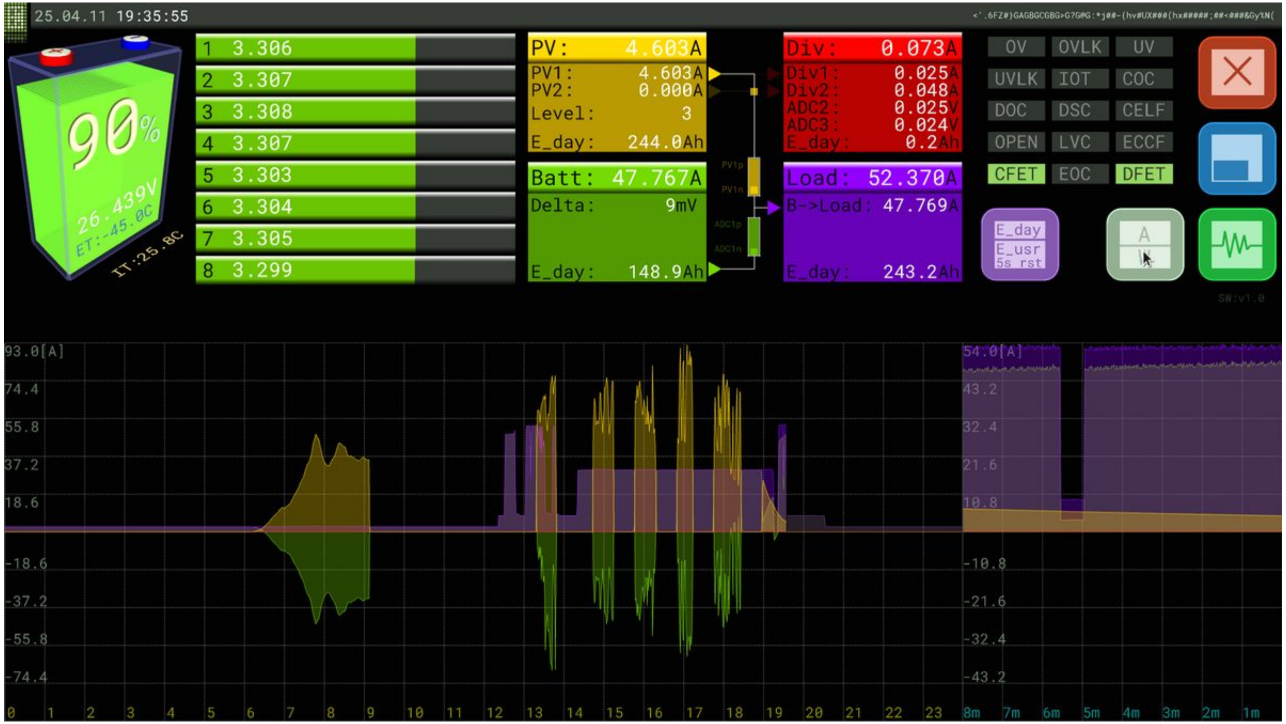
# ElectroDacus monitoring and data logging.


25.04.11 19:35:55

90%
26.439V
ET:-45.0C
IT:25.8C

1 3.306
2 3.307
3 3.308
4 3.307
5 3.303
6 3.304
7 3.305
8 3.299

PV: 4.603A
PV1: 4.603A
PV2: 0.000A
Level: 3
E_day: 244.0Ah

Batt: 47.767A
Delta: 9mV
E_day: 148.9Ah

Div: 0.073A
Div1: 0.025A
Div2: 0.048A
ADC2: 0.025V
ADC3: 0.024V
E_day: 0.2Ah

Load: 52.370A
B->Load: 47.769A
E_day: 243.2Ah

***user manual*** *v0.2*

# *Table of contents*

# Specification

| | |
|---|---|
| Model | DELL Wyse 3040 ( used ) |
| CPU | Intel Atom x5-Z8350 (Quad Core 1.44 GHz) |
| MEMORY | 2GB DDR3L 1600MHz, solder down |
| STORAGE | 8GB eMMC flash chip |
| USB | 1 x USB 3.1 Gen 1    3 x USB 2.0 |
| AUDIO | 1 x audio-mic combo jack |
| DISPLAY  OUTPUT | 2 X DisplayPort (max 2560x1600 @ 60Hz) |
| NETWORKING | 1Gb/s RJ45 (Wired Ethernet) |
| POWER INPUT | 12Vdc 1.5A (not included) |
| POWER CONSUMPTION | 2.6W (idle see page 5 for details) |
| DIMENSIONS | 101.6 mm x 101.6mm x 27.94mm |
| WEIGHT | 0.24kg  (0.53lb) |
| OS Installed | Customized version of PorteuX 2.0 |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |
| | |

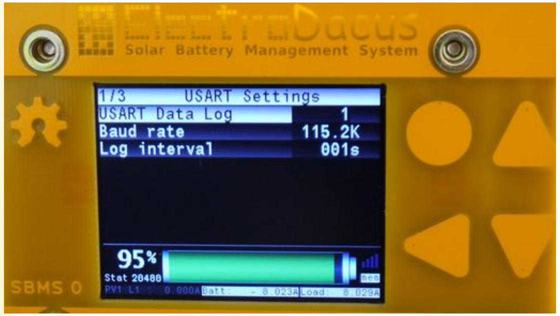
## Step 1 – SBMS settings

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

*USART Data Log* **1**

*Baud rate* **115.2K**

*Log interval* **1** *second .*



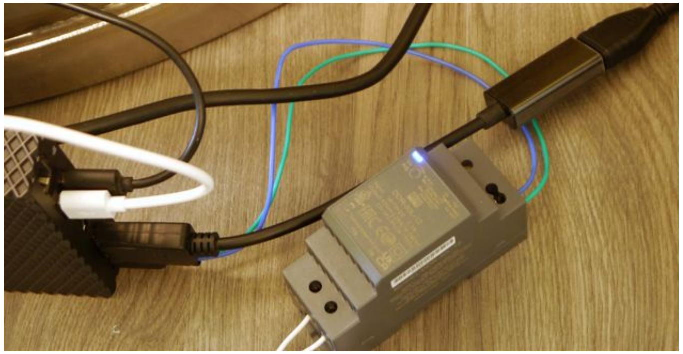
```
1/3          USART Settings
USART Data Log                1
Baud rate                115.2K
Log interval               001s
```

## 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

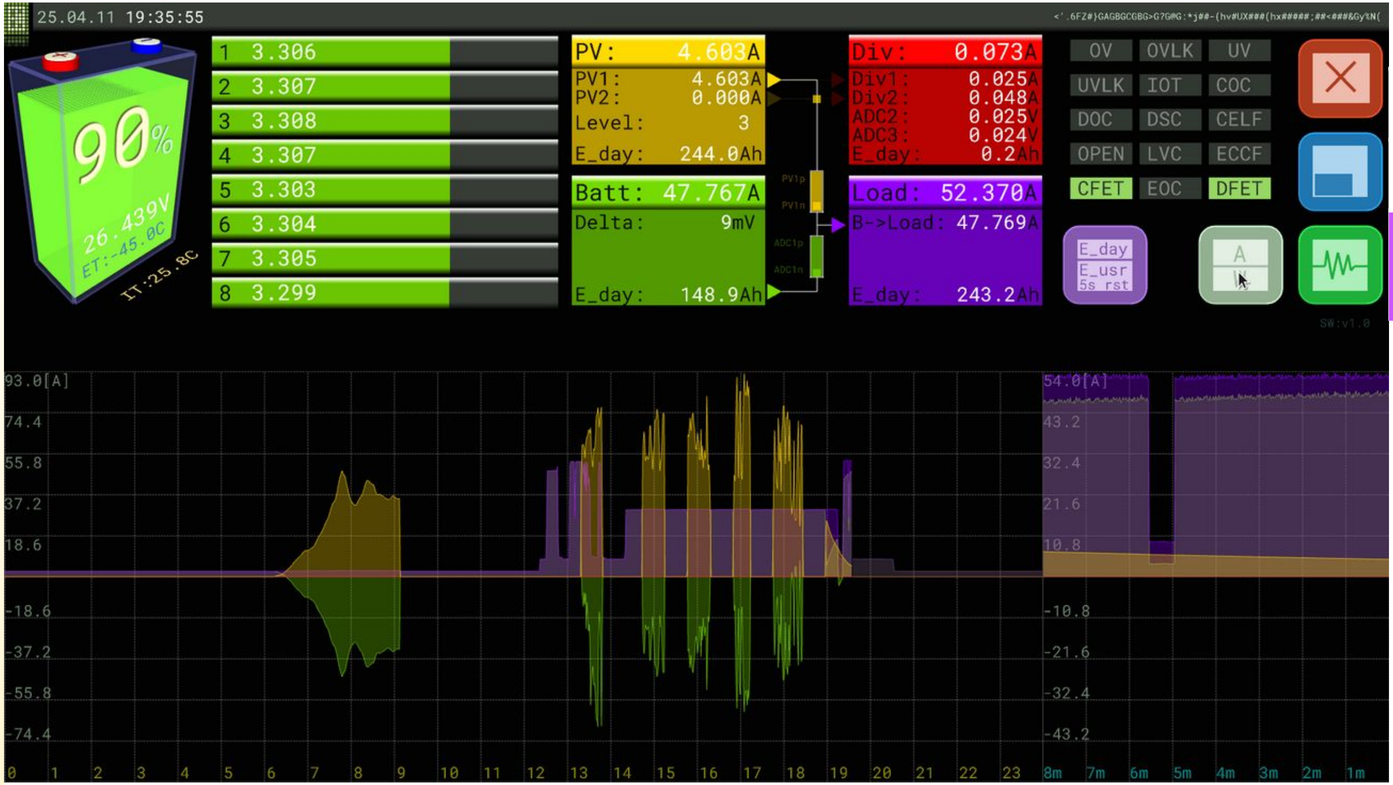
*Push the power button on top of the Wyse 3040 and it will automatically boot in to PorteuX Linux in about 30 seconds .*

