

ElectroDacus monitoring and data logging.



user manual v0.2

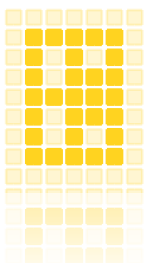


Table of contents

<i>Wyse 3040 Specifications</i>	<i>3</i>
<i>Quik start guide</i>	<i>4</i>
<i>1. ElectroDacus app</i>	<i>5</i>
<i>2. txt2csv converter application</i>	<i>9</i>
<i>3. HTML graph viewer</i>	<i>10</i>
<i>4. Power consumption</i>	<i>11</i>
<i>5. ProteuX Linux</i>	<i>12</i>
<i>6. Source code</i>	<i>13</i>
<i>7. DELL Wyse 3040 (what is included)</i>	<i>14</i>
<i>8. DELL Wyse 3040 (accessories you may want)</i>	<i>15</i>
<i>9. BIOS settings</i>	<i>17</i>

[illegible]

Step 1 – SBMS settings

Go to Device Settings / USART and set exactly as in below photo.

USART Data Log **1**

Baud rate **115.2K**

Log interval **1 second**.

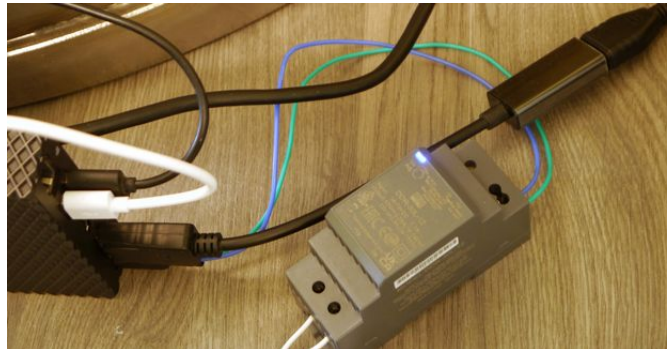


Step 2 – Setup the Wyse 3040

Connect the Wyse 3040 to a **12V** power supply capable of at least **1.5A**.

Connect the monitor (may need a DisplayPort to HDMI or whatever connection option your monitor has).

Connect a USB cable as long as you want between SBMS0 USB micro connector to Wyse 3040 USB type A.



Step 3 – Power up Wyse 3040

Push the power button on top of the Wyse 3040 and it will automatically boot in to PorteuX Linux in about 30 seconds.

Double click the ElectroDacus icon that you will see on top left side on the Desktop and the application will start.

Push the blue button on the right side of the application to make it full screen.



1 ElectroDacus application.

The app should be simple and intuitive to use and displays all the same information as the SBMS0 just in a different graphical format. A file containing the log data at 1 second interval will be saved as yymmdd_Data.txt in the same directory where the app executable file is located so you can make a copy of the app folder on say an USB flash memory and then data files will be saved there.



Close application.

Toggle full screen.

Toggle between current day energy and user resettable* energy counter.

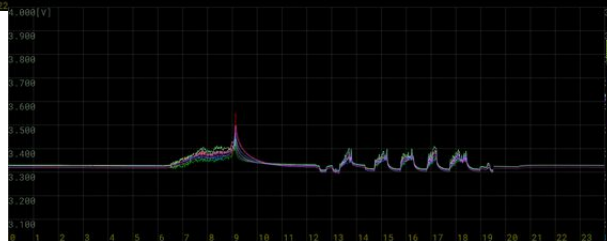
Toggle between Amp and Watt

Toggle between 3 types of signals.

* Keep the button pressed for 5 seconds to reset the energy counter .

1.1 ElectroDacus application signal types.

Showing the 3 types of signals. Energy, Cell voltages and miscellaneous (includes SOC, ADC2, ADC3, internal and external temperatures).



1.2 ElectroDacus application signal types and cursors.

Showing the 3 types of signals. Energy, Cell voltages and miscellaneous (includes SOC, ADC2, ADC3, internal and external temperatures). Also the two cursors yellow selecting the minute in the 24h graph and cyan selecting the second in the zoomed in (60x) portion-of the graph. While cursors are visible the data is log data not live and to exit this mode you just click anywhere outside the graph area.

