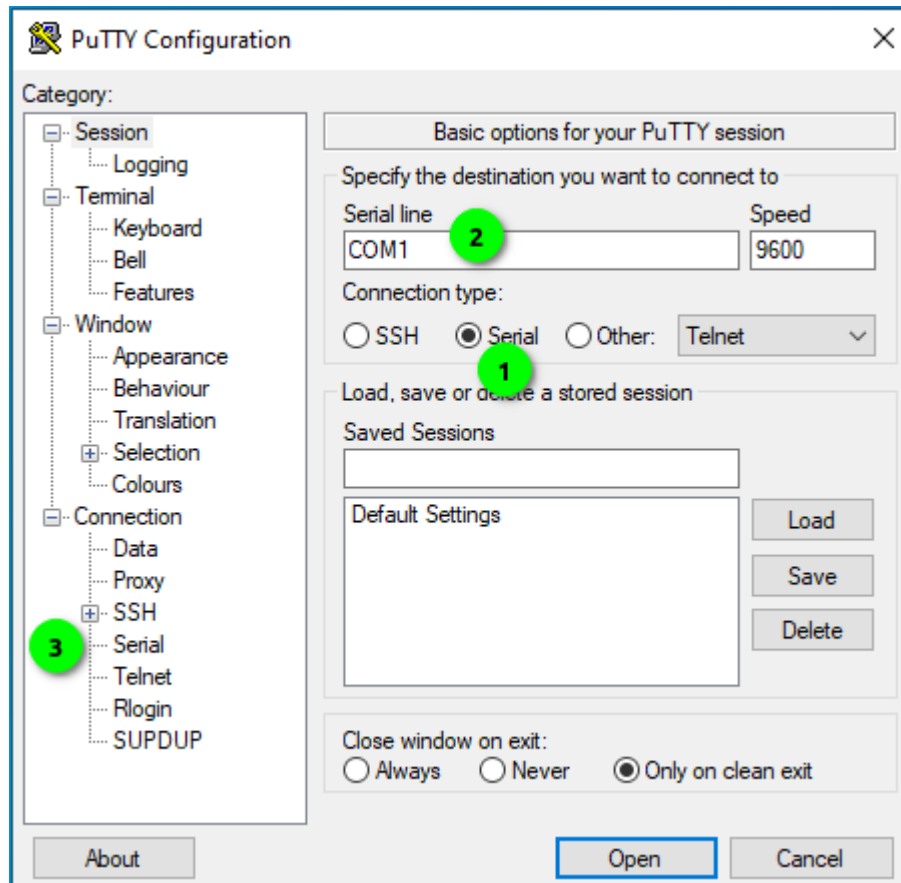
Change the parameters Receive and Transmit to CR+LF or Auto.



III. PuTTY (WINDOWS / UNIX)

Launch PuTTY

In the window, click on `Serial` (1), then fill in the COM port name (2).
DO NOT CLICK ON "Open".



Serial communication parameters

Click on `Connection` → `Serial` (3)

Fill in the communication parameters and click on `Open`.

If you observe line feed or carriage return problems

Right-click on the PuTTY's title bar

→ `Change Settings...` → `Terminal` → `Implicit LF in every CR` → `Apply`

The serial communication parameters are accessible through this menu:

Right-click on the title bar → `Change Settings...` → `Connection` → `Serial`

IV. GTKTERM (LINUX)

Install GTKTerm

From a Linux terminal (Ctrl+Alt+T), with the root privileges:

```
sudo apt-get install gtkterm
```

Launch GTKTerm

From a Linux terminal: `gtkterm`

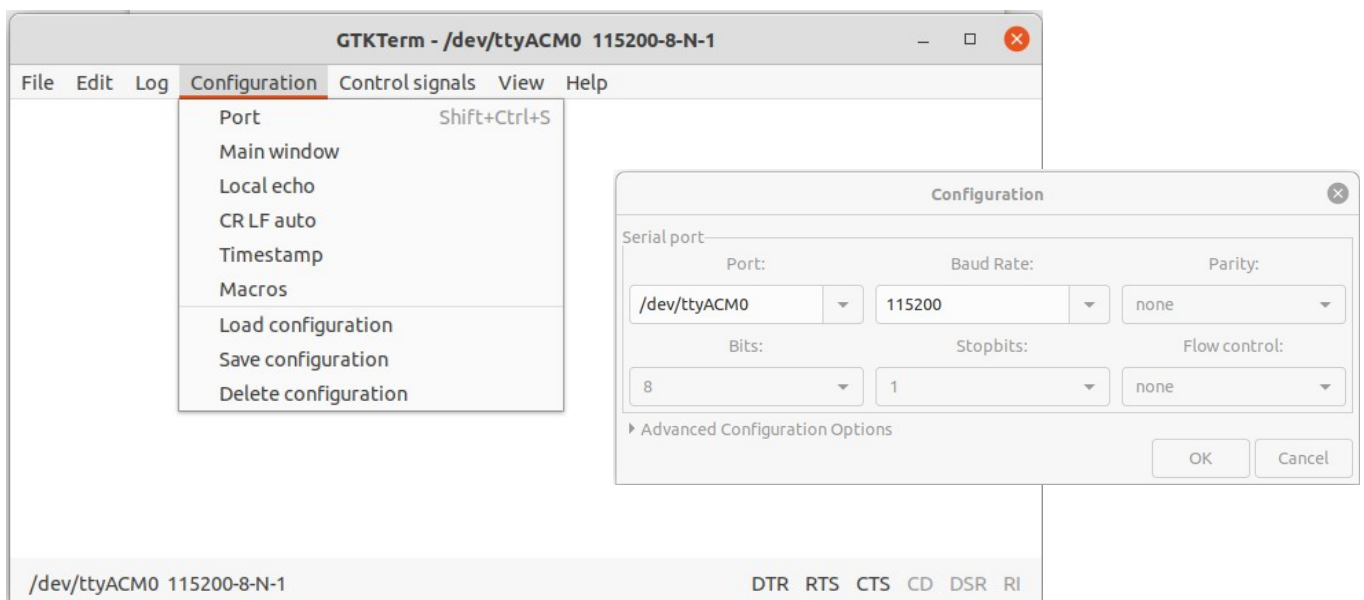
From the graphical interface: Applications → gtkterm