*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.*



25.04.11 19:35:55

```
<'.6FZ#}GAGBGCGBG>G?G#G:*j##-(hv#UX###(hx######;##<###&Gy%N(
```

90%
26.439V
ET:-45.0C
IT:25.8C

1 3.306
2 3.307
3 3.308
4 3.307
5 3.303
6 3.304
7 3.305
8 3.299

PV: 4.603A
PV1: 4.603A
PV2: 0.000A
Level: 3
E_day: 244.0Ah

Batt: 47.767A
Delta: 9mV
E_day: 148.9Ah

Div: 0.073A
Div1: 0.025A
Div2: 0.048A
ADC2: 0.025V
ADC3: 0.024V
E_day: 0.2Ah

Load: 52.370A
B->Load: 47.769A
E_day: 243.2Ah

Close application.

Toggle full screen.

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

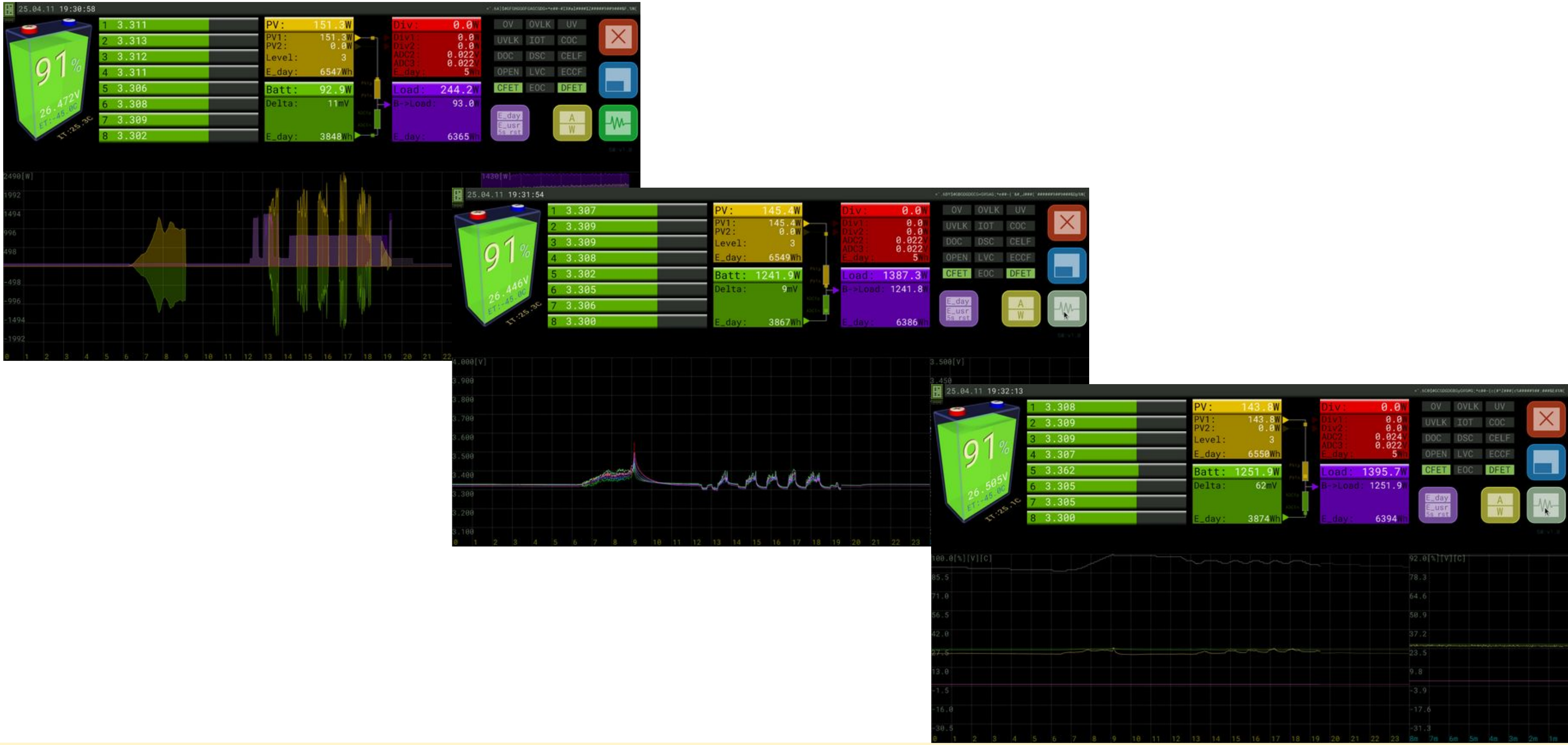
Toggle between Amp and Watt

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).*



25.04.11 19:30:58

91% 26.472V ET:-45.0C IT:25.3C

1 3.311
2 3.313
3 3.312
4 3.311
5 3.306
6 3.308
7 3.309
8 3.302

PV: 151.3W
PV1: 151.3W
PV2: 0.0W
Level: 3
E_day: 6547Wh

Batt: 92.9W
Delta: 11mV
E_day: 3848Wh

Div: 0.0W
Div1: 0.0
Div2: 0.0
ADC2: 0.022
ADC3: 0.022
E_day: 5Wh

Load: 244.2W
B->Load: 93.0
E_day: 6365Wh

25.04.11 19:31:54

91% 26.446V ET:-45.0C IT:25.3C

1 3.307
2 3.309
3 3.309
4 3.308
5 3.302
6 3.305
7 3.306
8 3.300

PV: 145.4W
PV1: 145.4W
PV2: 0.0W
Level: 3
E_day: 6549Wh

Batt: 1241.9W
Delta: 9mV
E_day: 3867Wh

Div: 0.0W
Div1: 0.0
Div2: 0.0
ADC2: 0.022
ADC3: 0.022
E_day: 5

Load: 1387.3W
B->Load: 1241.8
E_day: 6386

25.04.11 19:32:13

91% 26.505V ET:-45.0C IT:25.1C

1 3.308
2 3.309
3 3.309
4 3.307
5 3.362
6 3.305
