

BOREDOM PROJECTS

(/)

[Home \(/index.php\)](#)

[PROJECTS \(/index.php/projects\)](#)

[Files \(/index.php/files\)](#)

[WebLinks \(/index.php/web-links\)](#)

[Comments \(/index.php/comments\)](#)


[Privacy Policy \(/index.php/privacy-policy\)](#)



Home Energy Monitoring System

Details

 Created: 10 December 2013

 Last Updated: 28 January 2016



Contents

1. The Purpose (Mission) (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h1-the-purpose-mission>)
2. Components (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h2-components>)
3. Progress Report (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-progress-report>)
 - 3.1. Measurement Box (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-1-measurement-box>)
 - 3.1.1. Hardware (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-1-1-hardware>)
 - 3.1.2. RFM12B library for Arduino Due (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-1-2-rfm12b-library-for-arduino-due>)
 - 3.1.3. Current / Voltage Sensors (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-1-3-current-voltage-sensors>)
 - 3.1.4. Fitting it all inside my fuse box (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-1-4-fitting-it-all-inside-my-fuse-box>)

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

[Read more \(/index.php/privacy-policy\)](#)

3.2. Data Collection Box - Web Server

(<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-2-data-collection-box-web-server>)

3.2.1. Hardware (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-2-1-hardware>)

3.2.2. Software - JeeNodeUSB (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-2-2-software-jeenodeusb>)

3.2.3. Software - RaspberryPi (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h3-2-3-software-raspberrypi>)

4. Next Steps (<https://boredomprojects.net/index.php/projects/home-energy-monitor#h4-next-steps>)

1. The Purpose (Mission)

- Monitor, archive and analyze energy consumption in the house
- Visualize historical consumption data
- Access data remotely from Web
- Eventually include data from other sensors, like temperature, pressure, humidity, motion, light, security, etc.
 - UPDATE: Temperature, Pressure and Humidity sensors added: Additional Sensors for Home Energy Monitor (</index.php/projects/sensors-for-hems>)

2. Components

- Concept based on OpenEnergyMonitor (<http://www.openenergymonitor.org>) project
- Hardware based on Arduino (<http://www.arduino.cc>) and Raspberry Pi (<http://www.raspberrypi.org>) platforms
 - UPDATE: Raspberry Pi has now been replaced with Ubuntu NAS and Backup Server (</index.php/projects/ubuntu-nas-backup-server>)
- Software: Arduino IDE (<http://www.arduino.cc/en/Main/Software>), EmonLib (<https://github.com/openenergymonitor/EmonLib>) and RFM12B library by LowPowerLab (<http://lowpowerlab.com/blog/2012/12/28/rfm12b-arduino-library/>)

3. Progress Report

3.1. Measurement Box

To begin with, I looked at the electrical fuse box in my apartment and noticed that I have 10 or so circuit breakers leading to separate electrical house circuits. I thought that it would be nice to monitor these 10 output lines individually and independently. This way I could have detailed power data as well as compute total house consumption.

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

[Read more \(/index.php/privacy-policy\)](/index.php/privacy-policy)



