# Nixie documentation

Nixie is a virtual clock-thermometer using a representation of 3D Nixie tubes to display the time and the temperature. It provides great numbers of features to make this one a vintage representation clock with high end capabilities.
Nixie can work on different systems (Raspberry, Windows and Linux) but the full capabilities are valid on the Raspberry PI 3. Moreover, with the official touch screen Nixie is completely autonomous.

The picture below is Raspberry PI 3 with official touch screen photography.
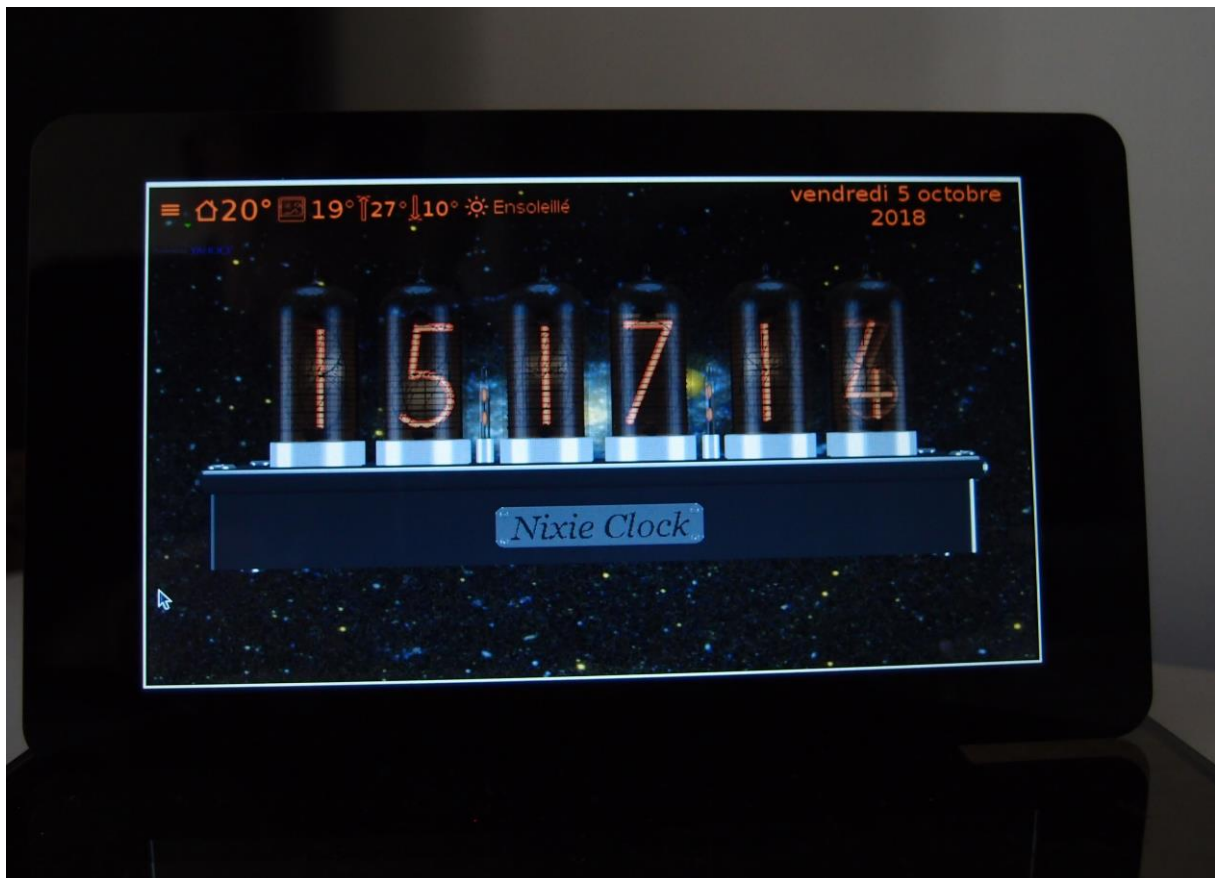
License

Nixie is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License (GPLv3) as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

Nixie is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

Install Nixie

1. **Raspberry PI 3**.
   Use **nixie-rpi3-0.0.0.deb** installer.
   Nixie has been tested on Raspberry PI 3 under Raspbian Stretch with the official 7" touch screen. You must change the default configuration.