7 3.305
8 3.300

PV: 143.8W
PV1: 143.8W
PV2: 0.0W
Level: 3
E_day: 6550Wh

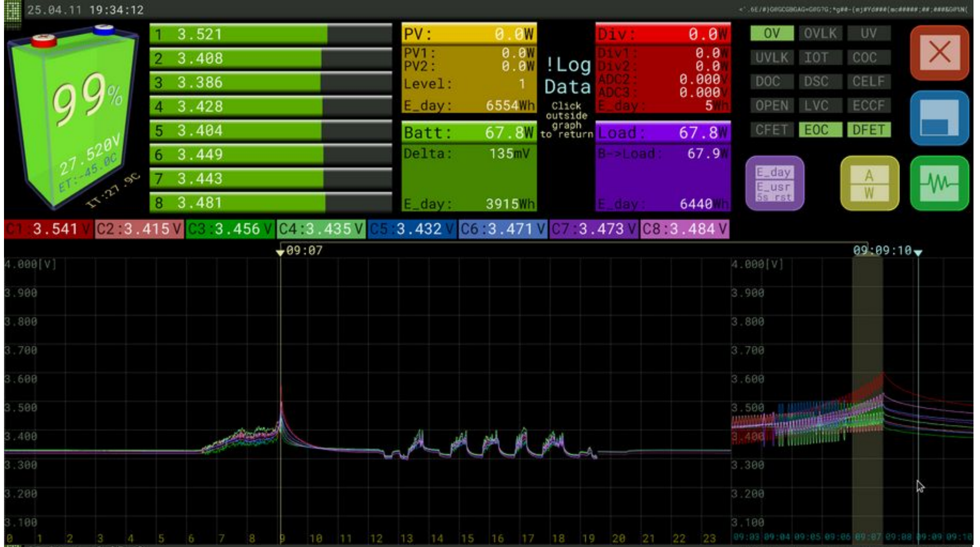
Batt: 1251.9W
Delta: 62mV
E_day: 3874Wh

Div: 0.0W
Div1: 0.0
Div2: 0.0
ADC2: 0.024
ADC3: 0.022
E_day: 5

Load: 1395.7W
B->Load: 1251.9
E_day: 6394

# *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.*



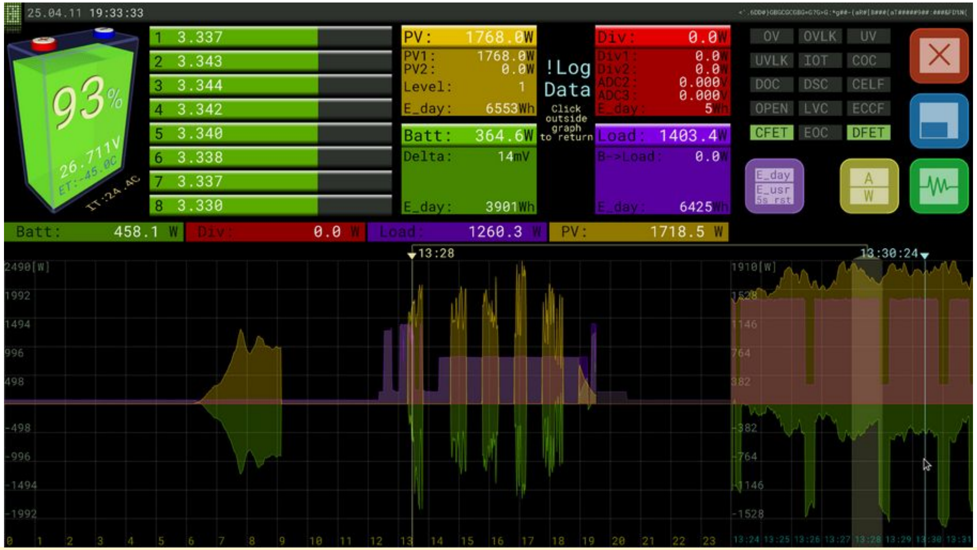
25.04.11 19:34:12

99% 27.520V ET:-45.0C IT:27.9C

1 3.521
2 3.408
3 3.386
4 3.428
5 3.404
6 3.449
7 3.443
8 3.481

PV: 0.0W
PV1: 0.0W
PV2: 0.0W
Level: 1
E_day: 6554Wh
Batt: 67.8W
Delta: 135mV
E_day: 3915Wh
!Log Data Click outside graph to return
Div: 0.0W
Div1: 0.0
Div2: 0.0
ADC2: 0.000
ADC3: 0.000
E_day: 5Wh
Load: 67.8W
B->Load: 67.9
E_day: 6440Wh

C1 3.541 V C2:3.415 V C3 3.456 V C4:3.435 V C5 3.432 V C6:3.471 V C7:3.473 V C8:3.484 V



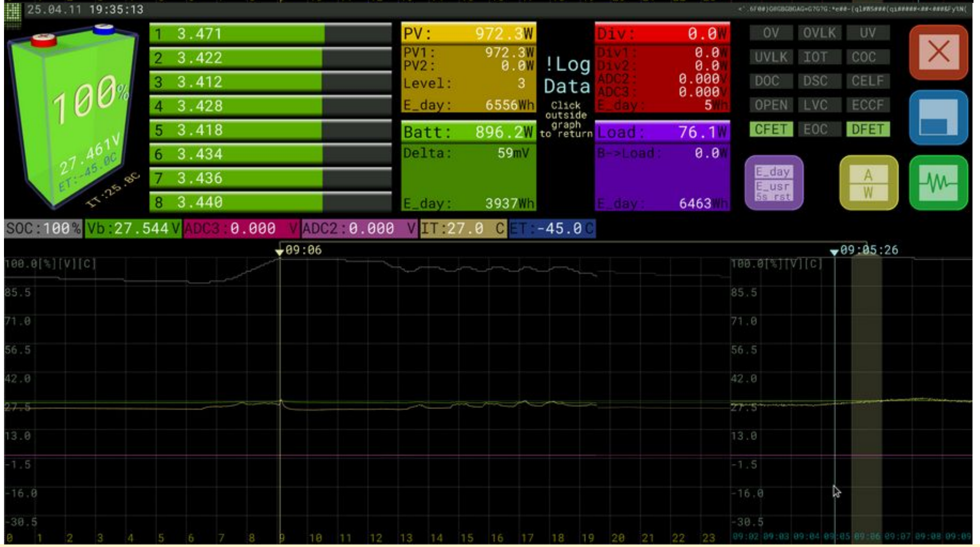
25.04.11 19:33:33

93% 26.711V ET:-45.0C IT:24.4C

1 3.337
2 3.343
3 3.344
4 3.342
5 3.340
6 3.338
7 3.337
8 3.330

PV: 1768.0W
PV1: 1768.0W
PV2: 0.0W
Level: 1
E_day: 6553Wh
Batt: 364.6W
Delta: 14mV
E_day: 3901Wh
!Log Data Click outside graph to return
Div: 0.0W
Div1: 0.0W
Div2: 0.0W
ADC2: 0.000V
ADC3: 0.000V
E_day: 5Wh
Load: 1403.4W
B->Load: 0.0W
E_day: 6425Wh

Batt: 458.1 W Div: 0.0 W Load: 1260.3 W PV: 1718.5 W



25.04.11 19:35:13

100% 27.461V ET:-45.0C IT:25.8C

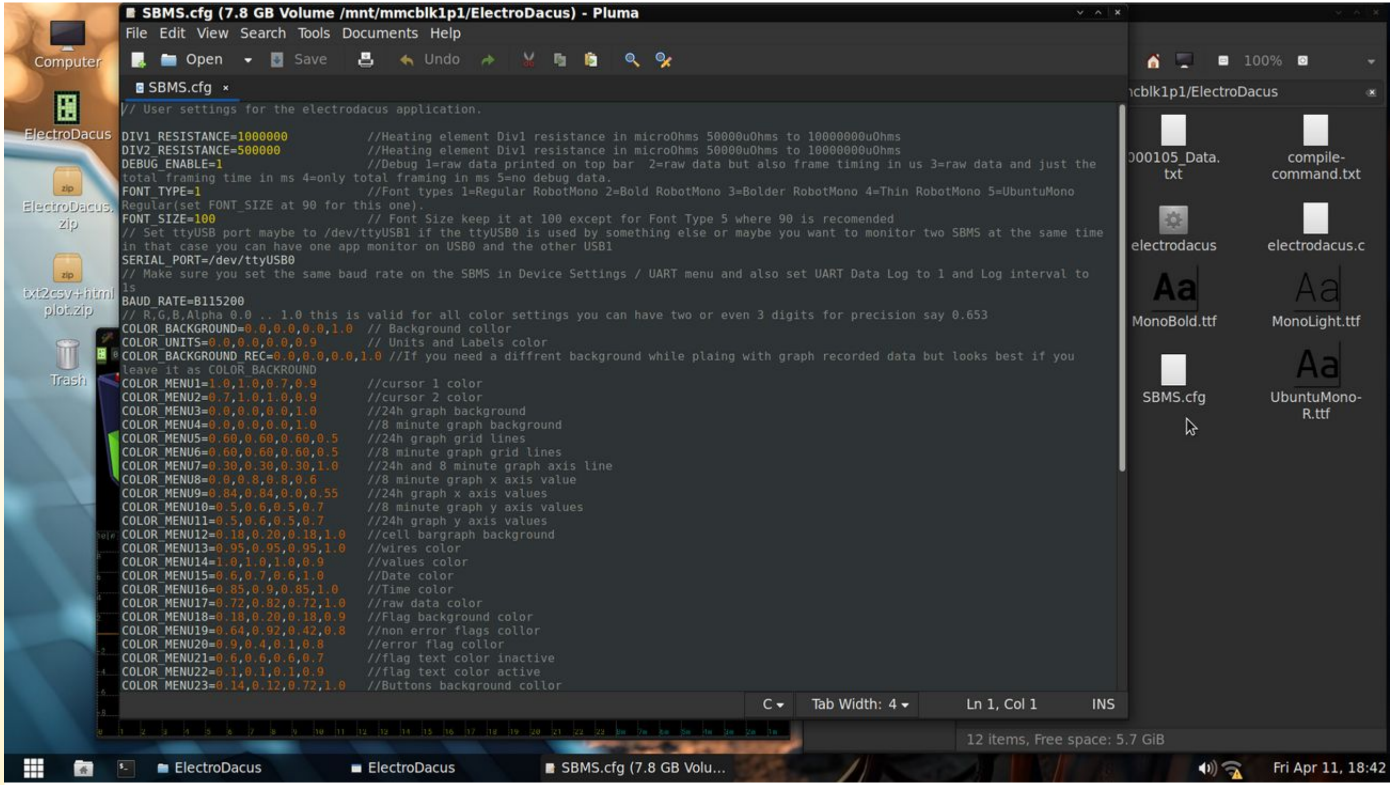
1 3.471
2 3.422
3 3.412
4 3.428
5 3.418
6 3.434
7 3.436
8 3.440

