WIP: Advanced Real-Time Grid Energy Monitor System Project

Kyle Levesque, Charles White, Ikaika Mokiao, Kristian Kilcommons; Faculty Advisor: Kevin Davies
Dept. of Electrical Engineering, University of Hawaii at Manoa

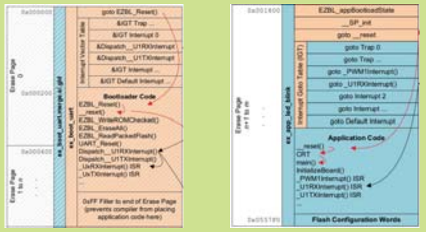
Program Memory

By: Kyle Levesque

The objective was to enable Run-Time Self-Programming (RTSP) on the microcontroller for over-the-air configuration. A bootloader application allows a microcontroller to RTSP when sent a configuration file. Microchip provided a program called EZ Bootloader, which contains a library for creating bootloader applications.

Benefits of EZ Bootloader:

- Keeps track of bootloader memory to prevent accidental erasure
- Can receive data from UART or I2C
- Supports PIC24 and dsPIC33 devices



The work pursued included creating an initialization file, merging bootloader code into our main application and loading the program into flash memory.

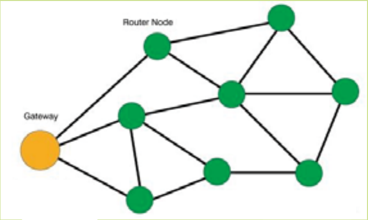
RF Communications

By: Kristian Kilcommons

Every Gateway device requires a data plan to contact the server. The mesh network is crucial to the economics of this project because it allows multiple devices to use the same data plan. The remote devices relay their readings through the XBee RF module. This semester software was developed so that the device could take advantage of the XBee's API mode.

Advantage of API mode:

- Transmission to and from multiple nodes
- Received packets identify sender
- Success/Failure of each transmitted RF packet
- Wireless configuration of remote XBees



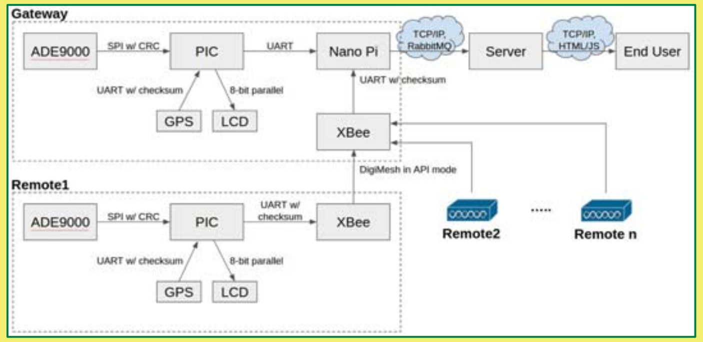
Example of the mesh network. Hopping of nodes extends the range of the network.

Introduction & Motivation

In order to achieve Hawaii's aggressive Renewable Portfolio Standard (RPS) targets, high amounts of variable renewable energy must be integrated into the grid. The transient nature of renewable energy requires careful monitoring of the electric grid. However, current methods of distribution level grid monitoring are large and costly. Our project aims to implement a flexible, high-resolution, low cost metering solution to help the utility implement smart-grid technologies and increase the penetration of renewable energy.

Project Description

Our devices measure voltage, current and power at the distribution level of the electrical grid. The architecture consists of a mesh network with remote meters which pass data to gateways for collection. Remote devices consist of a microcontroller which interfaces to a power monitor IC, GPS and LCD. The microcontroller sends data to a central gateway device via a wireless RF module. The gateway device hosts a single board computer that streams data to a server. The server provides a database with a web interface for data analytics. A block diagram of the components and data flow is shown below:



Materials & Methods

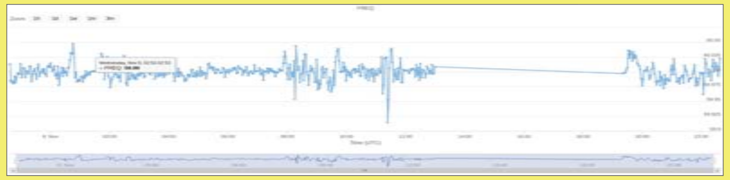
The device has the following parts:

- Power Monitoring IC
- Microcontroller
- GPS
- Wireless RF Module
- LCD
- Single-Board Computer

We employed a modular approach to allow students to work independently to accomplish the overall project. The following sections (noted by the green panels) were given to students for development, testing and deployment on the hardware platform: Over the air programming and configuration, wireless networking and communications, real time signal processing and SPI interfacing between power monitor IC and microcontroller. Project management was accomplished during biweekly meetings to discuss the status of each student. Rapid prototyping was accomplished using tools such as MATLAB, Python, Digi's XBee software and Microchip's MPLab.