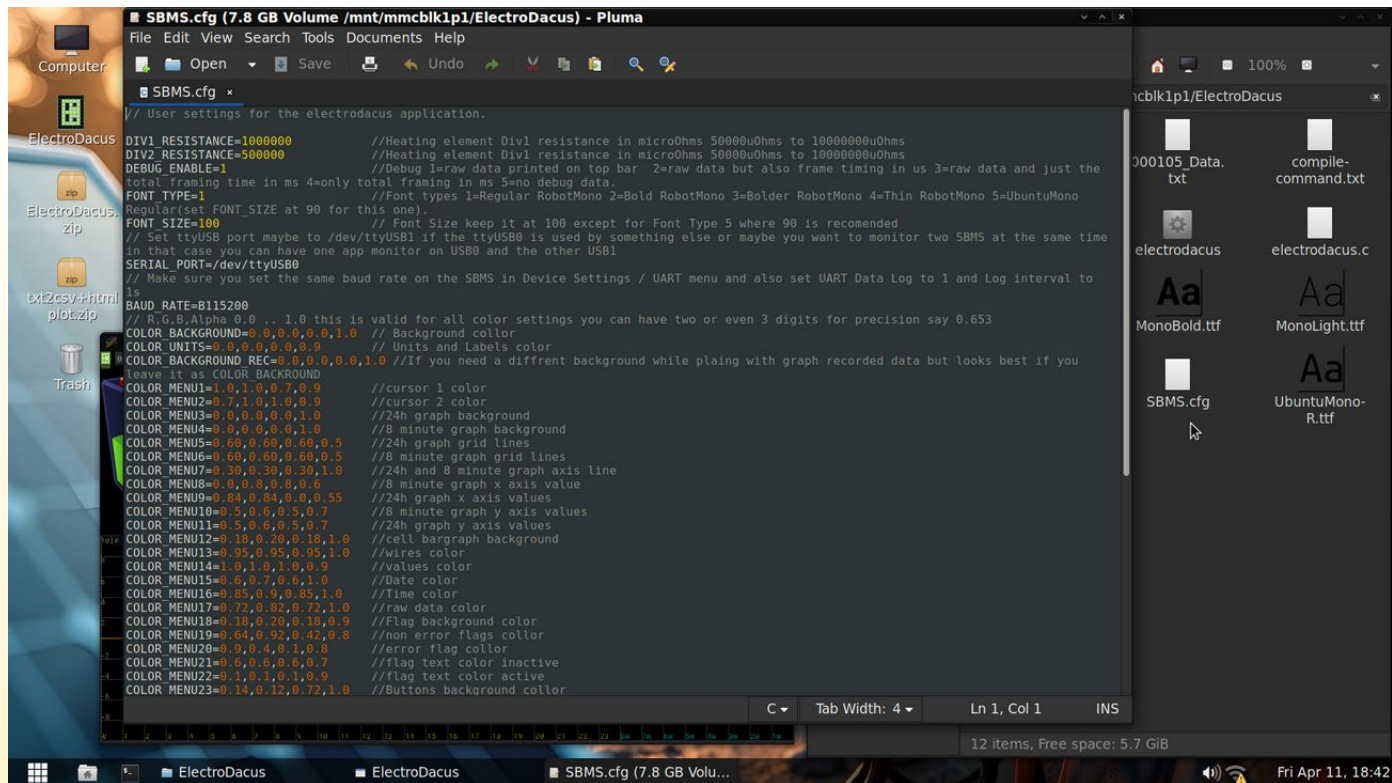


1.3 ElectroDacus configuration file (SBMS.cfg).

There are multiple things that you can set including the heating element's resistance for diversion if you are using that, serial port useful if you want to monitor more than one SBMS0 or if you have some other USB devices that have serial interface and SBMS0 was not connected first.

You can also change the baud rate but 115.2K is a great choice.