Serial communication parameters

The device name must have been determined first.

Configuration → Port

Fill in the device name (Port) and the communication parameters (*Baudrate, Parity, Bits, Stopbits, Flow control*). Advanced parameters will not be set.



If you observe line feed or carriage return problems

Configuration → CR LF auto (Carriage Return + Line Feed = '\r' + '\n').

V. MINICOM (CONSOLE LINUX)

Install minicom

From a Linux terminal (Ctrl+Alt+T), with the root privileges:

```
sudo apt-get install minicom
```

Launch minicom

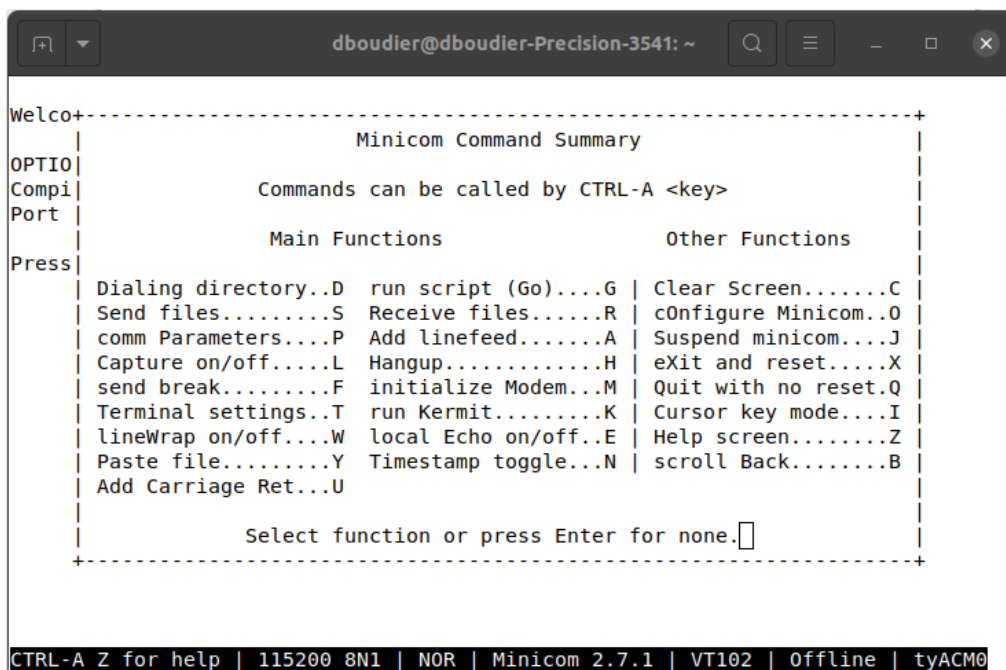
The device name must have been identified first: /dev/tty____.

From a Linux terminal: `minicom -D /dev/tty____`

Accessing the menus

Press Ctrl+'A', then release all keys, then press 'Z'.

The user interface only uses the keyboard (no mouse).



```

dbourier@dbourier-Precision-3541: ~
-----+-----
Welco+
|                                     Minicom Command Summary
|
|                                     Commands can be called by CTRL-A <key>
|
|                                     Main Functions                Other Functions
|
| Press
|
| Dialing directory..D  run script (Go)...G  Clear Screen.....C
| Send files.....S    Receive files.....R  cOnfigure Minicom..O
| comm Parameters....P  Add linefeed.....A  Suspend minicom...J
| Capture on/off....L  Hangup.....H        eXit and reset....X
| send break.....F    initialize Modem...M  Quit with no reset.Q
| Terminal settings..T  run Kermit.....K    Cursor key mode...I
| lineWrap on/off...W  local Echo on/off..E  Help screen.....Z
| Paste file.....Y    Timestamp toggle...N  scroll Back.....B
| Add Carriage Ret...U
|
|                                     Select function or press Enter for none.
|
|-----+-----
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.7.1 | VT102 | Offline | tyACM0

```

Serial communication parameters

Press Ctrl+'A', then release, then press 'O' (cOnfigure Minicom).

Serial port setup → Press the keys according to the parameters to set.

'Enter' to quit, then 'Exit'.

If you observe line feed or carriage return problems

Press Ctrl+'A', release, then 'A' (Add linefeed) or 'U' (Add Carriage Return).