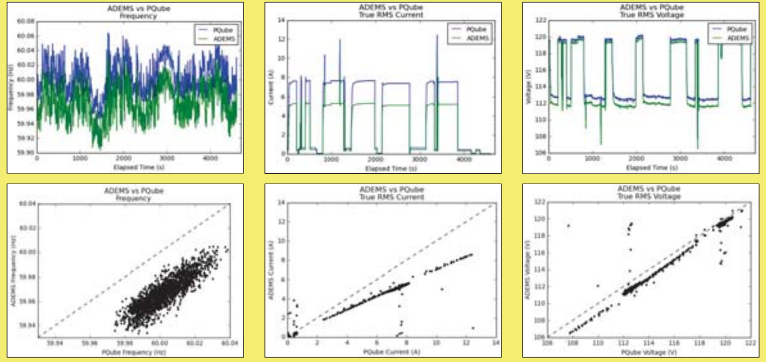
Results & Analysis

Server Log of Frequency vs. Time



This plot shows real time streaming of power monitor readings to our server. Currently we are sampling at one sample per second for 24 different variables including: three phase voltage, current, active and reactive power, device temperature and GPS timestamp.

Benchmarking



A benchmark was created between our unit and a commercial power monitor, PQube. Our results for frequency and RMS voltage were similar between units (0.05% and 0.25%, respectively) but there was a 40% difference in RMS current, which is being further investigated. This data comes from default scaling based on the power monitor manual. Calibration is expected to improve accuracy.

Conclusions

This semester we demonstrated that

- A 16 bit microcontroller is capable of performing the tasks required of a power monitoring device.
- A 10 tap bandpass FIR filter can be designed to locate the fundamental line frequency.
- A bootloader application requires careful manipulation of flash memory, making EZ Bootloader an attractive solution.
- XBee DigiMesh can aggregate measurements from multiple remote devices and lower the data cost.

This work has contributed to the development of a low cost power monitor to help integrate high penetrations of renewable energy.

Acknowledgements

We would like to give a special thanks to our instructor Kevin Davies and mentors Thai Tran, Motin Howlader and Saeed Sepasi of Hawaii Natural Energy Institute. They have given us continued guidance to be able to perform each section to the best of our abilities. Also, without all of them, we wouldn't be able to have the privilege of working on this project with this team. We greatly enjoyed working on this project.

Signal Processing

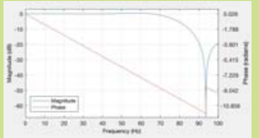
By: Ikaika Mokiao

The objective of this task was to design a digital filter that could identify the fundamental components of the voltage and current measurements. To accomplish this, MATLAB's Filter Designer was used. FIR and IIR filters were simulated prior to implementation on the microcontroller.

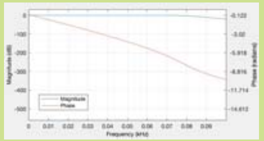
Filter Design Guidelines:

- Lowpass filter with a cutoff frequency of 80Hz
- Filter order ≤ 13 due to computational limits
- Sampling rate of either 8 kS/s or 128 samples per line cycle

FIR Kaiser Window 10th order filter



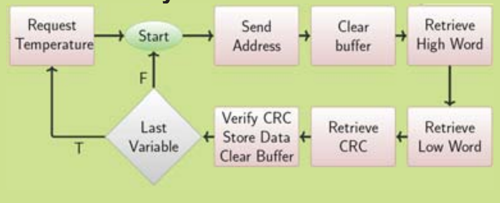
IIR Butterworth 10th order filter



The IIR filter achieves the required filter performance with a lower order than the FIR filter.

SPI Interfacing

by: Charles White



Shown above is the diagram of the state machine that our PIC used to communicate with the Power Monitoring Chip.

- The PIC utilized a hardware based cyclical redundancy check that assigns an Nth order polynomial (shown below) to an Nth number of bits which is then divided by a generated polynomial of the same order where the remainders are then compared.

$$A_n \cdot x^n + A_{n-1} \cdot x^{n-1} + A_{n-2} \cdot x^{n-2} + \cdots + A_0 \cdot x^0$$

- Our current configuration uses a traditional current transformer (iron core), but if required a Rogowski coil (air core) can also be used.

Afterwards data is scaled and transmitted over UART to a server where further calculations are made using Python.