Distributed Power Monitoring System [DPMS] – EasyPower

ECE Capstone Project, Spring 2013

Avinash Oza
Aakash Patel
Mozam Todiwala
Rylan Uherek

Advisor: Dr. Manish Parashar

Introduction:

In the energy-centric and environmentally friendly world we live in, consumers have access to a host of electronic devices which are designed to help them monitor and conserve energy. Yet, such devices lack certain features. Our group has developed a device and software to monitor not just the power consumption of a *single* device, but of a full building. Users can connect our device to their house, and via our website (or Android device) view their power usage over time, their estimated energy usage for the month, and past power bills. We provide similar functionality for a would-be power company to use our platform: they can monitor *aggregate* usage of all users of the platform.

Motivation:

As the economy requires Americans to cut back on power usage, consumers are looking to find methods to save power. Comparable devices such as the 'Kill-A-Watt' are limited to one device, and cannot give an estimated power bill. Similarly, you must be physically at the device to check usage. If there are thermostats which can change temperature via the internet, or android apps to start your car, why not a piece of software to track your power usage?

Design:

Our system is built in two stages: a hardware and software stage. The hardware is built using the Arduino board and current transducers. The Arduino board is a programmable logic controller which uses the current transducers to sense the amount of current going through a wire. It can use this information to calculate power usage via Ampere's Law. The Arduino connects to a web server via a PHP script to update usage for users. The web server runs Python and MySQL. The website is built using the Django (Python based MVC) open-source framework. Our mobile application is a wrapped version of the desktop site, packaged into an Android APK file.

Fig 1. A screenshot of the EasyPower Home Page

Conclusion:

The project proved to be a success. We broke the project into two components: the hardware and software development. Our team developed each component in parallel to speed up design and development time. In order to validate the accuracy of the device we designed, we benchmarked our Arduino against the 'Kill-A-Watt' device. Our results differed from the Kill-A-Watt with a margin of 5% accuracy, which we considered reasonable.

Cost:

Our design is very cost-effective. The Arduino device cost approximately $30, and the Ethernet shield, to connect it to the internet, was also $30. The current transducer cost $50. The software we used to develop our project was open source, and free. It cost approximately $110 to build our device, and connect a user to our system. Most users would be more than willing to pay that cost in lieu of the potential energy savings.