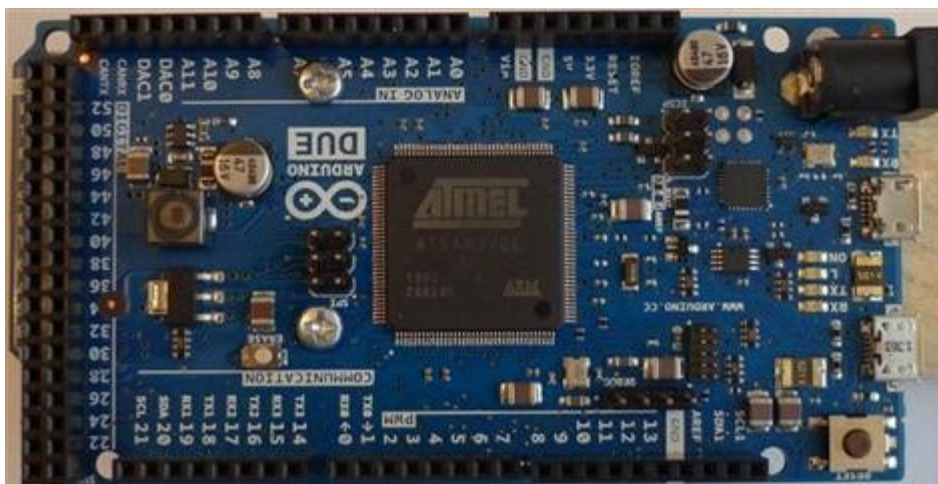
(/images/Articles/HomeEnergy/ebox-

closed-large.jpg)

When I removed the plastic cover, I noticed that there is quite a lot of space under it, so I hoped I could nicely fit all required hardware inside and hide it from view.

3.1.1. Hardware

As far as I could see, the original EmonTx (<http://openenergymonitor.org/emon/Hardware>) approach from OpenEnergyMonitor (<http://openenergymonitor.org/>) was to measure current at the input to the breakers. This could be the result of limited measurement speed and/or limited processing power or the Arduino Uno (<http://arduino.cc/en/Main/arduinoBoardUno>) -based hardware. EmonTx has only three current probe inputs with 10-bit resolution. So, instead I decided that I'd need faster processor, more analog inputs and 12-bit ADC resolution. In other words, this **Arduino Due** (<http://arduino.cc/en/Main/ArduinoBoardDue>) board:



(/images/Articles/HomeEnergy/ArduinoDue.JPG)

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

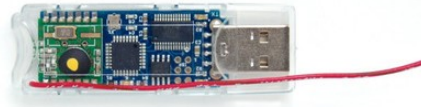
Read more (/index.php/privacy-policy)

The next logical step was to test wireless communication between this sensor controller and whatever the base might be. For this I purchased a couple of RFM12B (<http://jeelabs.com/products/rfm12b>) modules and a JeeLink (<http://jeelabs.com/products/jeelink>) from JeeLabs shop (<http://jeelabs.com/>):

(NOTE: JeeLink was later replaced with JeeNodeUSB)



(/images/Articles/HomeEnergy/RF12B_large.jpg)



(/images/Articles/HomeEnergy/jeelink-top_large.jpg)

3.1.2. RFM12B library for Arduino Due

As a starting point, I used a great port of RFM12B library to C++ by Felix of LowPowerLab.com (<http://lowpowerlab.com/blog/2012/12/28/rfm12b-arduino-library/>)

Turned out, this library was not compatible with my ARM-based Due board. So I spent the next few weeks on modifying this code and eventually made it to work on my Due. The files are below. Just unzip this into *library* folder of the Arduino IDE and don't forget to select Due before compiling ;)

https://github.com/icboredman/RFM12B/tree/arduino_due
(https://github.com/icboredman/RFM12B/tree/arduino_due)

Not all functionality has been implemented yet, like sleep mode, but overall it does work.

3.1.3. Current / Voltage Sensors

Once I had RF communication working properly between Due and JeeLink, I started working on setting up sensors.

I purchased 12 non-invasive current probes and 1 AC-AC voltage adapter:

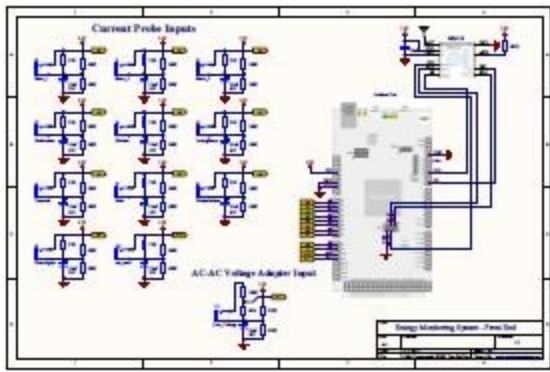


By visiting our website you agree that we are using cookies to ensure you to get the best experience.
(/images/Articles/HomeEnergy/SCT013-Small-rot.jpg) (I understand!)