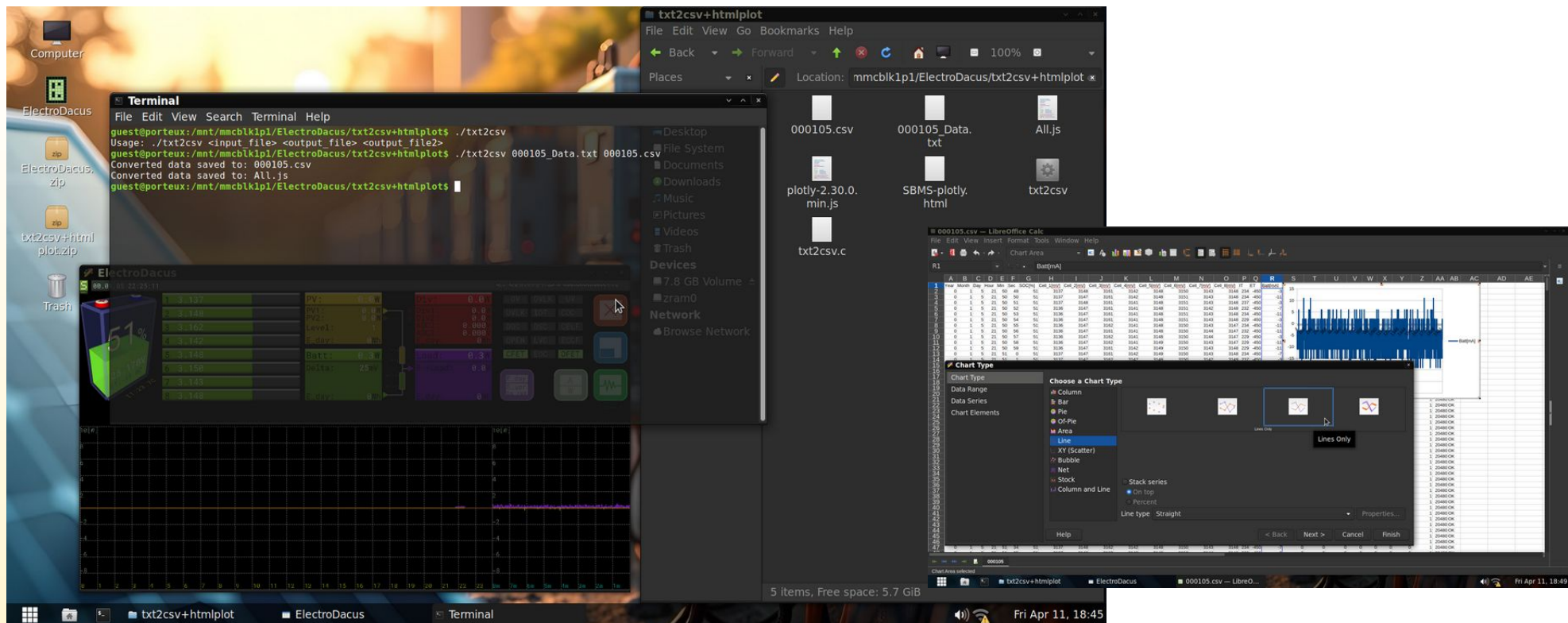
And maybe more exiting for some of you you can change all the colors of the app if you get bored or do not like my choices :)



2 ElectroDacus csv converter application.

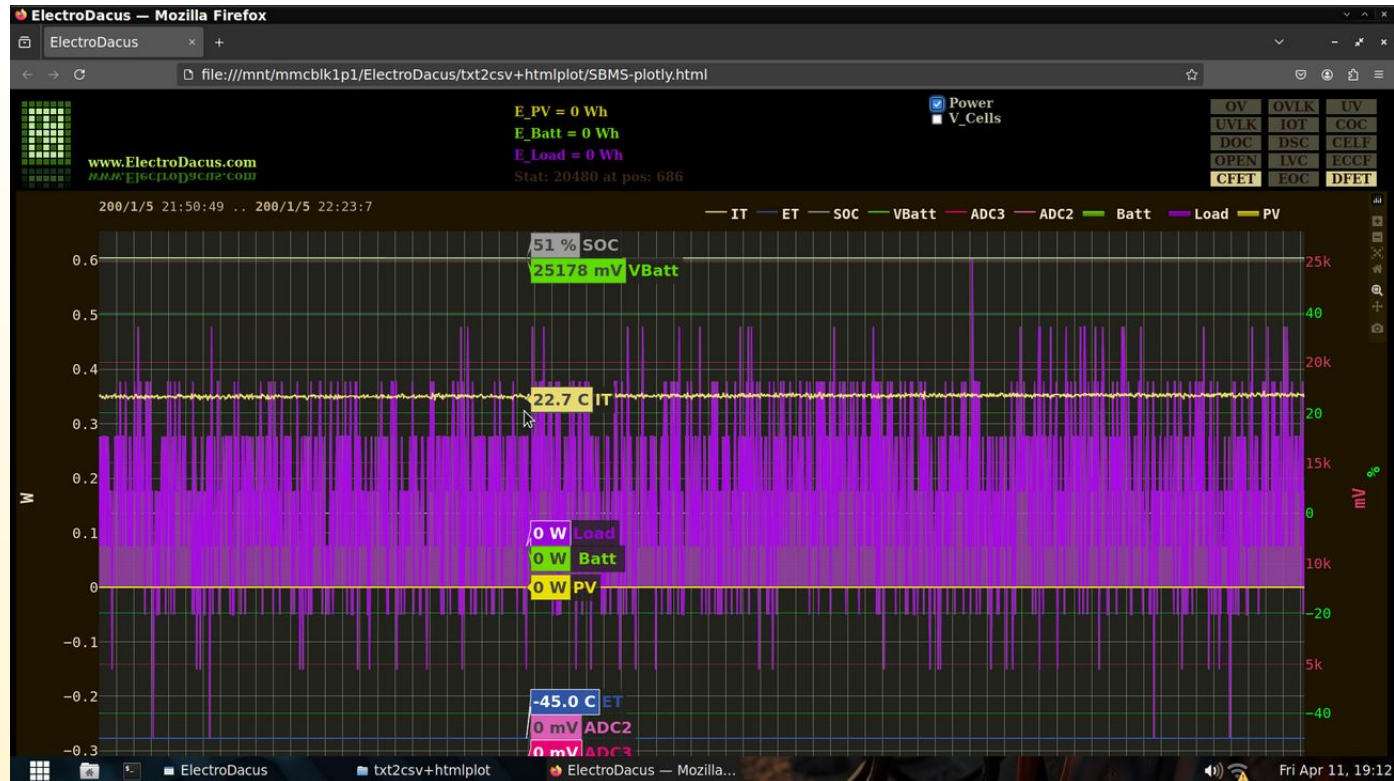
You will also find a command line utility called `txt2csv` that will convert this `yymmdd_Data.txt` files to `.csv` so that you can use them in your preferred spreadsheet app.