PV: 972.3W
PV1: 972.3W
PV2: 0.0W
Level: 3
E_day: 6556Wh
Batt: 896.2W
Delta: 59mV
E_day: 3937Wh
!Log Data Click outside graph to return
Div: 0.0W
Div1: 0.0
Div2: 0.0
ADC2: 0.000
ADC3: 0.000
E_day: 5Wh
Load: 76.1W
B->Load: 0.0
E_day: 6463Wh

SOC:100% Vb:27.544 V ADC3:0.000 V ADC2:0.000 V IT:27.0 C ET:-45.0 C

## *1.3 ElectroDacus configuration file (SBMS.cfg).*

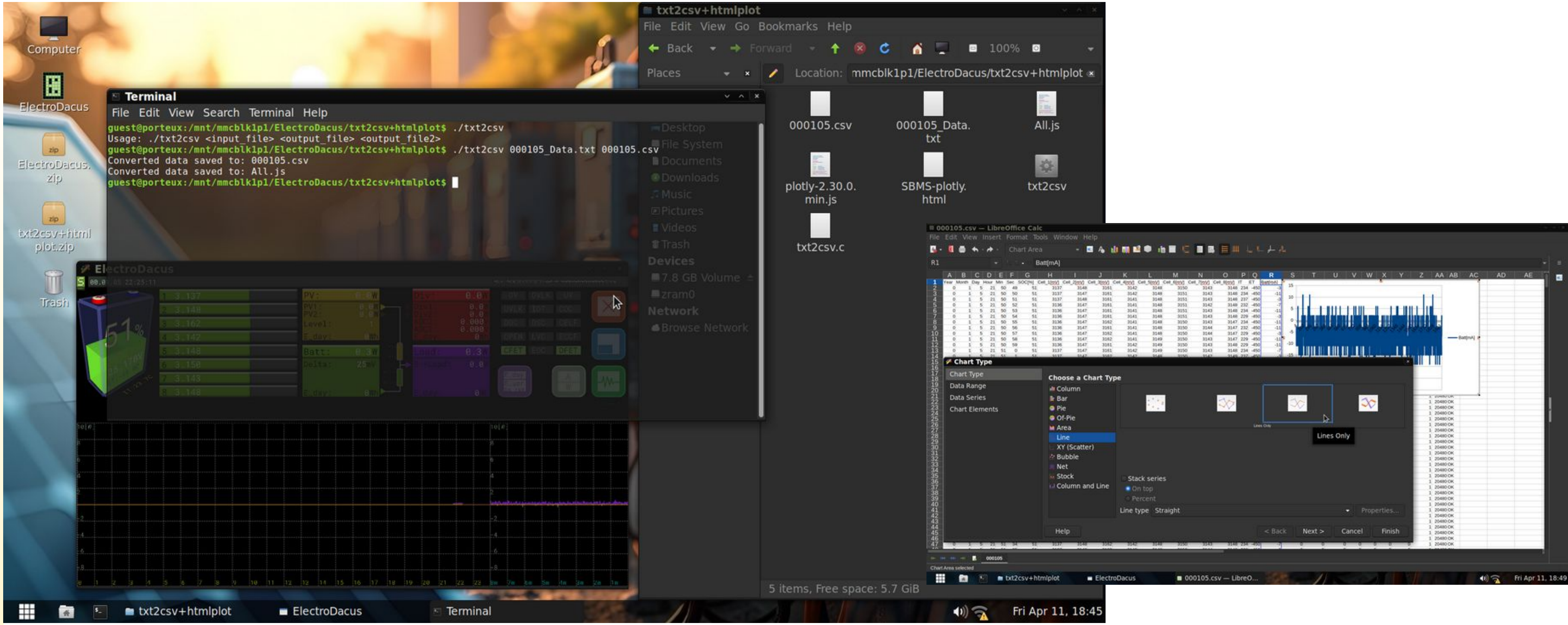
*There are multiple things that you can set including the heating element’s resistance for diversion if you are using that, serial port useful if you want to monitor more than one SBMS0 or if you have some other USB devices that have serial interface and SBMS0 was not connected first.*
*You can also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



```
// User settings for the electrodacus application.

DIV1_RESISTANCE=1000000           //Heating element Div1 resistance in microOhms 50000uOhms to 10000000uOhms
DIV2_RESISTANCE=500000            //Heating element Div1 resistance in microOhms 50000uOhms to 10000000uOhms
DEBUG_ENABLE=1                    //Debug 1=raw data printed on top bar  2=raw data but also frame timing in us 3=raw data and just the
total framing time in ms 4=only total framing in ms 5=no debug data.
FONT_TYPE=1                       //Font types 1=Regular RobotMono 2=Bold RobotMono 3=Bolder RobotMono 4=Thin RobotMono 5=UbuntuMono
Regular(set FONT_SIZE at 90 for this one).
FONT_SIZE=100                     // Font Size keep it at 100 except for Font Type 5 where 90 is recomended
// Set ttyUSB port maybe to /dev/ttyUSB1 if the ttyUSB0 is used by something else or maybe you want to monitor two SBMS at the same time
in that case you can have one app monitor on USB0 and the other USB1
SERIAL_PORT=/dev/ttyUSB0
// Make sure you set the same baud rate on the SBMS in Device Settings / UART menu and also set UART Data Log to 1 and Log interval to
1s
BAUD_RATE=B115200
// R,G,B,Alpha 0.0 .. 1.0 this is valid for all color settings you can have two or even 3 digits for precision say 0.653
COLOR_BACKGROUND=0.0,0.0,0.0,1.0  // Background collor
COLOR_UNITS=0.0,0.0,0.0,0.9       // Units and Labels color
COLOR_BACKGROUND_REC=0.0,0.0,0.0,1.0 //If you need a diffrent background while plaing with graph recorded data but looks best if you
leave it as COLOR_BACKROUND
COLOR_MENU1=1.0,1.0,0.7,0.9       //cursor 1 color
COLOR_MENU2=0.7,1.0,1.0,0.9       //cursor 2 color
COLOR_MENU3=0.0,0.0,0.0,1.0       //24h graph background
COLOR_MENU4=0.0,0.0,0.0,1.0       //8 minute graph background
COLOR_MENU5=0.60,0.60,0.60,0.5    //24h graph grid lines
COLOR_MENU6=0.60,0.60,0.60,0.5    //8 minute graph grid lines
COLOR_MENU7=0.30,0.30,0.30,1.0    //24h and 8 minute graph axis line
COLOR_MENU8=0.0,0.8,0.8,0.6       //8 minute graph x axis value
COLOR_MENU9=0.84,0.84,0.0,0.55    //24h graph x axis values
COLOR_MENU10=0.5,0.6,0.5,0.7      //8 minute graph y axis values
COLOR_MENU11=0.5,0.6,0.5,0.7      //24h graph y axis values
COLOR_MENU12=0.18,0.20,0.18,1.0   //cell bargraph background
COLOR_MENU13=0.95,0.95,0.95,1.0   //wires color
COLOR_MENU14=1.0,1.0,1.0,0.9      //values color
COLOR_MENU15=0.6,0.7,0.6,1.0      //Date color
COLOR_MENU16=0.85,0.9,0.85,1.0    //Time color
COLOR_MENU17=0.72,0.82,0.72,1.0   //raw data color
COLOR_MENU18=0.18,0.20,0.18,0.9   //Flag background color
COLOR_MENU19=0.64,0.92,0.42,0.8   //non error flags collor
COLOR_MENU20=0.9,0.4,0.1,0.8      //error flag collor
COLOR_MENU21=0.6,0.6,0.6,0.7      //flag text color inactive
COLOR_MENU22=0.1,0.1,0.1,0.9      //flag text color active
COLOR_MENU23=0.14,0.12,0.72,1.0   //Buttons background collor
```