   Launch in a terminal **sudo raspi-config.**
   Enable one wire interface. It is mandatory if you want to use DS18b20 sensor for ambient temperature. Nixie tries to detect automatically the sensor identifier. If more than one 1-wire device is connected, only the first is chosen.
   You can force an identifier in the nixie.ini file (section [thermometer], variable sensorId).
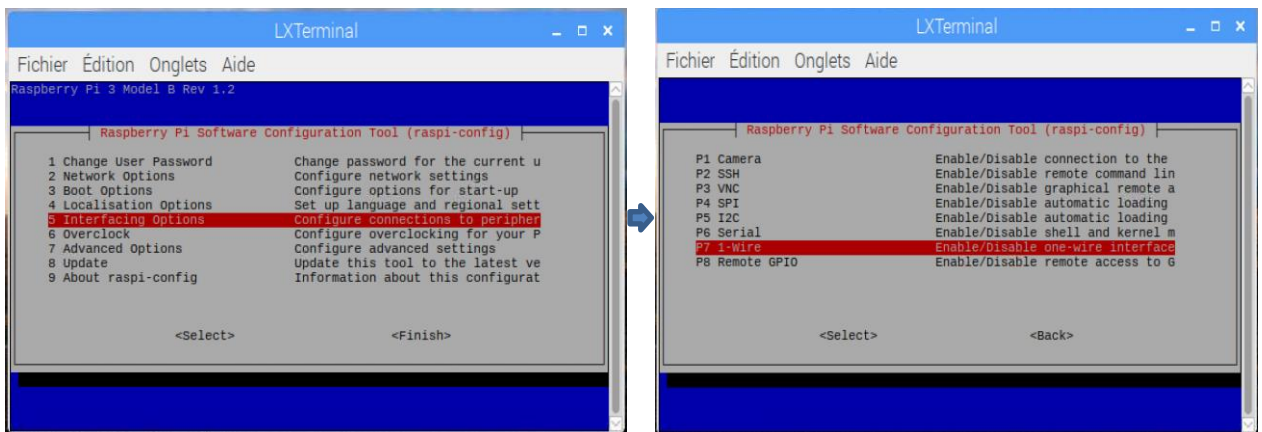
   OpenGL. It is the sensitive part.
   Because the OpenGL driver is experimental, at this time, only OpenGL desktop driver with fake KMS run correctly. The full KMS fails with official 7" touch screen (black screen at this time). The legacy is extremely and really extremely slow and the colors are wrong.
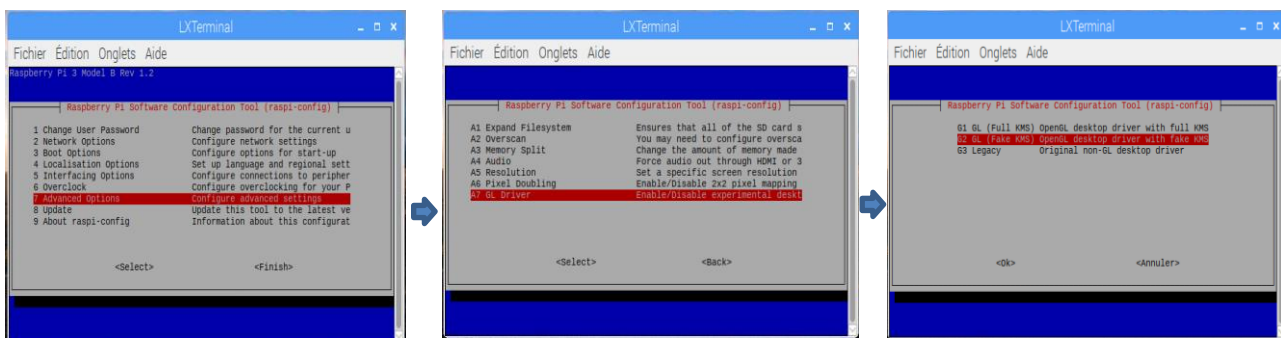
   With fake KMS driver, Nixie runs perfectly. The frame rate is approximately 10fps to 15fps (depend of Nixie tube number). You can test it with the "fast counter mode".
   Warning in the "fast counter mode" the temperature is high rapidly. In other modes there is no problem. The Raspberry is always cold and can work without aeration or cooling.

   By default, the brightness plugin uses a direct file access in the system. For this reason it is important to launch Nixie at **root** level otherwise the backlight files cannot be updated.

If you don't use the installer use **sudo nixie.run** to launch the application.