The use is fairly simple as you see below just open a terminal by right clicking in that directory and select `Open in Terminal` then write the following `“./txt2csv inputfile.txt outputfile.csv”` and it should be converted in a second or two.



3 ElectroDacus HTML graph viewer.

That same txt2csv converter will also create file called All.js that can be visualized in a web browser by opening the SBMS-plotly.html. Below it is how it looks opened on the Wyse 3040 but is best to copy this on your desktop or laptop computer for better performance. On your computer you will need 3 files the SBMS-plotly.html , plotly-2.30.0.min.js and All.js Any computer that has a web browser will work and all you do is open the SBMS-plotly.html in your browser.



4 Power Consumption

The setup was supplied from a 12V bench power supply in series with a 10Ohm resistor that was used to measure voltage drop (V_{drop}) across it on CH1 of the oscilloscope.

$$P = (12V - V_{drop}) * I$$

	Wyse 3040 [W]	14" USB powered LCD [W]	Total [W]
a) Idle no LCD	2.53	0.00	2.53
b) Idle LCD brightness 20%	2.53	3.30	5.83
c) Idle LCD brightness 100%	2.53	6.13	8.66
d) ElectroDacus app running no LCD	2.82	0.00	2.82
e) ElectroDacus app running LCD brightness 20%	2.82	3.27	6.09
f) 1080p video playback LCD brightness 20%	5.67	3.27	8.94