# *2 ElectroDacus csv converter application.*

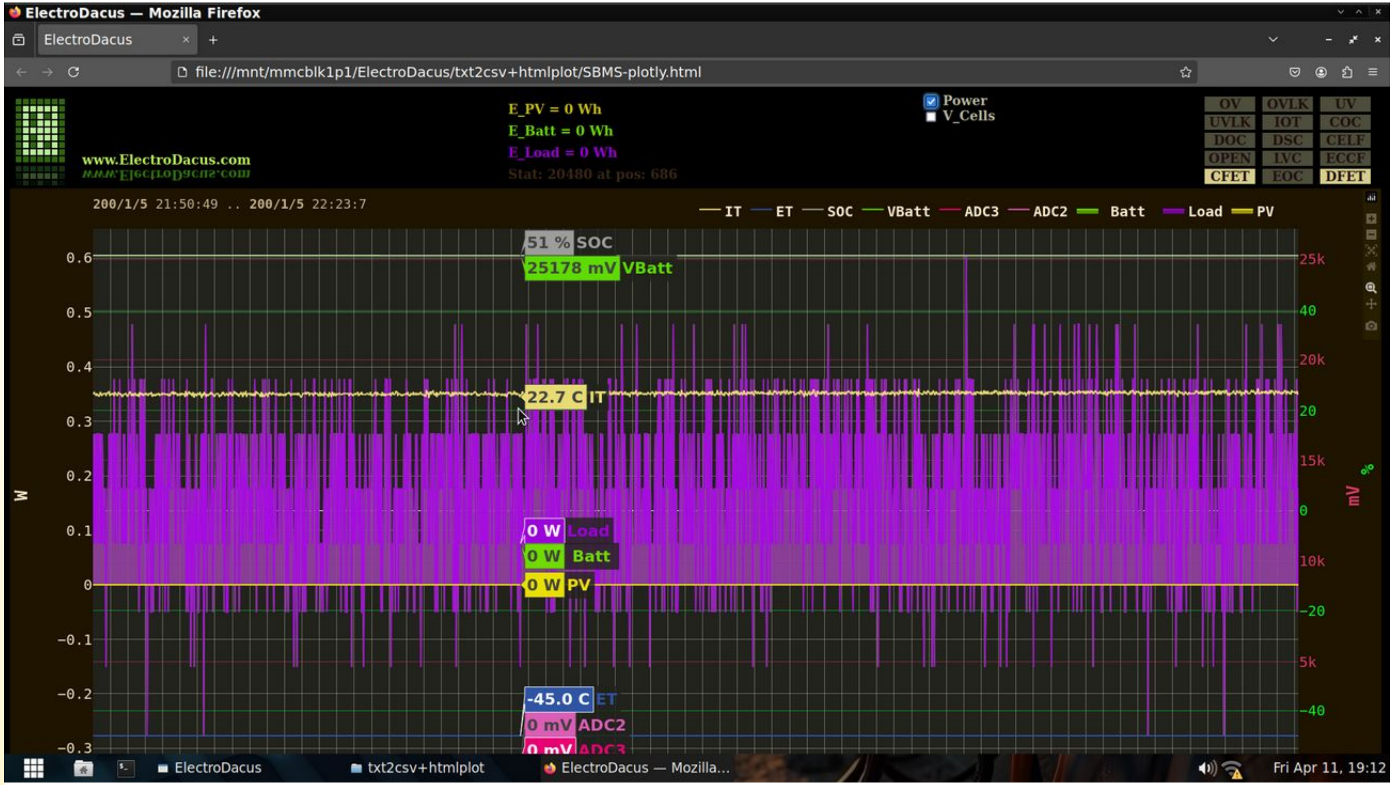
*You will also find a command line utility called txt2csv that will convert this yymmdd_Data.txt files to .csv so that you can use them in your preferred spreadsheet app.*
*The use is fairly simple as you see below just open a terminal by right clicking in that directory and select Open in Terminal the write the following "./txt2csv inputfile.txt outputfile.csv" and it should be converted in a second or two.*



```
guest@porteux:/mnt/mmcblk1p1/ElectroDacus/txt2csv+htmlplot$ ./txt2csv
Usage: ./txt2csv <input_file> <output_file> <output_file2>
guest@porteux:/mnt/mmcblk1p1/ElectroDacus/txt2csv+htmlplot$ ./txt2csv 000105_Data.txt 000105.csv
Converted data saved to: 000105.csv
Converted data saved to: All.js
guest@porteux:/mnt/mmcblk1p1/ElectroDacus/txt2csv+htmlplot$
```

# *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.*



ElectroDacus — Mozilla Firefox

file:///mnt/mmcblk1p1/ElectroDacus/txt2csv+htmlplot/SBMS-plotly.html

www.ElectroDacus.com

E_PV = 0 Wh
E_Batt = 0 Wh
E_Load = 0 Wh
Stat: 20480 at pos: 686

☑ Power
□ V_Cells

200/1/5 21:50:49 .. 200/1/5 22:23:7

51 % SOC
25178 mV VBatt
22.7 C IT
0 W Load
0 W Batt
0 W PV
-45.0 C ET
0 mV ADC2
0 mV ADC3

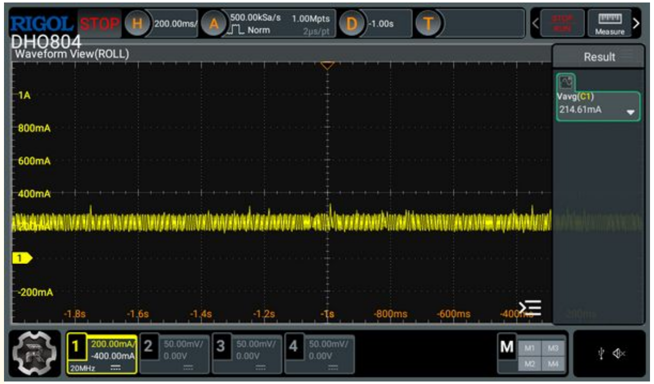
# *4 Power Consumption*

*The setup was supplied from a 12V bench power supply in series with a 1Ohm resistor that was used to measure voltage drop (Vdrop) across it on CH1 of the oscilloscope.*

*P = (12V – Vdrop) * I*

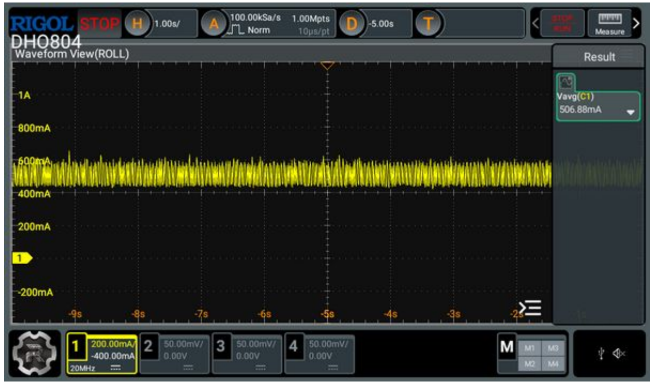
| | Wyse 3040 [W] | 14” USB powered LCD [W] | Total [W] |
|---|---|---|---|
| a) Idle no LCD | 2.53 | 0.00 | 2.53 |
| b) Idle LCD brightness 20% | 2.53 | 3.30 | 5.83 |
| c) Idle LCD brightness 100% | 2.53 | 6.13 | 8.66 |
| d) ElectroDacus app running no LCD | 2.82 | 0.00 | 2.82 |
| e) ElectroDacus app 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



