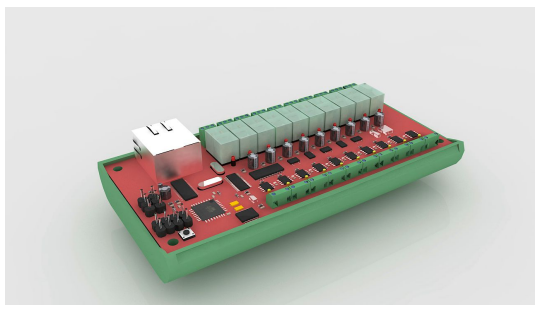
OpenStrom - Open Hardware Smart Meter Requirements

Please send feedback to m.zeitler@openstrom.com

The idea is to build an open hardware smart meter that can measure and control 10 circuits and will be used by makers and system integrators as part of their own smart electricity applications and projects. It will be connected via LAN[1]/wifi to the internet.
The main difference to existing smart meters is that we want to create an open solution (not proprietary systems), get a higher level of details (voltage/current/power factor, not only energy) and have relays to do some simple automation tasks compared to other commercial smart meters.
Also we want to get into a €30 price range for manufacturing costs (assuming contract manufacturing in China and an initial batch size of about 1000 units; not considering tooling and other manufacturing setup costs). This would allow a sales price around €100.



The solution consists of one (or more) din rail mounted hardware devices that plug directly into the main fuse box, the firmware that interacts with the sensors/relays and a simple backend server that logs the data and performs simple automation tasks.

The scope of this requirements document is the hardware aspect of the project. The backend system will be described in a different document.

We will provide a firmware that logs the data and accepts on/off commands for circuits via simple HTTP/XML communication. Special focus has to be put on the easy initial setup of the device, the ability to do firmware updates and safety/regulatory requirements.

Even though we expect others to use our smart meter in their solutions and will not design the final applications for endusers, our smart meter should support a wide range of use cases like:

- people that run data centers and want to know individual consumption data to determine if old hardware should be replaced

[1] There are currently no MCUs available that have Wifi and LAN on the chip. A possible solution for LAN is using Wiznet W5200, preferably as an add-on as it is not clear if we need to support wifi and LAN for most users.

- industrial or agricultural use cases to see if machinery has an unusual energy pattern (like a blocked pump).
- consumer use cases where people want to monitor their whole house and do simple home automation by switching circuits on and off

We will NOT cover the public power utility use case where energy consumption is metered and billed to the end customer. This use case has a lot of regulatory requirements and already huge competition, so we will stay away from it. This reduces our requirements and costs significantly.

The hardware has to be able to cover a range of use cases and we see the following requirements:

- measure electric power (P), reactive power (Q), apparent power (S), electric current (I), voltage (V), power factor (cos phi), others?
- detecting specific appliances through energy disaggregation[2]
  The idea is to do this on the backend (cloud), however depending on our sampling frequency this means we would potentially need to send a lot for data which could be a problem. So maybe a combination of some kind of “patterns” detection on the device and backend data processing is required.
  Also there are challenges getting into our target price range when aiming really high sampling (MHz range), so we will most likely limit this to 10kHz.
- detecting unusual patterns (e.g. a fridge using coolant and requiring more power than normal or a freezer that needs thawing/cleaning)
- switching off/on a whole circuit (we are not switching appliances, but the whole phase)
  This has a massive impact on size of the device and the cost to build it. I would prefer to build only one device (because of easier certification, testing, manufacturing, logistics etc.) instead of different models, but maybe we can make the switching functionality somewhat modular, so that the relays can be added via some kind of “shield” to the device to allow users to buy the device with and without switching functionality.
- combine multiple smart meters to interact with more than 10 circuits (most likely by combining them in the backend software). The management/aggregation of data will be done on the backend, so that we install multiple devices with the same hardware to accomplish this.
- fault tolerance (what happens when we lose internet connectivity or if we have a failure on one of the circuits). This might require some local data storage (if internet is lost) and some failure mode for switching (to prevent circuits to be permanently on or off when internet is lost). Also consider “missed” data packets and other similar error scenarios.

---

2 http://web.stanford.edu/group/peec/cgi-bin/docs/events/2011/becc/presentations/3%20Disaggregation%20The%20Holy%20Grail%20-%20Carrie%20Armel.pdf

Some initial considerations regarding components:

- Which MCU to use? Arduino would be great (very open), but maybe not enough processing power and possible issues with wifi
  Check: TI CC3200 and HLK-RMO4; also maybe ESP8266; also check MediaTek MT7681
- How to measure? Should we use a sensor chip or should we design our own circuit
  Check: ADE7753ARSZRL and CS5484; also maybe 78M6618 or STPM32/33/34 or ATM90E26
- Which relays should we use? Something like SSR25A might make it expensive for 10 circuits (especially if not everyone is using the switching functionality). Can the switching functionality be like an add-on “shield”?
  Check: T90S5D12-5 , jqx-15f , HF102F, JQX-102F
- Also consider power consumption of the device. Latching/solid state relays?
- Doing our own custom housing might be cost prohibitive for small production runs (as it would require injection molding). Can we fit our solution into a “standard” DIN rail housing? Which one?
- Everything should be as integrated as possible to reduce manufacturing and installation. Power supply has to be on the board itself.
- What is required to test the functionality of the device during manufacturing to prevent that we ship defective devices? Test points?
- The solution should be marketed worldwide (main focus: USA and European Union). There are several legal requirements to market electronic products in these markets like the CE mark, electromagnetic compatibility (EMC 2014/30/EU) and the low voltage directive 2014/35/EU. There might be other relevant regulations.
  The relevant legal requirements must be identified and considered when designing the device.

Roadmap:

Based on these initial requirements we will develop a technical specification and ask for feedback from the smart meter community and possible customers.
The idea is to build a working prototype, already designed for manufacturing and designed to cost.
After the prototype is completed, a vendor selection with different contract manufacturing companies will initiated to get quotes.
Then a crowdfunding campaign will be launched to presell the first production run.
In parallel we will develop the initial firmware and backend for the system.