If you want to start nixie at startup, you can add the line @**/usr/bin/nixie/nixie.run** in the file **/home/pi/.config/lxsession/LXDE-pi/autostart**

1. Windows.
   Use **nixie-win32-0.0.0.exe** installer.
   It has been tested on Windows 10 32 or 64 bits on two laptops.

2. Linux.
   Use **nixie-linux64-x86-0.0.0.deb** on Debian base systems and **nixie-linux64-x86-0.0.0.zip** for the others. Nixie has been tested on Kubuntu 17.10 - Ubuntu 18.04 using VirtualBox and Ubuntu 18.04 native. The brightness does not work with VirtualBox.
   To run correctly on Linux, you must install libssl1.0. The command is:
   sudo apt-get install libssl1.0-dev

Uninstal Nixie

1. Rapsberry and Linux Debian base.
   In a terminal use **sudo apt-get remove nixie**.

2. Windows
   Use standard uninstall applications.

Sensitive areas

The image below shows Nixie at first launch. The 8 blue zones allow to quickly activating some commonly used features.
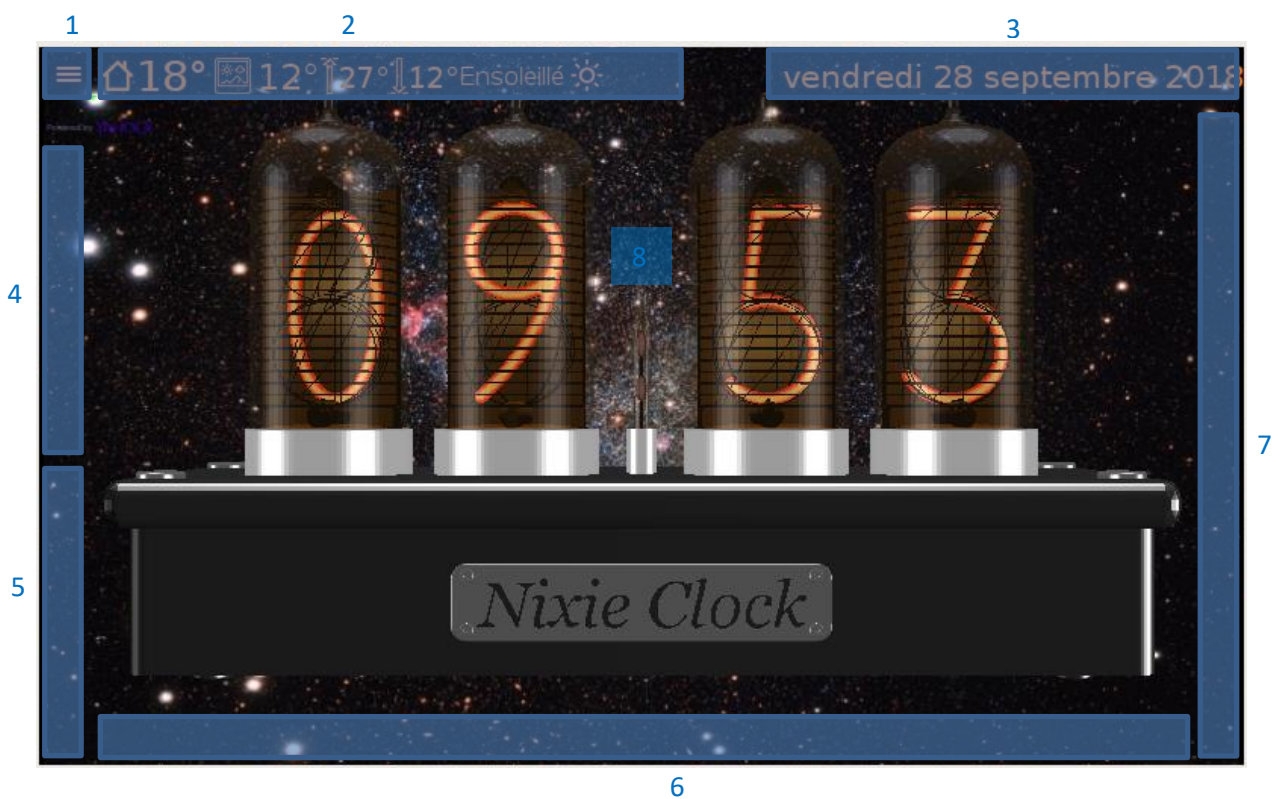
1. All features can be activated from the menu.
2. Show or hide the forecasts.
3. Switch between long and short date.
4. Switch between day or night brightness.
5. Switch between clock and thermometer.
6. Show or hide the audio player.
7. Change the digit brightness.
8. Enable dynamic rotation.
   When an alarm is activated, this zone stops temporarily the alarm. In this case, dynamic rotation is disabled.

