Design and Building of a Cheap Smart Meter

Arne Ellerbrock, Student Member, IEEE, Ahmad Abdel-Majeed, Member, IEEE and Stefan Tenbohlen, Senior Member, IEEE

Institute of Power Transmission and High Voltage Technology, Pfaffenwaldring 47, 70569 Stuttgart, Universität Stuttgart, Germany.

Abstract-- **This paper shows the design of a cheap smart meter which is designed not only to measure the customer's power consumption and generation but also to enable and support the new operation and control functions in the distribution networks. It is based on open source hardware (Arduino and Arduino Ethernet Board) and offers a plurality of communication possibilities, like USB, Ethernet, ZigBee or Bluetooth. The smart meter uses an ADE7753 as converter and saves only the most important data (voltage, current, frequency, active- and reactive energy) on a SD-card so that all further calculations can be performed by an external central system. The smart meter can be used as energy meter (few minutes average) or as power quality meter (10 seconds average).**

I. Introduction

In Germany 69% of the installed Photovoltaic- (PV) power is installed in the low voltage network [1]. This causes rising power generation in the distribution network, loss in power quality and reversal of the power flow.

To be able to handle those things, monitoring and operating the grid is a necessary task. This way the voltage amplitude and frequency can be regulated constantly.

Nowadays, real-time monitoring and control through state estimation is a routine task for the transmission system operators (TSOs) due to the availability of measurement data. On the other hand distribution system operators (DSOs) are also trying to extend their monitoring and control for medium and low voltage networks in order to enable Smart grid applications. However; distribution network measurements are limited due to the lack of measurement devices and sensors there.

Through the rollout of smart meters, which are considered as a key component of the future Smart grids, there would be enough metering data (voltage, current, active and reactive power consumption and generation) available at every customer connection point to enable state estimation functions in the low voltage networks. This data can be used for alerting DSO's for certain power failures or exceeded permissible power quality limits. On the other hand, the smart meter could be also used as a terminal for demand side management (e.g. heat pumps). This rollout seems to be economically challenging and the need for inexpensive design of smart meters is needed to fulfill the gap in the required measurement data.

Unfortunately this data is still raw. Smart meters are expensive and not widespread.

II. Functional needs for smart meters with grid benefits

By analyzing the demand of users and State Estimation applications, we designed the smart meter with the following functions:

(1) Low material cost for the hardware and usage of open source software. Enables the possibility to use it as a source for numerous measurements of data.

(2) Simple design of the connection with plug and play function for the meter. This way the smart meter can be used as a normal power meter for connection points, but also as an analysis tool for single electric devices.

(3) Data collection function over short and long periods of time to learn about own energy consumption behavior of the users. It also gives the possibility to verify power flow calculations and state estimations.

(4) The function of storing measurement data for at least 12 months.

(5) Online Data collection for the execution of state estimations.

(6) Parallel handling of measurements, saving and loading meter data so that there are no mistakes through lack of constant measurements.

(7) Various implemented and potential interfaces to load and send the saved and online data to an external system with a graphical user interface (GUI).

(8) Intuitive and flexible GUI to make sure the smart meter and its data are useable for several people, especially without a technical background.

III. Hardware Design of Cheap Smart Meter

The hardware design of the smart meter is based on the open source micro controller Arduino ADK R3. This board enables the usage of Android devices to control the smart meter. The Arduino board is equipped with an Ethernet shield, which has a micro SD-card slot and an Ethernet interface. The measurement is done by an ADE 5573 and sent via a Serial Peripheral Interface (SPI). The result is available via LCD or an external system.

A. Selection of the microcontroller

The Arduino ADK R3 is an open source microcontroller which is easy to configure with different available shields and libraries. For example the Ethernet shield which offers an Ethernet interface as well as the possibility of a micro SD-card socket. As a memory for measuring data a two Gigabyte micro SD-card is used, which is enough for approximately two years worth of data.

Not only the possibility to integrate Android devices but also the quantity of analog and digital in- and outputs is a reason to choose this version of Arduino. The smart meter is planned as a single phase power meter with the possibility to be upgraded as a three phase smart meter. The Arduino ADK R3 has enough pins to be used as a three phase power meter with the same integrated circuit (IC).

Figure 1: Microcontroller Arduino ADK R3

B. Integrated circuit ADE 5573

The ADE 5573 from Analog devices is a cheap metering IC as shown in figure 2, which is able to use the SPI interface for sending metering data.

By using RMS-average-values and power integration – which are only interrupted while reading (304 ns) and writing (319 ns) data to the SPI – the Arduino can use its limited pulsing rate for saving measured data on the SD-card or communicate with an external device without reducing the accuracy of the measured data.