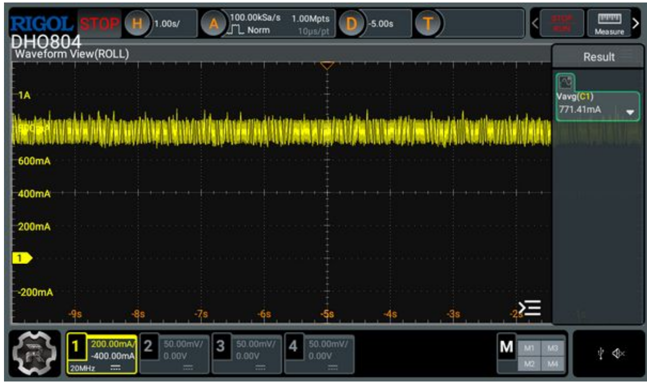
Vavg(C1) 214.61mA

b) Idle LCD brightness 20%



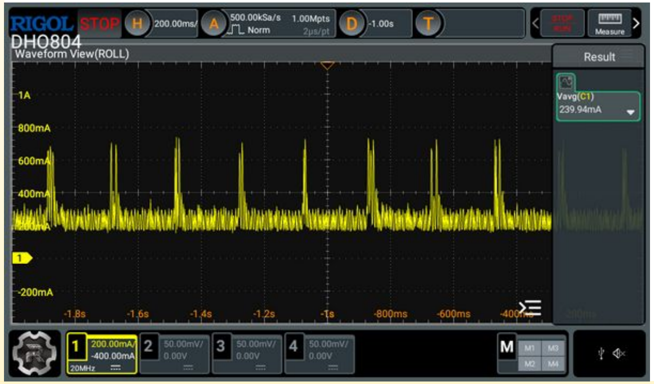
Vavg(C1) 506.88mA

c) Idle LCD brightness 100%



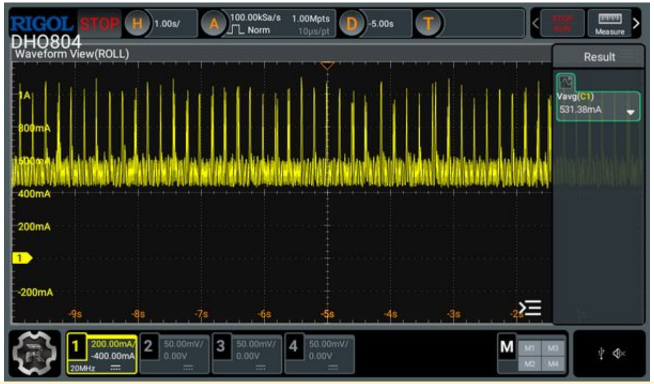
Vavg(C1) 771.41mA

d) ElectroDacus app running  no LCD



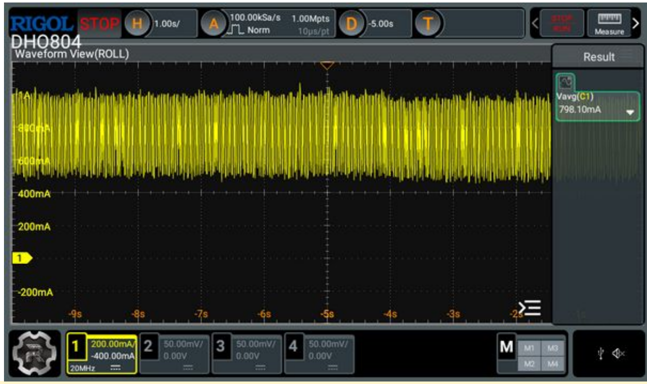
Vavg(C1) 239.94mA

e) ElectroDacus app running LCD brightness 20%



Vavg(C1) 531.38mA

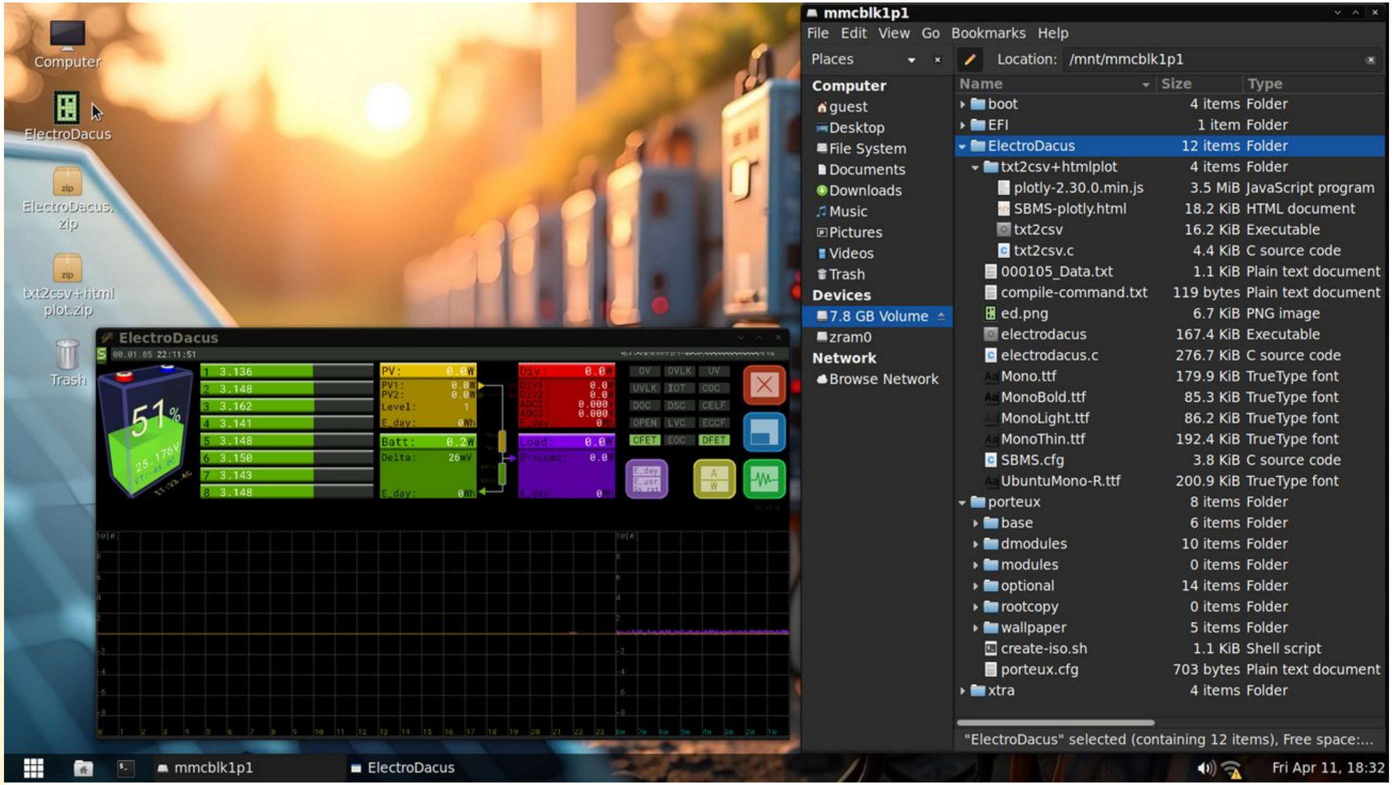
f) 1080p video playback LCD brightness 20%