Clock

The clock takes time from the system time.
If your computer is connected at internet, it update the time himself.
You can display the Nixie clock with four or six digits.

Weather

Ambiant(*)   Actual   Maximum   Minimum   Description
           ↘   ↓ today   today ↙

Nine day forecasts

(*) The ambient temperature is actually valid only with Raspberry PI 3 with one wire DS18B20 sensor (see the paragraph "Optional hardware (Raspberry PI 3)").

The weather plugin needs a location to retrieve the weather.
At start, the location is detected automatically from the external IP address of your router.
You can change the location using the feature Location in the menu or manually using the setting (see town and country entries).

Alarms

It is possible to create any number of alarms but a maximum of 3 are displayed.

Enable/disable Date Time Name   Repeat days   Add   Change Remove Active(2)

Save and / Repeat(1) hour minute Calendar Days of week Name Cancel
activate

(1) By default, the alarm is repeated 3 times with an interval of 10 minutes.
(2) Click on an active alarm display the alarm characteristics.
When the alarm has been triggered, click disables the alarm.

Timers

It is possible to create any number of timers but a maximum of 3 are displayed.

Enable/disable Delay Remove Change Add Active and displayed(1) Active(2)
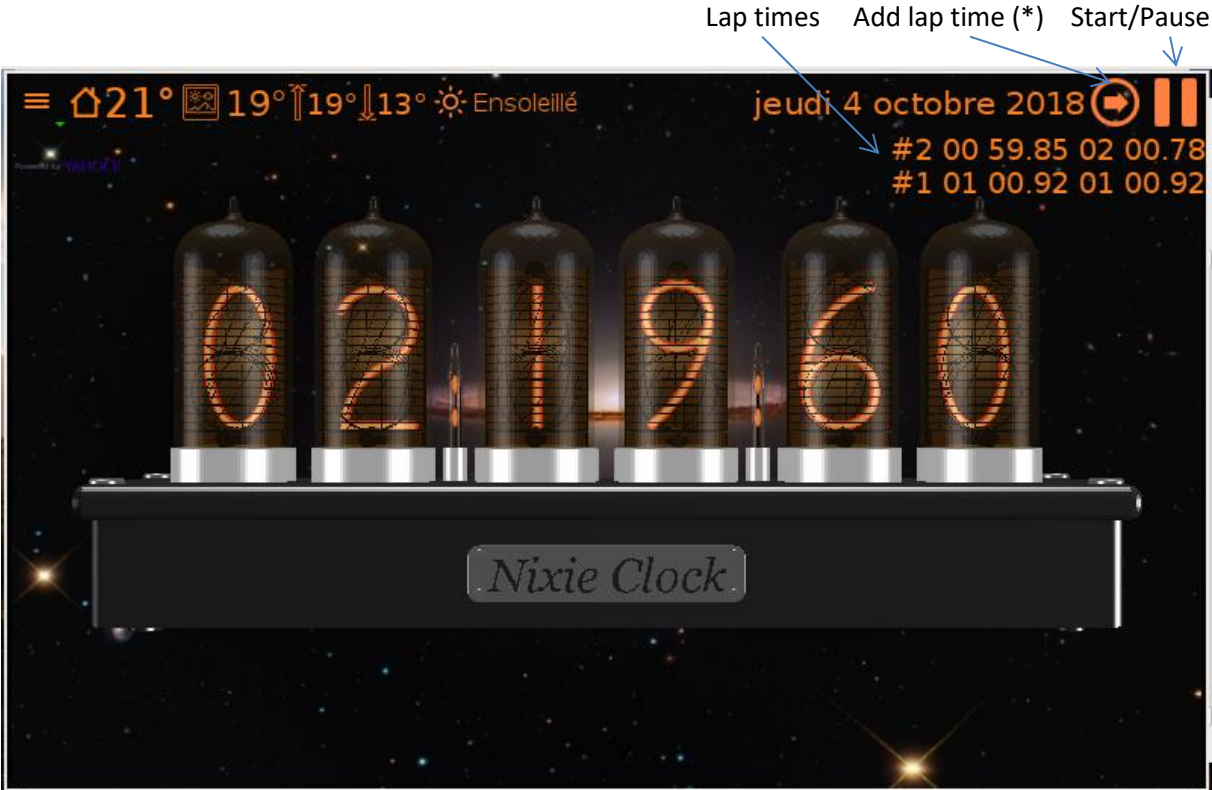
Name Reset Pause/restart

Save and activate minute second Name Cancel

(1) Clock on the displayed timer disable the timer.
(2) Click on an active timer to change the displayed timer.