a) Idle no LCD



b) Idle LCD brightness 20%



c) Idle LCD brightness 100%



d) ElectroDacus app running no LCD



e) ElectroDacus app running LCD brightness 20%



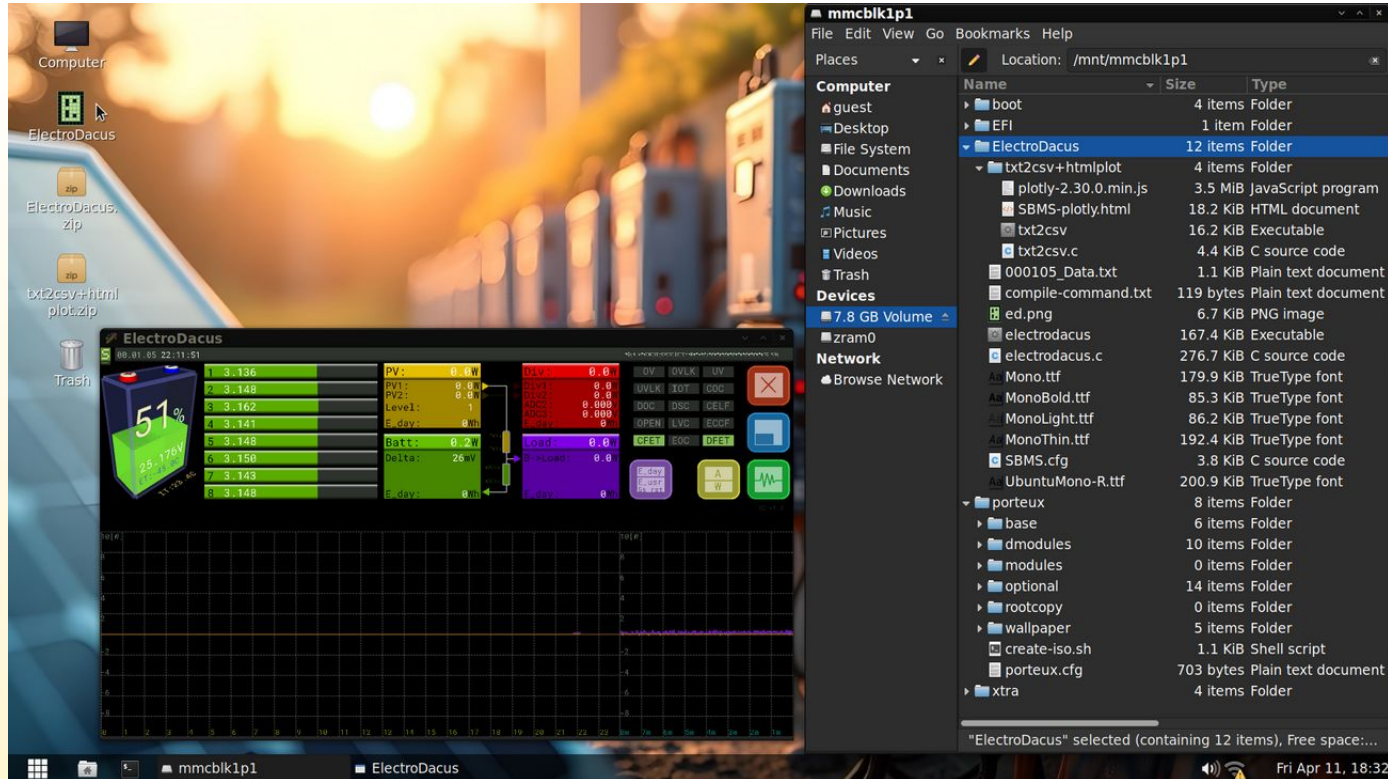
f) 1080p video playback LCD brightness 20%



5 PorteuX Linux OS using MATE desktop.

To open the ElectroDacus app you can just double click the logo that is next to the mouse cursor in below screenshot . That is a link to the actual application witch resides on the 8GB eMMC internal drive and you can see the content of that drive also in below screenshot.

Try not to delete boot, EFI or porteux folders as those are part of the operating system. If you mess up or delete the ElectroDacus folder you can just recreate that by decompressing the two zip files you see on the desktop .