Vavg(C1) 798.10mA

# *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 .*



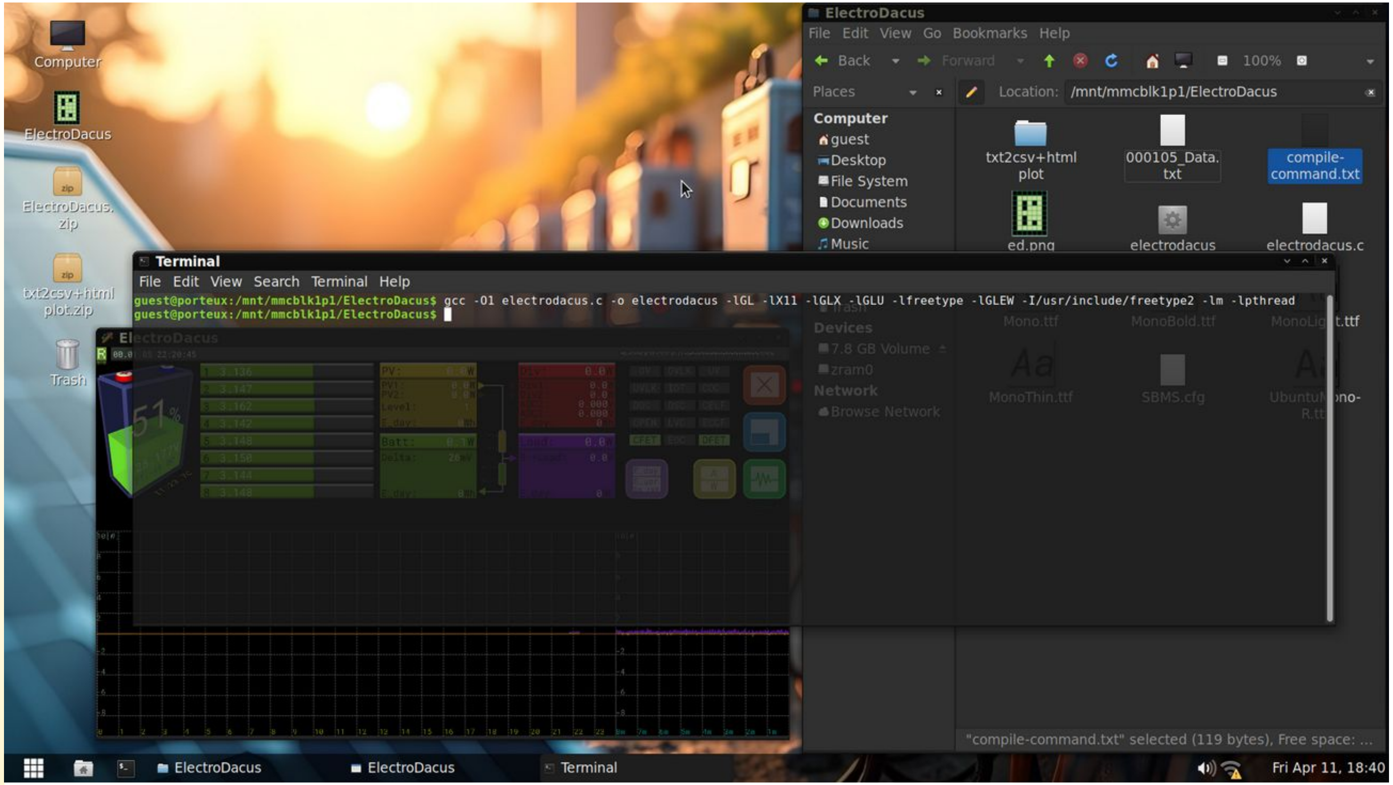
Location: /mnt/mmcblk1p1

| Name | Size | Type |
|---|---|---|
| boot | 4 items | Folder |
| EFI | 1 item | Folder |
| ElectroDacus | 12 items | Folder |
| txt2csv+htmlplot | 4 items | Folder |
| plotly-2.30.0.min.js | 3.5 MiB | JavaScript program |
| SBMS-plotly.html | 18.2 KiB | HTML document |
| txt2csv | 16.2 KiB | Executable |
| txt2csv.c | 4.4 KiB | C source code |
| 000105_Data.txt | 1.1 KiB | Plain text document |
| compile-command.txt | 119 bytes | Plain text document |
| ed.png | 6.7 KiB | PNG image |
| electrodacus | 167.4 KiB | Executable |
| electrodacus.c | 276.7 KiB | C source code |
| Mono.ttf | 179.9 KiB | TrueType font |
| MonoBold.ttf | 85.3 KiB | TrueType font |
| MonoLight.ttf | 86.2 KiB | TrueType font |
| MonoThin.ttf | 192.4 KiB | TrueType font |
| SBMS.cfg | 3.8 KiB | C source code |
| UbuntuMono-R.ttf | 200.9 KiB | TrueType font |
| porteux | 8 items | Folder |
| base | 6 items | Folder |
| dmodules | 10 items | Folder |
| modules | 0 items | Folder |
| optional | 14 items | Folder |
| rootcopy | 0 items | Folder |
| wallpaper | 5 items | Folder |
| create-iso.sh | 1.1 KiB | Shell script |
| porteux.cfg | 703 bytes | Plain text document |
| xtra | 4 items | Folder |

"ElectroDacus" selected (containing 12 items), Free space:...

# 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.*



```
guest@porteux:/mnt/mmcblk1p1/ElectroDacus$ gcc -O1 electrodacus.c -o electrodacus -lGL -lX11 -lGLX -lGLU -lfreetype -lGLEW -I/usr/include/freetype2 -lm -lpthread
guest@porteux:/mnt/mmcblk1p1/ElectroDacus$ 
```

Location: /mnt/mmcblk1p1/ElectroDacus

"compile-command.txt" selected (119 bytes), Free space: ...

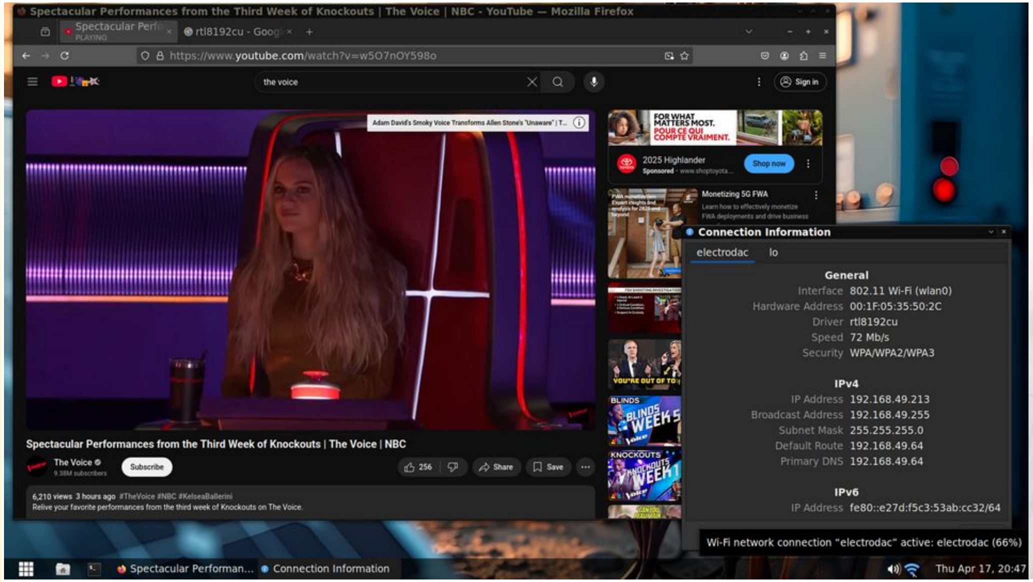
# 7 DELL Wyse 3040 (what is included).

*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.*

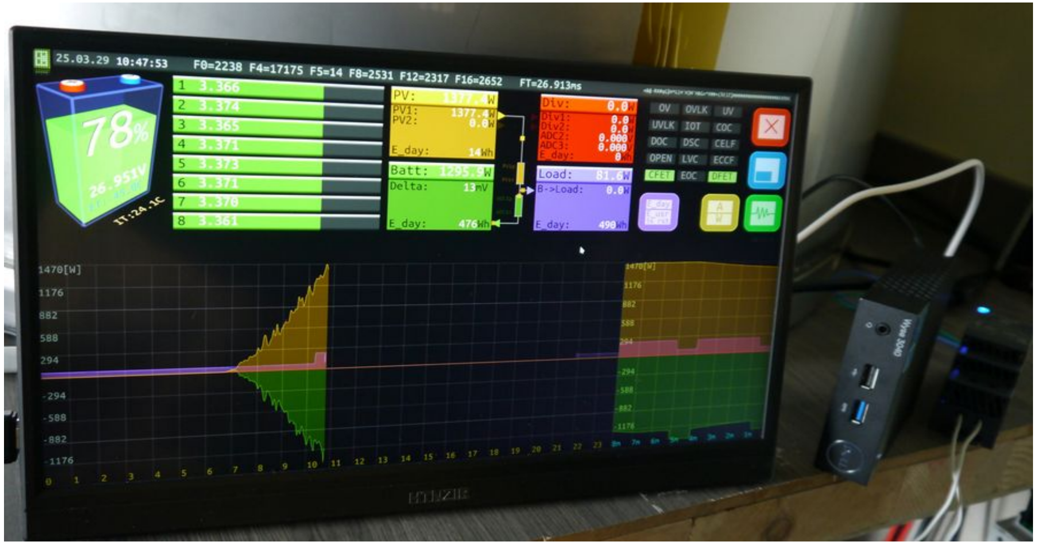
*USB extension is 25ft also from Amazon https://www.amazon.ca/dp/B0893R82RK It is the longest USB 2.0 passive cable I found as from 30ft and up most are active and of course those will work as they do signal amplification . To this 25ft I added another lower quality 6ft extension I had and a 3ft USB 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 .*


Spectacular Performances from the Third Week of Knockouts | The Voice | NBC - YouTube — Mozilla Firefox
Spectacular Perfo...
PLAYING
rtl8192cu - Googl
https://www.youtube.com/watch?v=w5O7nOY598o
the voice
Sign in
Spectacular Performances from the Third Week of Knockouts | The Voice | NBC
The Voice
Subscribe
256
Share
Save
2025 Highlander
Shop now
Monetizing 5G FWA
Spectacular Performan...
Connection Information
Thu Apr 17, 20:47

```
Connection Information
electrodac    lo

                    General
          Interface 802.11 Wi-Fi (wlan0)
   Hardware Address 00:1F:05:35:50:2C
             Driver rtl8192cu
              Speed 72 Mb/s
           Security WPA/WPA2/WPA3

                    IPv4
         IP Address 192.168.49.213
  Broadcast Address 192.168.49.255
        Subnet Mask 255.255.255.0
      Default Route 192.168.49.64
        Primary DNS 192.168.49.64

                    IPv6
         IP Address fe80::e27d:f5c3:53ab:cc32/64

Wi-Fi network connection "electrodac" active: electrodac (66%)
```