Chronometer

(*) If pause mode is activated this button reset the chronometer.
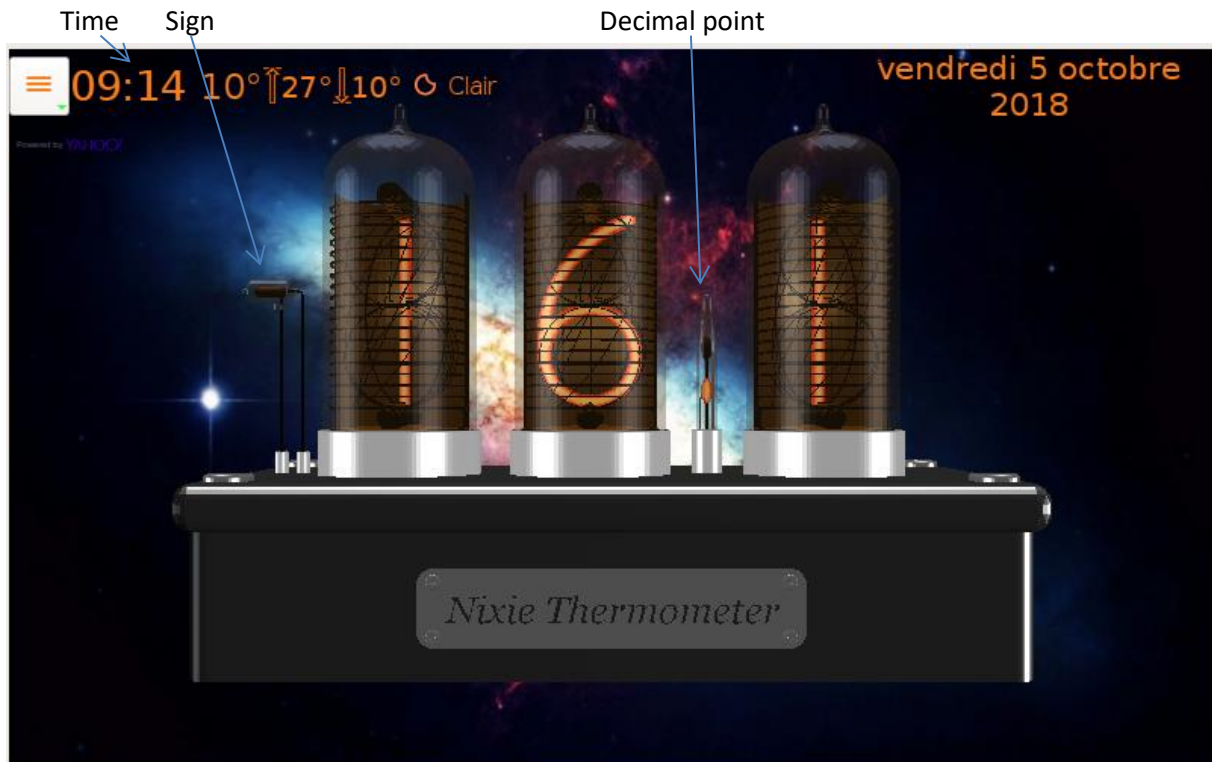
Thermometer

The thermometer measures the ambient temperature. It is implemented actually only for Raspberry PI 3. This feature needs a small electronic circuit. This circuit is described at the paragraph "Optional hardware (Raspberry PI 3)".

You can choose between degree Celsius and Fahrenheit.