(/images/Articles/HomeEnergy/AC-AC-Small.jpg) Read more (/index.php/privacy-policy)

Here is a great explanation of Power measurement theory using these sensors (<http://openenergymonitor.org/emon/buildingblocks>).

You can download a complete schematic diagram of my energy monitor here:



(/images/Articles/HomeEnergy/EMS_schematic.pdf)

As you can see, I used 22 Ohm burden resistors, which gave me more than enough head room to measure currents up to full range of SCT013 (100A). In addition, to measure line voltage using AC-AC adapter, I adjusted resistive divider to 100K - 6.6K, which slightly reduced resultant voltage range at the Arduino to below 3.3V p-p.

I soldered all components onto this Proto Shield for Arduino MEGA, (http://store.arduino.cc/index.php?main_page=product_info&cPath=37_5&products_id=205) which also fits perfectly to Due.

UPDATE (Jan. 2016):

Following a suggestion to convert the original crude wire-solder design into something that could be easily duplicated and manufactured, I've taken on Fritzing (<http://fritzing.org>). I must say, it is a pain-in-the-***. Nevertheless, here's the result, if anybody wants to try it:

<http://fritzing.org/projects/emontx-shield-for-arduino-due> (<http://fritzing.org/projects/emontx-shield-for-arduino-due>)

Warning, I have not tested this Fritzing design myself, so try at your own risk ;)

In order to utilize full 12-bit ADC resolution of Arduino Due, I reworked original EmonLib library of OpenEnergyProject (<https://github.com/openenergymonitor/EmonLib>). Here's my modified source code:

https://github.com/icboredman/EmonLib/tree/feature/12bit_3phase
(**https://github.com/icboredman/EmonLib/tree/feature/12bit_3phase**)

This mod also contains 3-phase buffer-and-delay algorithm, since my CT inputs are connected to all three line phases, but my AC_AC voltage adapter is connected to only one of them (obviously).

And finally, here's the Arduino sketch that ties this all together:

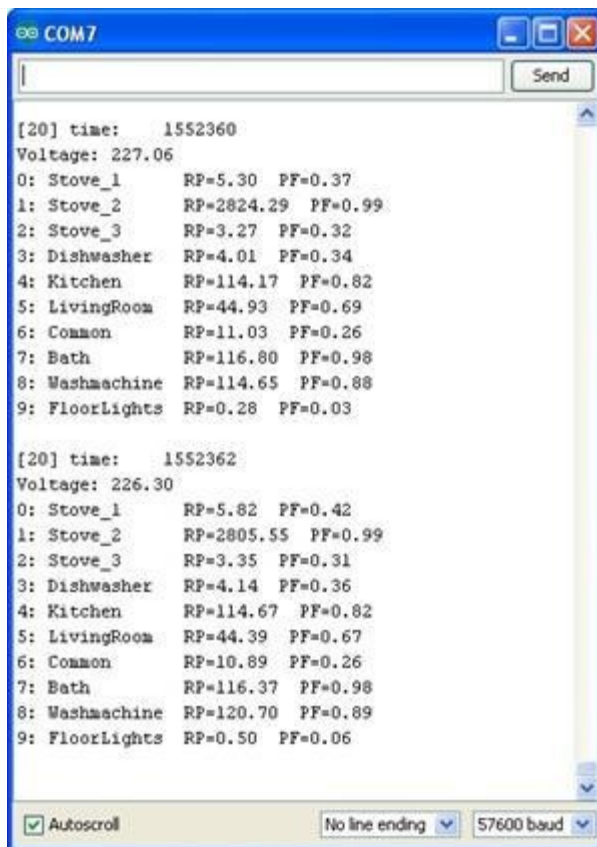
EmonTxDue.zip (</images/Articles/HomeEnergy/EmonTxDue.zip>)

At the end, after adjusting calibration constants, I was able to see this raw dump of sensor data, occurring every 2 sec:

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

[Read more \(/index.php/privacy-policy\)](/index.php/privacy-policy)



(/images/Articles/HomeEnergy/jeelink-

output.jpg)

NOTES:

1. The above formatted output was generated using this test sketch **EmonRxUno.ino** (</images/Articles/HomeEnergy/EmonRxUno.ino>), not to be used in working system.
2. The Arduino sketch for a working RaspberryPi-based system is here: Software (</index.php/projects/home-energy-monitor#h3-2-2-software-jeenodeusb>)

3.1.4. Fitting it all inside my fuse box

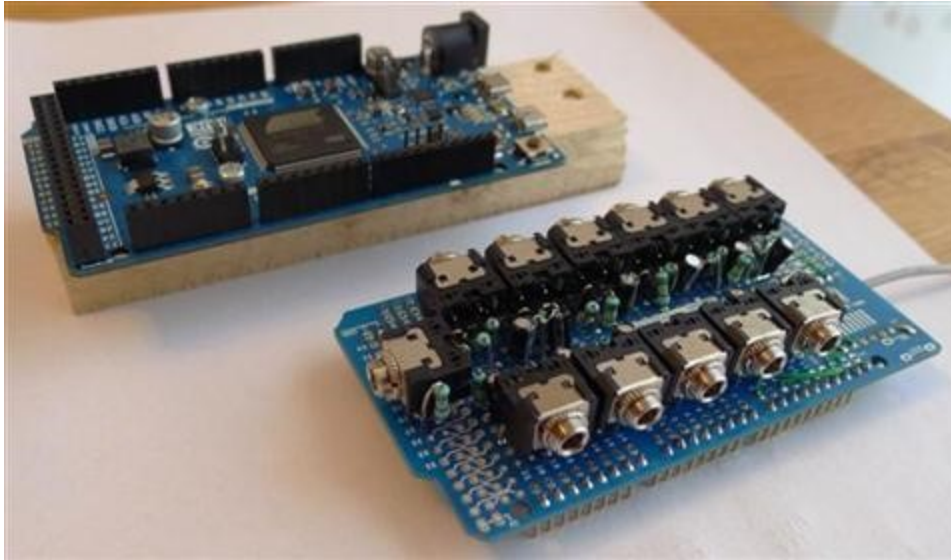
Here I will let pictures tell the story...

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

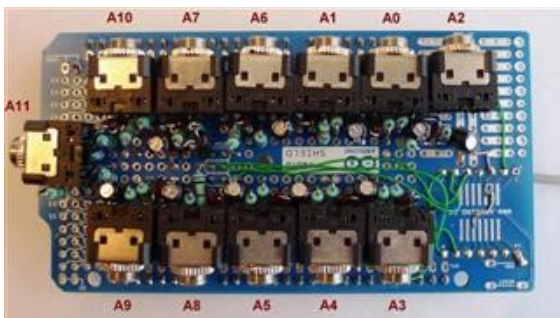
[Read more \(/index.php/privacy-policy\)](/index.php/privacy-policy)

Arduino Due and my EmonTx Shield



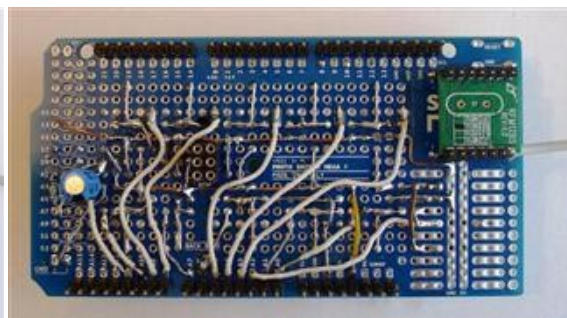
(/images/Articles/HomeEnergy/EmonTx-2parts-large.jpg)

EmonTx shield - Top



(/images/Articles/HomeEnergy/EmonTx-shield-top-large.jpg)

EmonTx shield - Bottom



(/images/Articles/HomeEnergy/EmonTx-shield-bot-large.jpg)

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

[Read more \(/index.php/privacy-policy\)](/index.php/privacy-policy)

Fuse box - Before



(/images/Articles/HomeEnergy/ebox-before-large.jpg)

Fuse box - After



(/images/Articles/HomeEnergy/ebox-after-large.jpg)

3.2. Data Collection Box - Web Server

- NOTE: Raspberry Pi has since been replaced with Ubuntu NAS and Backup Server
(/index.php/projects/ubuntu-nas-backup-server)

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I understand !