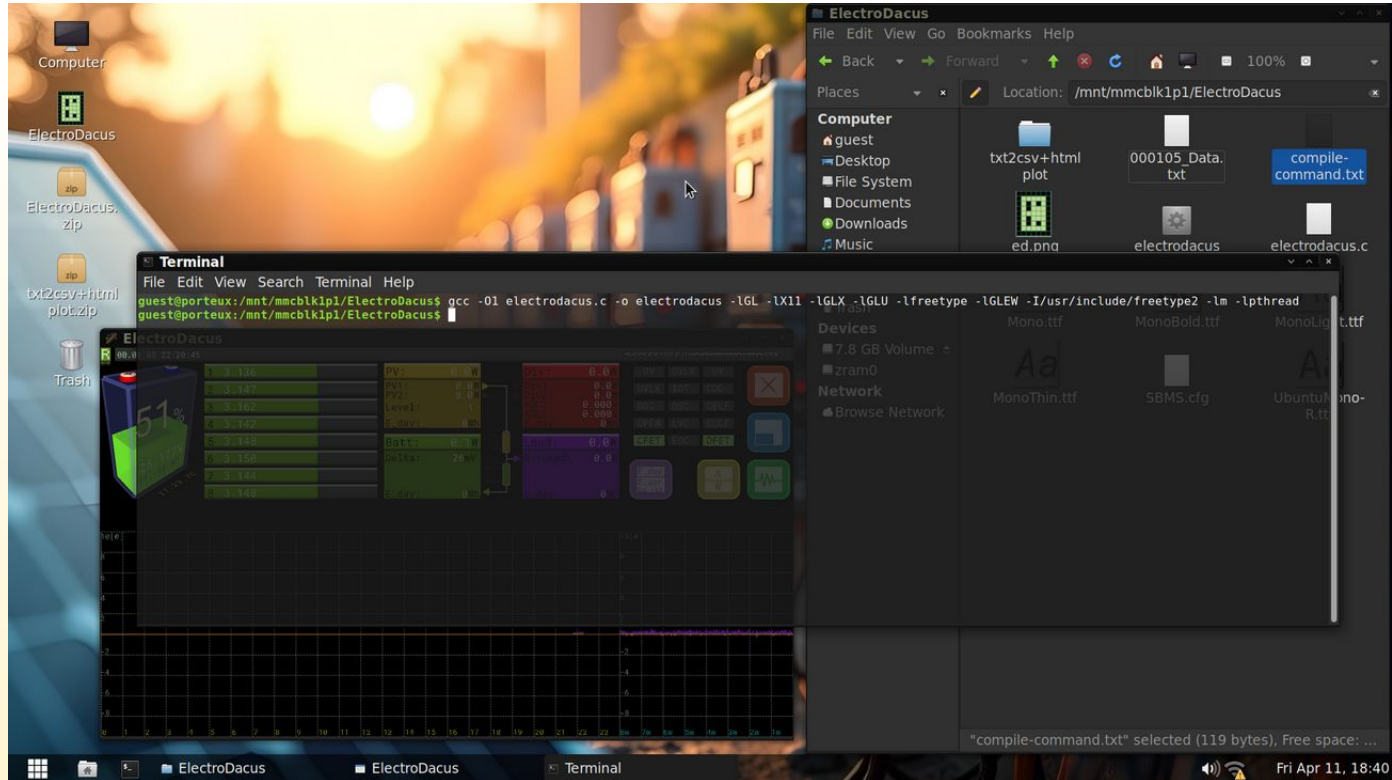


6 ElectroDacus source code.

The source code is included and free to use with no warranty of any type :).

You can modify the app or build your own based on it and you can even do so directly on the mini PC since the gcc compiler is already setup. There is a file called compile-command.txt from where you can copy the command and paste in terminal.

The txt2csv converter source code is also included and there you can compile simply as it is not using any special library.



7 DELL Wyse 3040 (what is included).

A used Dell Wyse 3040 with preinstalled PorteuX Linux and ElectroDacus app + a 1m long DC power cable with the matching 4x1.7mm connector are included. This batch of used Wyse 3040 are manufactured around 2020 and BIOS battery is no longer working (only 4 of the 10 tested still had a BIOS battery that was functional but I expect that will not last long).

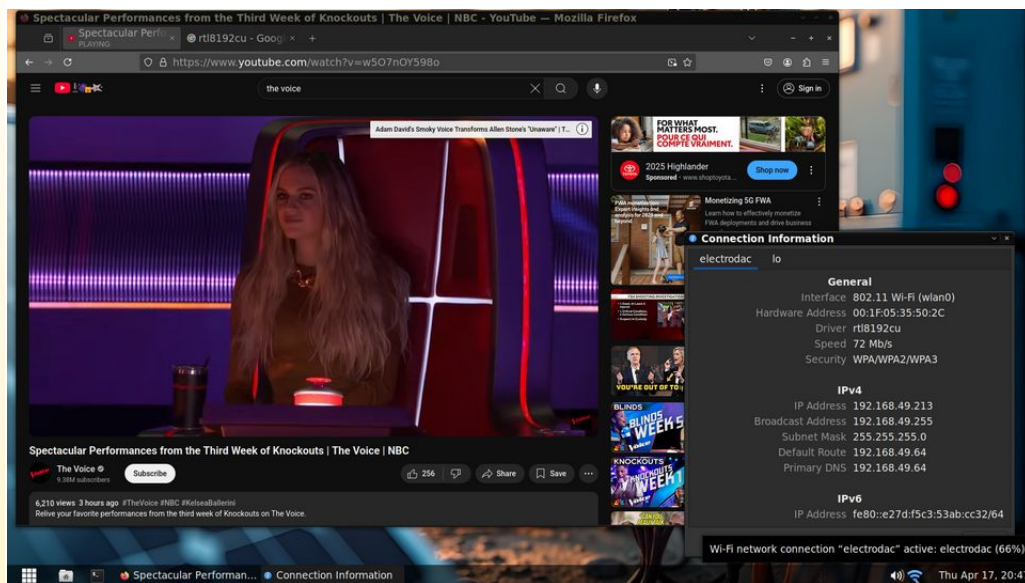
I think a battery is not needed for this type of applications where once you setup the Wyse and set the time and date it will always be powered by your large Lithium battery so if you need to remove power occasionally maybe once a year it takes just 2 minutes to setup the BIOS date and time. I remove the default BIOS password but in case I forget to do so it is “Fireport” (notice the capital F).



8 DELL Wyse 3040 (accessories you may want).

There is no build in WiFi for this models (I have just a few that have it built in contact me if you want one of those). I got one and just tested and it works great. This is the Amazon link <https://www.amazon.ca/dp/B01ERHE18S> and you can see it in the photos next to the USB cables (HomeSpot 150Mbps Wireless N WiFi USB Nano Adapter) it was just CA\$ 10.99 (about US\$ 8). You can also see a screenshot with details (it uses RTL8192cu driver and signal is at 66 to 70% about 5m from the phone that is another room same signal strength as my Asus P1801 all in one computer next to it). Measured power consumption is 350mW with peaks around 400mW so decently low.