Time   Sign   Decimal point

Web radios

Favorites   Search   Remove   Add   Play directly

Station logo   Web radios manager   Play/pause   Volume   Mute   Sleep
Current station
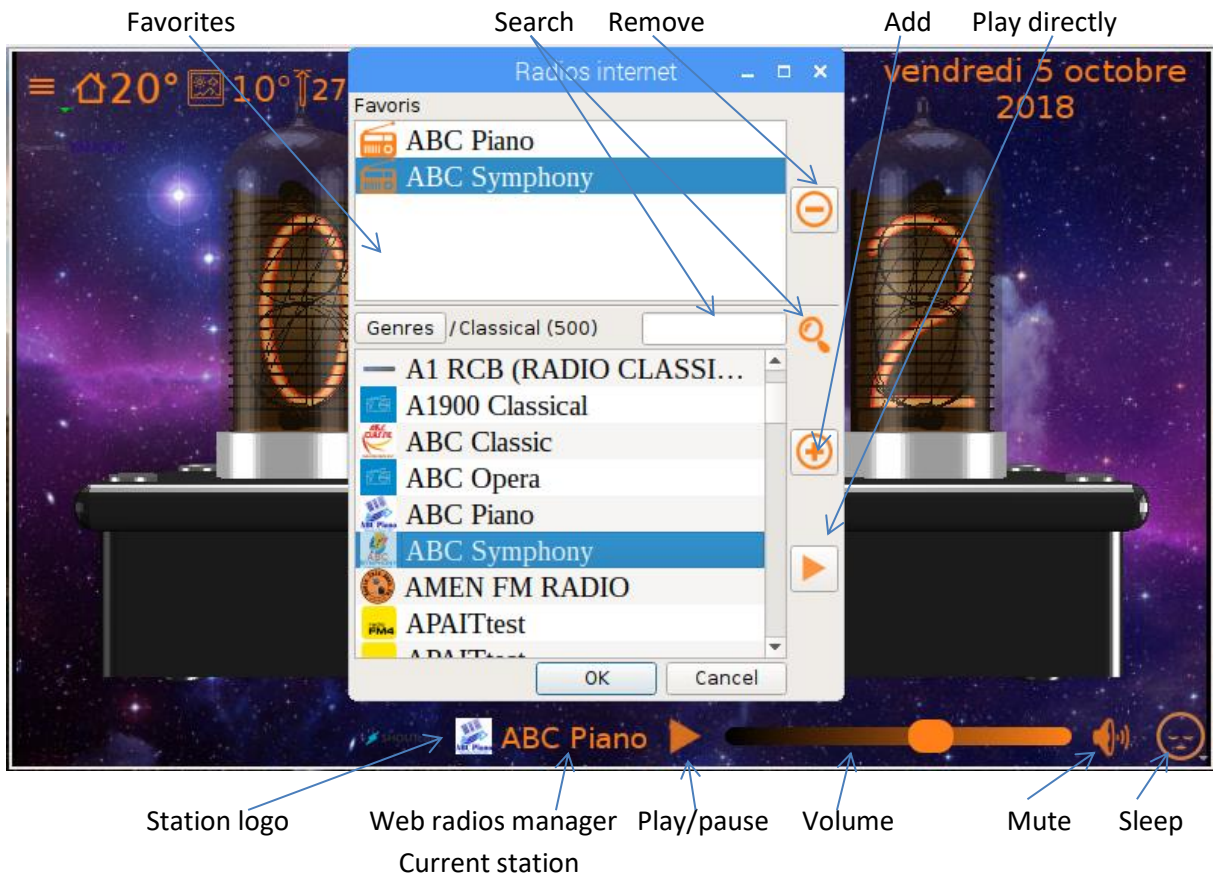
You can add your favorites manually using the nixie.ini file. See the paragraph "settings" section "webradios".

Optional hardware (Raspberry PI 3)

- A **DS18B20** Programmable Resolution1-Wire Digital Thermometer to measure the ambient temperature.

  I used this model, because the resistor is incorporate. The DS18B20 is connected in 3.3V. In my case, the measuring pin is connected at the gpio connector pin number 4.

- A **GL5539** Photo resistor to adapt the brightness at the ambient luminosity.