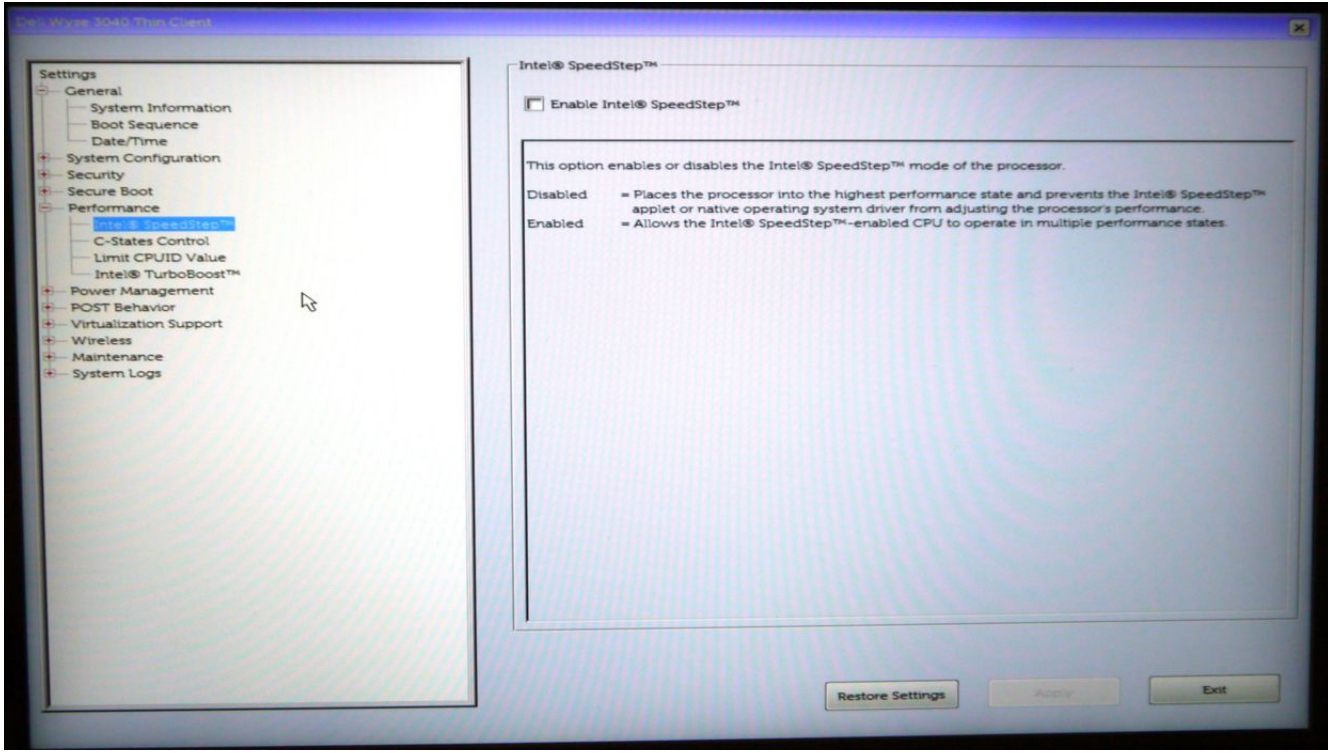
## … (accessories you may want).

*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. Juts 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.*



Intel® SpeedStep™

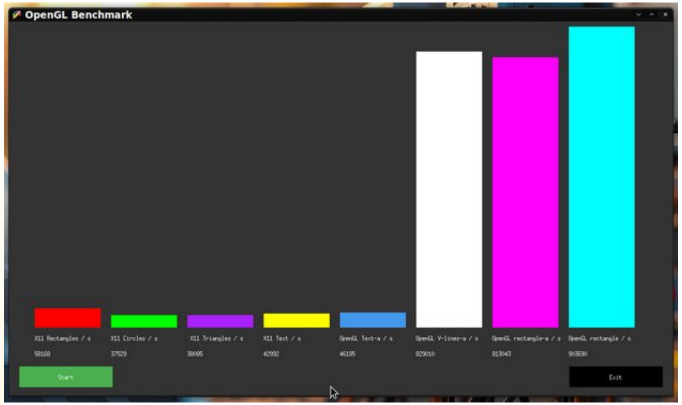
☐ Enable Intel® SpeedStep™

This option enables or disables the Intel® SpeedStep™ mode of the processor.

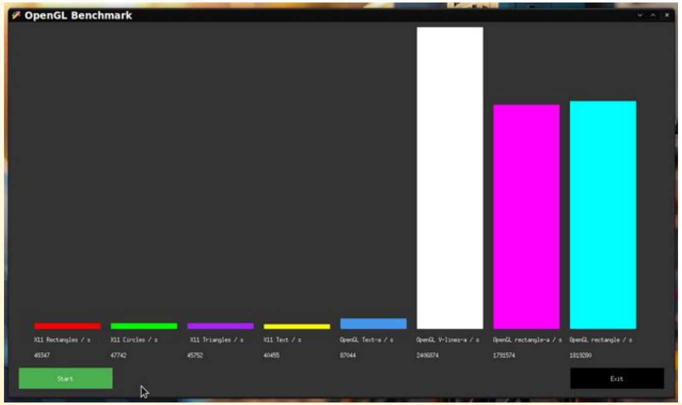
Disabled = Places the processor into the highest performance state and prevents the Intel® SpeedStep™ applet or native operating system driver from adjusting the processor's performance.
Enabled = Allows the Intel® SpeedStep™-enabled CPU to operate in multiple performance states.

# *9.1 BIOS settings (performance).*

*To increase performance I found that best setting is to disable everything in Performance so SpeedStep, C-States and TurboBoost.*
*This way the CPU will stay at a fix 1440Mhz instead of going trough many steps from 480MHz to 1920Mhz (turbo).*
*When the OS sees that an app or more with low CPU demand are started it keeps the CPU mostly at 480MHz but this also means lower iGPU performance.*

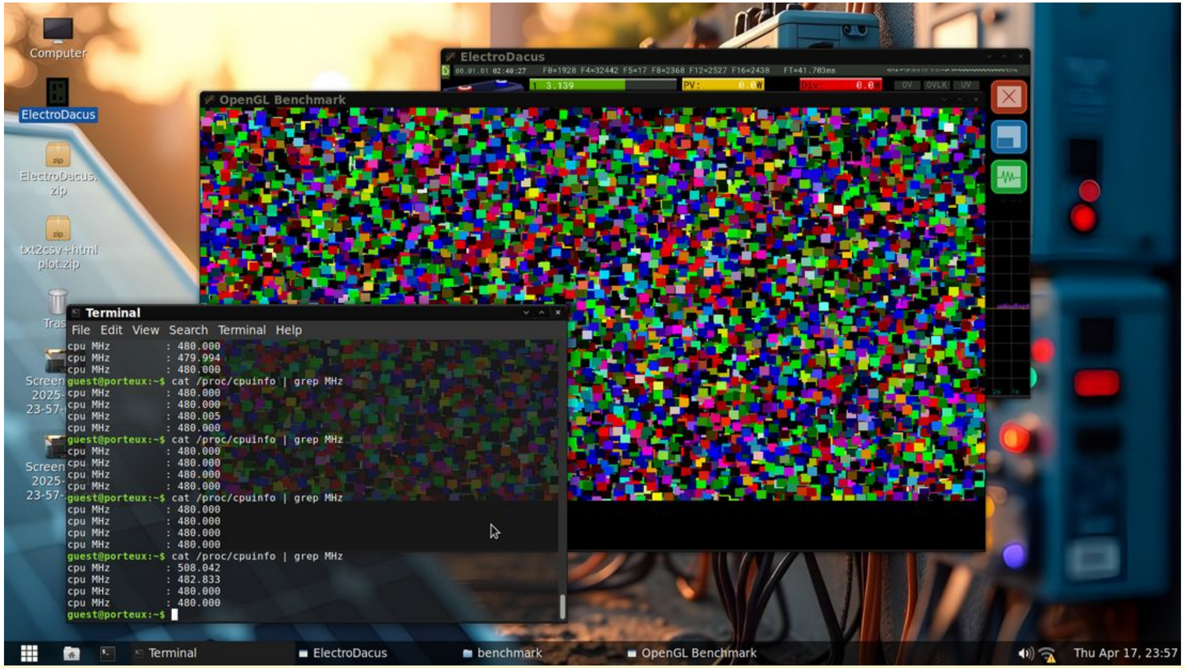


```
X11 Rectangles / s   X11 Circles / s   X11 Triangles / s   X11 Text / s   OpenGL Text-a / s   OpenGL V-lines-a / s   OpenGL rectangle-a / s   OpenGL rectangle / s
58168                37529             38085               42932          46185               829010                 813043                   903630
```



```
X11 Rectangles / s   X11 Circles / s   X11 Triangles / s   X11 Text / s   OpenGL Text-a / s   OpenGL V-lines-a / s   OpenGL rectangle-a / s   OpenGL rectangle / s
48347                47742             45752               40455          87044               2406874                1791574                  1819280
```

*You can notice the OpenGL Benchmark (I made to test X11 vs OpenGL) score is about 2x better when CPU frequency is fixed and no C-States and TurboBoost used.*
*Also notice the CPU frequency while running the OpenGL part of the benchmark as OpenGL mostly used the iGPU not much need for the CPU but because of that OS decides to save power and so both CPU and GPU are in power-save state.*



```
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.*