USB extension is 25ft also from Amazon <https://www.amazon.ca/dp/B0893R82RK> It is the longest USB 2.0 passive cable I found as from 30ft and up most are active and of course those will work as they do signal amplification. To this 25ft I added another lower quality 6ft extension I had and a 3ft USB A to micro USB to connect to SBMS0 and it works just great. I think it will even work with 2x 25ft extensions as SBMS0 requires just 9mA to power the digital isolator and communication is at low speed.



... (accessories you may want).

There monitor I purchased to connect to this is a small 14" with touchscreen (touchscreen is not needed but fairly useful). The monitor is also from Amazon and was CA\$ 109.99 (about US\$ 80) and here is the link <https://www.amazon.ca/dp/B0CP3NTK2T> but if the link no longer works just search for portable monitors as there are plenty. Just make sure it has an HDMI and you also get the DisplayPort to HDMI adapter. I can not find the exact one I got but all adapters are the same since is just connectors and cable and it looks like this one <https://www.amazon.ca/dp/B017Q8ZVWK> and works great.

A 12V power supply is likely needed even if you have a 12V battery tho I tested the mini PC and seems to be fine with lower voltage tested down to 6V but tested for days at 9V and it is perfectly fine (likely it has a DC-DC converter internally from 12V to 5V).

You will need at least 1.5A at 12V 2A if you chose a 9V supply. I chose a fairly expensive isolated DIN mountable power supply but you do not need to do that as SBMS0 USB is isolated so you do not need an isolated power supply.

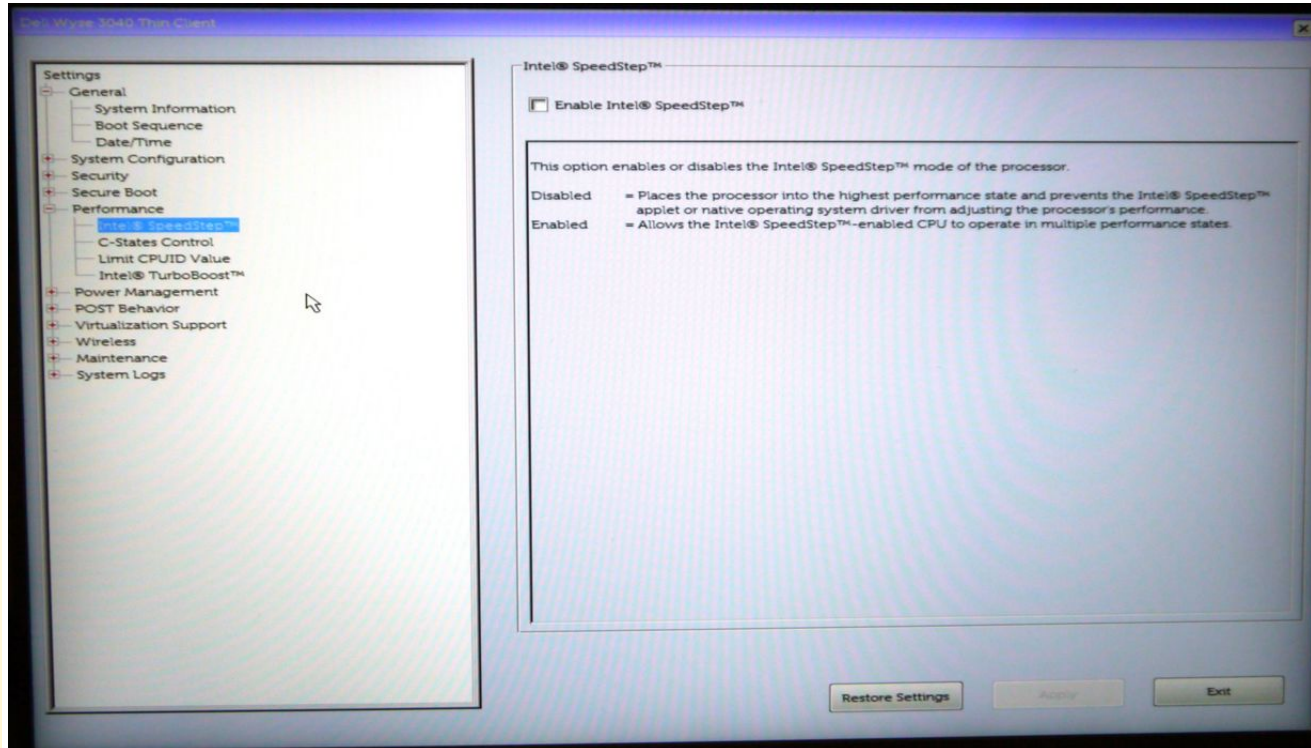
The power supply I used is Mean Well DDR-30L-12 but that is good just for 24V battery the DDR-30G-12 is good for both 12V and 24V batteries.