  The photo resistor is welded on the DS18B20 throw a 45 KΩ resistor. The resistor is connected at the 3.3V pin and the photo resistor pin is connected at the ground. The junction is connected at the gpio connector pin number 17.
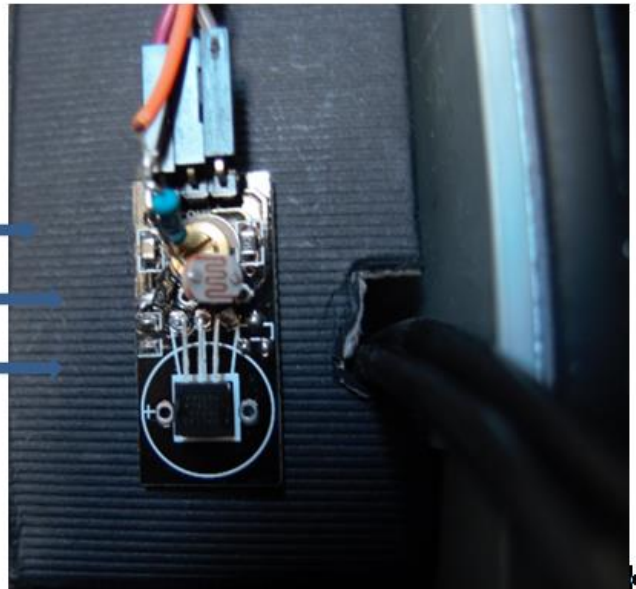
The wiring is very simple:
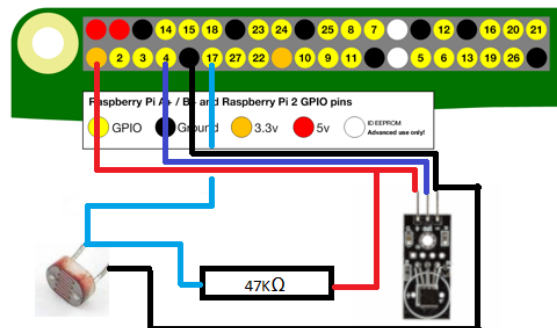
The sensor circuit is at the external left side of the box and hidden if you look at the screen.

Settings

Nixie settings are managed from nixie.ini file. All entries are optional. To validate an entry remove "#" at the left of the entry.

In fact, a subset of settings is changed during the session. This subset will be saved in the user disk space. The position of this file depends of the system.

- Windows: **%APPDATA%\nixie\nixie.ini**.

- Linux: **~/.local/share/nixie/nixie.ini**.

The values followed by '*' will be overridden by the current session.
The values followed by '**' are reserved for internal used.

[Tubes]
display=0*
- 0 Nixie tubes.
- 1 simple text.
- 2 LCD display.
- 3 Leds display.

mode=0*
- 0 Clock.
- 4 Thermometer.

clockTubeNumber=4*
- 3 Thermometer.
- 4 Clock.
- 6 Clock.

backgroundTexture=exe/images/bkgnd.jpg
It's background image file. Active only if diaporama is empty and forceBackgroungUniform=false.

aboutTexture=exe/images/about.png
It's the label on the back side of the box.

tableTexture=exe/images/table.jpg
It's the table texture.

stl=exe/stl
It's stl files folder.

diaporama=exe/diaporama
It's the diaporama images folder.

diaporamaInterval=5000
It's the diaporama interval. By default 5s.

diaporamaMode=1
- 0 the images is adapted at the windows (may be cropped).
- 1 the images is entirely visible (black bands can be created).
  In all cases, the proportions are kept.

backgroundUniformColor=0 0 32 255
If diaporama and backgroundTexture are empty or if forceBackgroungUniform=true, a simple color is used.

forceBackgroungUniform=false*
If true, the backgroundUniformColor is always used.

moodLightPositions=200.0 0.0 100.0**
It's the position of the mood light.

moodLightIntensities=1.0 1.0 1.0
It's the mood light intensities.

partCode=-table –pencil*
- -table: The table is hidden.
- -pencil: The pencil is hidden.

No other code is available. For example, to see the table and not the pencil, it is possible to enter partCode=+table –pencil. The same effect is possible with partCode=-pencil.

fontFamily=*
It's the font family of all texts. For blank value, the system font family is used.

menuFontCoef=0
It's the coefficient of the size of menu text. It's generally used by Raspberry touch screen. 0 means the text size depend of the system text size. An equivalent value is 1.

brightness=rpiblbrightness
It's the brightness plugin name. By default, nixie detects automatically the plugin from the system.
- Windows: winbrightness.
- Linux: xrandrbrightness.
- Rapberry: rpiblbrightness in case of official touch screen otherwise the plugin xrandrbrightness is used.

sensor=rpigpiobrightness
It's the automatic brightness sensor plugin name (valid only on Raspberry).

[opengl]
samples=8
It's the antialiasing sampler count if the system allows it

depthBufferSize=16
It's the depth buffer size if the system allows it.

title=true
- false the window has no title.
- true the window has a title.

title=false is selected by default for Raspberry with official touch screen.

size=default
- default: True for the default window size.