Read more (/index.php/privacy-policy)

3.2.1. Hardware



(/images/Articles/HomeEnergy/rpi_setup_lb.jpg)

All the components above are powered from a single 5V-2A Adapter through the USB Hub, as explained here: <http://www.raspberrypi.org/phpBB3/viewtopic.php?t=9070&p=108006>
(<http://www.raspberrypi.org/phpBB3/viewtopic.php?t=9070&p=108006>)

Plus, I found a hub with a very convenient power switch.

I decided to use a Solid State Drive mainly because it is silent, after I got fed up with constant rattling of my old HDD.

I also added a small mod to JeeNodeUSB to measure RSSI from RFM12B (<http://forum.jeelabs.net/comment/6203.html#comment-6203>) when packets are received. (<http://forum.jeelabs.net/comment/6203.html#comment-6203>)

3.2.2. Software - JeeNodeUSB

- Here's the sketch running inside this Arduino-based JeeNodeUSB: **EmonRxRPI.ino (/images/Articles/HomeEnergy/EmonRxRPI.ino)**.
- The configuration of RF parameters in this sketch is hard-coded. Modifying config at run time through emonhub is not implemented, at this time.

3.2.3. Software - RaspberryPi

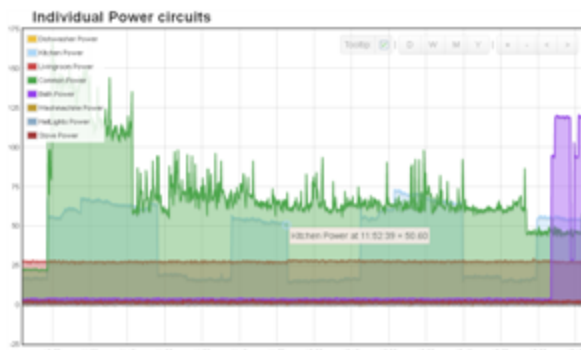
By visiting our website you agree that we are using cookies to ensure you to get the best experience.

I started with installing EmonCMS pre-built HDD image (<http://emoncms.org/site/docs/raspberry/hdd>) dated 30.11.2013. It is quite old, but the newer image from 20.01.2014 didn't work. See this thread! [Read more \(/index.php/privacy-policy\)](#)

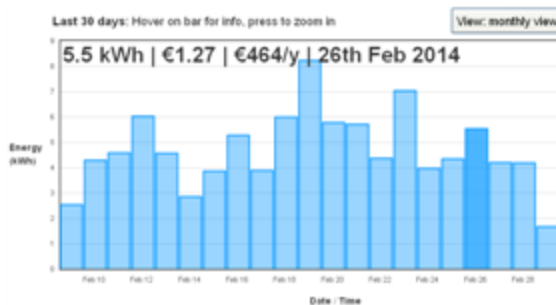
<http://openenergymonitor.org/emon/node/3688>

(<http://openenergymonitor.org/emon/node/3688>) (update: there seems to be a fix now! should give it a try.)

- Hardened SSH with key files and changed default access port. For that, modified firewall (ufw) rules by adding the new port there.
- Disabled registering new EmonCMS users: <http://openenergymonitor.org/emon/node/2116> (<http://openenergymonitor.org/emon/node/2116>)
- Enabled site Password, HTTPS and other security enhancements:
<http://blog.al4.co.nz/2011/05/setting-up-a-secure-ubuntu-lamp-server/>
(<http://blog.al4.co.nz/2011/05/setting-up-a-secure-ubuntu-lamp-server/>)
<https://help.ubuntu.com/10.04/serverguide/httpd.html>
(<https://help.ubuntu.com/10.04/serverguide/httpd.html>)
- Finally, configured some Feeds and Dashboards in my running EmonCMS:

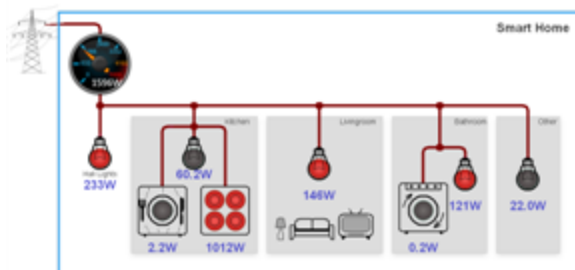


(/images/Articles/HomeEnergy/Individual.png)



Total: 99 kWh : €23 | Average: 4.7 kWh : €1.08 | €8 a week, €395 a year | Unit price: €0.23

(/images/Articles/HomeEnergy/Daily.png)



(/images/Articles/HomeEnergy/Live.png)

4. Next Steps

Something to think about...

By visiting our website you agree that we are using cookies to ensure you to get the best experience.

- Adding Temperature / Humidity sensors. Indoor and Outdoor.

I understand !

Read more (/index.php/privacy-policy)

- UPDATE: Additional Sensors for Home Energy Monitor (/index.php/projects/sensors-for-hems)
- Tying in some hardware controls, to remotely switch things on/off, for example.
- Creating a plugin for my MediaPortal box (/index.php/projects/media-portal) to show dashboards in it!
 - UPDATE: MediaPortal viewer for my dashboards (/index.php/projects/media-portal#h3-3-viewer-for-my-energy-monitor)

Comments (/index.php/component/jcomments/feed/com_content/2)



(/index.php/projects/home-energy-monitor#comment-15) **Mr. — Radovan** 2014-02-11 15:48

+1

Hi, Im folowing your step by creating multichannel energy meter based on Arduino Due. I just got in troubles. Having Arduino running your sketch and sending data via RF12B, Im unable to setup receiving arduino with other RF12B. The matter is, I have no sketch to that receiving site and I have no idea how to write correctly receiving commands. I need simply receive data sent by Due and print them via serial.print to serial monitor. Could you give me some hint, what is the correct syntax of data receiving commands set?
Thanks for any tip.
Radovan



(/index.php/projects/home-energy-monitor#comment-16) **RE: Mr. — boredman** 2014-02-11 18:31

+1

Hi Radovan,
Since as a starting point I used RFM12B library from LowPowerLab, which I modified to work on Due, you would also need to use this library (unmodified version, for Uno) on the receiving end. Here's the link to LowPowerLab's library: (<http://https://github.com/LowPowerLab/RFM12B>)
github.com/LowPowerLab/RFM12B
(<https://github.com/LowPowerLab/RFM12B>)
and here's my Uno sketch that uses this library:
boredomprojects.net/images/Articles/HomeEnergy/EmonRxUno.ino
(<http://boredomprojects.net/images/Articles/HomeEnergy/EmonRxUno.ino>)
I hope this helps!
Let me know your progress.



(/index.php/projects/home-energy-monitor#comment-304) **Use of RFM69HW instead of RFM12B — EddyBin** 2016-11-05 15:30

While collecting the required components I was told that RFM69HW-443-S2 would be backwards compatible with the RMF12B. Would it be a good idea to use RFM69HW-443-S2 in the EmonTx shield?



(/index.php/projects/home-energy-monitor#comment-305) **RE: Use of RFM69HW instead of RFM12B — boredman** 2016-11-07 07:35

Yes, I've also heard that RFM69 should be backwards compatible. Unfortunately, as far as I know, the existing library will not work with 69 directly, without modifications. I didn't researched this topic enough to understand why. There exist 69 specific libraries, although probably not for Due. You would have to make the mod yourself. Shouldn't be too difficult. The only difference is SPI related stuff.
If you find a way to talk to 69 on Due, please share! :)

By visiting our website you agree that we are using cookies to enhance your navigation. Our Cookie Policy



(/index.php/projects/home-energy-monitor#comment-306) **RE: Use of RFM69HW instead of RFM12B — rjsc2000** 2016-11-07 09:48

Quoting EddyBin:

I understand !

Read more (/index.php/privacy-policy)