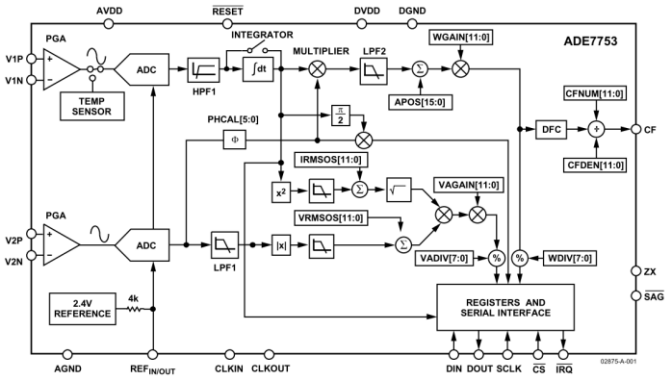

Figure 2: Functional block diagram of the ADE 5573 [2]

C. Design of the Metering shield

The metering shield designed with the ADE 5573 is used as a connection between measuring shunts and the Arduino. It is occupied with connection terminals, potential dividers, noise reduction electronics, an oscillator and the IC.

The dimensions of the board are based on the Arduino so they can be stacked together and build a stable device.

IV. Software Components of the Cheap Smart Meter

There are two main software components used for the smart meter. First the programming of the microcontroller, second the software for the external system. Both systems are using internal communication based on a serial interface. For the two different readouts described above there are different communication protocols.

A. Communication Protocol of SPI and SD-card

Using the SPI interface of the ADE 5573, the SPI library of the arduino.cc project is used [3]. The Arduino is using a loop to constantly check the time of the last read data. Once every ten minutes it reads the values of the IC using interruptions on both – IC and Arduinos – side. After reading the values for:

- Active energy consumption
- Reactive energy consumption
- Average active power
- Average reactive power
- Momentary value of voltage
- Average frequency

Those values are written with a time stamp and saved to a text file on the SD-card, using the SD library of the Arduino [4]. This method makes sure that the measured values are not inaccurate by using a delay between the measurements of the different values because of the long writing time of the SD library.

B. Communication Protocol for the serial port interface

If an external system is trying to get access to the Arduino data it has to use the serial port. Depending on which mode is used, it sends a different numeric code into the serial port buffer. After every loop, the Arduino is trying to read this buffer. If the buffer is not empty it decides what to do based on the numeric code.

This offers different access permissions for different users or systems, but also an easy way to make sure both sides know which reading mode is used. If the external system gets permission, there are two different reading modes. The first one is sending historical data from the SD-card to the external system. The second one is sending instantaneous data.

Because measuring and saving the data has the highest priority, the software is developed on an agent basis. Here it means both modes are aware of the time, what the next values to measure are, and are interrupted while getting the next values.

C. Visualization by the external system

The external system is based on a graphical user interface (GUI). This allows the user to easily choose what kind of data and – if historical data – which time period he wishes to see. This data may be presented in a table or a graph.

The interface used here is a minor problem, because USB, Ethernet and Bluetooth can be used as a serial port communication.

V. CONCLUSION

Low cost for smart meters are a major requirement. The described example costs less than 100€. This includes all measurement and connection hardware. Compared to free buyable products this is around 1/3 of the price [5], so that the requirement (1) can be considered as fulfilled.

Thanks to the internal memory of the Arduino the program automatically runs after getting power. This enables the in requirement (2) described plug and play functionality. Only the software of the external system has to be synchronized with the hardware.

The requirements (3) to (6) and (8) are fulfilled through interrupt and agent based software programming. Due to the usage of serial ports the interface being used is not determined, as specified by the requirement (7). Depending on the use case the chosen communication is wired or non-wired.

The smart meter achieves all requirements mentioned in section 2, which are recorded for customer usage and generation as well as to enable state estimation functions in the low voltage networks. Since these requirements are fulfilled the smart meter can be considered as suitable.

VI. OUTLOOK

Currently, smart phones and tablet computers are becoming more and more popular. This makes readout via Android very important. Since the Arduino has the Android USB interface the next plausible step would be to develop an Android App, which allows the user to visualize his energy consumption.

On the other side there are processes such as state estimations, used by utility companies. The live Data is especially important because readout via cable is not sufficient. The Ethernet board has to be used for online transmissions to webservers.

Due to the lack of multitasking possibilities of the Arduino the Agents have to be extended to decide whether personal or industrial usage of measurement data has a higher priority.

REFERENCES

[1] Abdel-Majeed, Ahmad, Haonan Wang, Martin Braun, and Daniel Schöllhorn. "Zustandsschätzung in Niederspannungsnetzen mit Hilfe von Smart Metern." In VDE-Kongress 2012.

[2] Analog Devices, “Single-Phase Multifunction Metering IC with di/dt Sensor Interface”, Norwood, U.S.A., 2010.

[3] Cristian Maglie, “SPI Master library for arduino“, published under GNU v2.1, 2010.

[4] SparkFun Electronics, “SD – a slightly more friendly wrapper for sdfatlib“, published under GNU v3, 2010.

[5] Arne Ellerbrock, Jörn Loviscach; “Das Stromnetz”; c’t magazin 02/2010.