- max: Full screen.

size=max is selected by default for Raspberry with official touch screen.
The window is saved at the end of cession and restored at restart.

[positions]

3=(-1.4 0 0),(-0.14 0 0),(1.4 0 0)**
It's the tubes position for 3 tubes (depend of stl files).

4=(-2.0 0 0),(-0.8 0 0),(0.8 0 0),(2.0 0 0)**
It's the tubes position for 4 tubes (depend of stl files).

6=(-3.2 0 0),(-2.0 0 0),(-0.6 0 0),(0.6 0 0),(2.0 0 0),(3.2 0 0)**
It's the tubes position for 6 tubes (depend of stl files).

motionIncs=0.002,0.002,0.002
It's the motion increments in [0,1] interval in percent of the scene length.

motionMin=-0.1,-0.1,-0.5
It's the motion min position.

motionMax=0.1,0.1,0.3
It's the motion max position.

motionInterval=0
It's the interval time in ms (0 as fast as possible).

motion=false*
- false the motion is disable.
- True the motion is enable.

geometry=*
Geometry of the main window.

[clock]

dateFormat=0*
- 0 long date format.
- 1 short date format.

date=true*
- False date is hidden.
- True date is visible.

h24=true*
1. false 12 hours format.
2. true 24 hours format.

[alarms]

repeat=3
It's the maximum of repeats for an alarm.

interval=10
It's the interval of repeat in minutes.

ringtone=exe/ringtones/1.mp3
It's the ringtone file or url for the alarms.

volumeMin=20
It's the min volume of the alarm when it starts.

volumeMax=50
It's the max volume of the alarm.

volumeStep=5
The volume increase each 5 secondes.

[timers]

ringtone=exe/ringtones/2.mp3
It's the ringtone file or url for the times.

[thermometer]

degree=C*

- C for Celcius.
- F for Fahrenheit.

sensorModel=ds18b20
It's temperature sensor plugin name.

sensorId=
It's the temperature sensor identifier. For DS18B20, the program tries to detect it automatically.

[weather]

provider=yahooweather
It's the weather plugin name.

iploc=ip_api_com
It's the localization plugin name.

ipext=api_ipify_org
It's the public IP address plugin.

interval=15
It's the interval to update the weather in minutes.

town=*
It's the town name. You can enter the town name but you must verify that the weather provider know this name.

country=*

It's the country code. You can used ISO code or zip code (depend of plugin).
You can enter the ISO or zip code but you must verify that the weather provider know this code.

active=true*
- false the weather is disable.
- true the weather is enable.

[webradios]
provider=shoutcast
It's the web radio plugin name.

index=0*
It's the default web radio.

volume=20*
It's the start volume.

player=cvlc --quiet --extraintf=rc --repeat|add |quit
It's the player command line (valid only with Raspberry). You can use also:
player= mpg123 --quiet --remote --control|load |quit|@P 0|silence.
mpg123 is a very light player for mp3 format only. Must be used in conjunction with *mp3Only=true*.
The value contains at least 3 parts separate by '|':
- The command line to launch the player in background (mandatory).
- The command to load and start the music (mandatory).
- The command to quit the player (mandatory).
- A code emitted at the end of track (mandatory if the player has no repeat mode).
- You can add, always separate by '|' some strings needed for initialization (optional).

mixer=amixer
It's the volume mixer (valid only with Raspberry).

mp3Only=false
- false all radios are authorized.
- true only mp3 radios are authorized.

1=-xTream-|http://i.radionomy.com/documents/radio/f6023542-617e-4c98-af2f-45f962e8f67f.s165.jpg|http://listen.radionomy.com:80/-xTream-

2=.113FM Classic One||http://104.200.142.184:80/1864_128

3=Radio Classique||http://radioclassique.ice.infomaniak.ch/radioclassique-high.mp3

4=...
These entries are optional. They will be created if you choose web radios.
You can add manually some entries. The format is: [number]=title|logo url|radio url.
If you don't know the logo url, the entry is: [number]=title||radio url (e.g. entry number 3).

[brightness]
gpiopin=0
It's the gpio pin number for the brightness sensor.

active=true

- false disable the automatic ambient brightness.
- true enable the automatic ambient brightness.

hight=1.0

It's the highest brightness.

low=0.078

It's the lowest brightness.