9 BIOS settings.

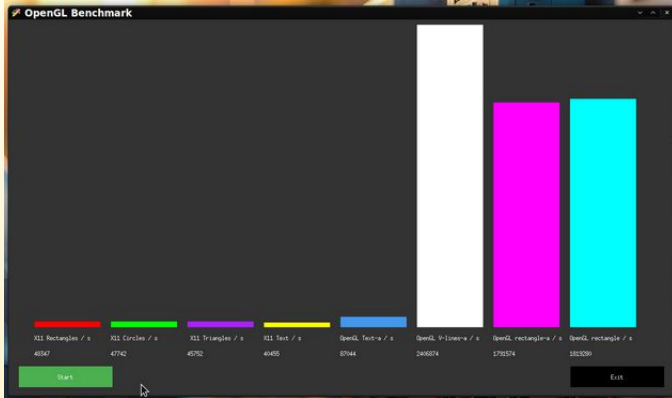
To increase performance I found that best setting is to disable everything in Performance so SpeedStep, C-States and TurboBoost. This way the iGPU will work at full speed no longer entering power save mode and same for CPU and this significantly improves the performance of almost any app including the ElectroDacus app but about a factor of 2x

The issue seems to be that with all those power saving settings enabled processor will allays go down to 480Mhz and it takes time to change the speed making many application look more sluggish than they should be. The iGPU also stays at full speed and both in OpenGL applications and video decoding (YouTube or video playback) is better.



The screenshot shows the OpenGL Benchmark application window. The title bar reads "OpenGL Benchmark". The main content area displays a bar chart with eight bars of different colors (red, green, purple, yellow, blue, white, magenta, cyan) representing different performance metrics. Below each bar is a label and a numerical value. The metrics are: R11 Rectangles / s (58350), R12 Circles / s (37529), R13 Triangles / s (38495), R11 Text / s (42982), Speed, Text-m / s (48186), Speed, Triangles-m / s (829610), Speed, rectangles-m / s (818463), and Speed, rectangle / s (863830). At the bottom left is a green "Start" button, and at the bottom right is a black "Exit" button.

Metric	Value
R11 Rectangles / s	58350
R12 Circles / s	37529
R13 Triangles / s	38495
R11 Text / s	42982
Speed, Text-m / s	48186
Speed, Triangles-m / s	829610
Speed, rectangles-m / s	818463
Speed, rectangle / s	863830



The screenshot shows a virtual machine desktop environment. In the foreground, a terminal window is open, displaying the output of several commands. The background features a large window titled 'OpenGL Benchmark' which shows a dense, colorful, pixelated pattern. The desktop also includes icons for 'Computer', 'ElectroDacus', and 'ElectroDacus.zip'. The taskbar at the bottom shows the 'Terminal' and 'ElectroDacus' windows.

Terminal Output:

```

guest@porteux:~$ cat /proc/cpuinfo | grep Mhz
cpu MHz: 480.000
cpu MHz: 479.994
cpu MHz: 480.000

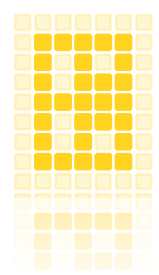
guest@porteux:~$ cat /proc/cpuinfo | grep Mhz
cpu MHz: 480.000
cpu MHz: 480.000
cpu MHz: 480.005
cpu MHz: 480.000

guest@porteux:~$ cat /proc/cpuinfo | grep Mhz
cpu MHz: 480.000
cpu MHz: 480.000
cpu MHz: 480.000
cpu MHz: 480.000

guest@porteux:~$ cat /proc/cpuinfo | grep Mhz
cpu MHz: 480.000
cpu MHz: 480.000
cpu MHz: 480.000
cpu MHz: 480.000

guest@porteux:~$ cat /proc/cpuinfo | grep Mhz
cpu MHz: 508.042
cpu MHz: 482.833
cpu MHz: 480.000
cpu MHz: 480.000

guest@porteux:~$
  
```



SW files both source code and executable are available on the Wyse 3040 and are free to use as you wish.