Proposal for Smart Home Energy Management System

Sponsored by: SDG&E

Submitted by:

Hardware Team
Jose Cardenas
Kyle Dickerson
Wade Dorough
Saul Lima
Marizol Perez
Bob Severson

Software Team
Jhen Almeda
Wahid Baha
Jodi Quilalang
Chris Skalka
James Smith

Submitted to:

John Kennedy and Lal Tummala
Design Co. Ltd, San Diego, CA
Abstract

Smart Energy Management Solutions will build a model to demonstrate the vision that San Diego Gas & Electric has for the future of home energy management. This model will include real time, device-specific power monitoring and switching, as appropriate. It will use a graphical user interface to provide information to the user, which includes power consumption and cost-of-use. The model will use scaled-down devices to represent real-world household appliances and the method of power management for each device or appliance. These scaled-down devices will include a mini-refrigerator to represent a full-sized refrigerator, a siphon pump to represent a pool pump, a rechargeable radio-controlled car to represent a full-sized electric car, and a clothes dryer control panel with load data to represent a working clothes dryer.
Table of Contents

1 Project Description .............................................................................................................. 1
  1.1 Background Information .............................................................................................. 1
  1.2 The Smart Home Energy Solution ................................................................................. 1
2 Design .................................................................................................................................. 2
  2.1 Block Diagrams ............................................................................................................. 2
    2.1.1 Component Description ......................................................................................... 3
  2.2 Mock-Up Illustrations .................................................................................................... 5
  2.3 Performance Requirements .......................................................................................... 8
    2.3.1 Power Monitors ...................................................................................................... 8
    2.3.2 Radio Network ........................................................................................................ 8
    2.3.3 Graphical User Interface ....................................................................................... 8
3 Testing and Verification ....................................................................................................... 9
  3.1 Testing Procedures ....................................................................................................... 9
    3.1.1 Hardware ............................................................................................................... 9
    3.1.2 Software ............................................................................................................... 10
  3.2 Benchmarks .................................................................................................................. 11
4 Project Management .......................................................................................................... 11
  4.1 Project Plan .................................................................................................................. 11
  4.2 Milestones .................................................................................................................... 11
5 Cost Analysis ..................................................................................................................... 12
Appendix A Gantt Charts and Milestones ............................................................................ A-1
List of Figures

Figure 1: System-level block diagram ............................................................................................2
Figure 2: Node-level block diagram .................................................................................................3
Figure 3: Proposed home screen for Android App. ..........................................................................5
Figure 4: Proposed appliance screen for Android App .................................................................6
Figure 5: Proposed hardware mock-up ............................................................................................7
Figure 6: Allocation of funds ............................................................................................................12

List of Tables

Table 1: Project Milestones and Anticipated Dates of Completion................................................. A-4
1 Project Description

1.1 Background Information

The ever-increasing demand for energy has been putting a strain on utility companies for decades. Energy companies struggle during peak usage times to provide their customers with a continuous flow of power. The result is the creation of a new tiered pricing system that attempts to reward customers who “flex their power” during peak times. With the advent of “smart meters”, energy companies are able to charge customers in real time. Unlike their energy companies, this creates a blind spot for the consumer who does not have a window into their own power usage. We at Smart Energy Management Solutions aim to fill this gap and provide the customer with an easy to use system that enables quick monitoring of the appliances in their household.

1.2 The Smart Home Energy Solution

Smart Energy Management Solutions (SEMS) will design and build a smart home energy network in order to allow the consumer to monitor and control their appliances. The system will consist of three elements: the monitoring equipment, a central communication gateway, and a tablet computer. Multiple energy monitoring modules will be produced to monitor various loads throughout the home. These modules will be capable of reporting the energy consumption of the appliances they are connected to as well as provide a virtual on/off switch. Data will be transmitted to the central base station via an ISM band radio (802.15) where it will be buffered and retransmitted over WiFi (802.11) to a tablet computer. Running on the Android operating system, the tablet will present the data in a user friendly manner over multiple graphical user interfaces. The user will be able to view past and present data of their energy consumption as well as have the ability to turn on/off their appliances from within the Android application. Most importantly, the application will be able to take into account the tiered pricing of energy and see how much their appliances are costing them at any given time of the day. This type of system will allow consumers to be more aware of their power consumption and help the power company and power consumer save money.
2 Design

2.1 Block Diagrams

Figure 1: System-level block diagram
2.1.1 Component Description

The following list provides descriptions for the components in the system- and node-level block diagrams:

- **Cortex M0 (LPC1114)**
  - Central processor for each node
  - Sends commands to power chip via SPI
  - Receives voltage, current, and power information from the power metering chip
  - Turns solid-state relay on or off
  - Sends voltage, current, and power information to the tablet via Xbee radio

- **Manual Override**
  - Allows the user to bypass the on/off state received from the tablet’s automation system
• Cirrus Logic CS5464
  o Power metering IC
  o Receives input from current monitoring sensor and voltage divider network
  o Provides amplification of the analog current and voltage signals
  o Samples analog voltage and current signals and converts them to digital via internal ADC
  o Power, voltage, and current readings are held in on-board registers to be read by microcontroller
  o Communicates with microcontroller via SPI interface
• Xbee Radio
  o Provides two-way communication between node and tablet via gateway
  o Utilizes Zigbee protocol
  o Simple UART to wireless interface
• Current Monitor
  o Shunt Resistor
    ▪ Provides current sensing capability for loads greater than 1A
  o Hall-Effect Sensor
    ▪ Provides current sensing capability for loads less than 1A
• Voltage Divider
  o Scales voltage from 120VAC to a voltage the metering IC can tolerate
• Netburner Embedded Web Server (NNDK-MOD5282-KIT)
  o Functions as a TCP server for the entire system
  o Stores/sends data received from Cortex M0 and tablet
  o Gateway between Zigbee and WiFi protocols
• Router/Access Point
  o Allows communication between tablet and embedded web server through WiFi
• Asus Eee Pad Transformer (Tablet)
  o Provides a graphical user interface (via Android App) that allows the user to monitor and control household appliances and other power-consuming devices
2.2 Mock-Up Illustrations

![Smart Energy Management Solutions](image)

**Figure 3**: Proposed home screen for Android App.
Figure 4: Proposed appliance screen for Android App
Figure 5: Proposed hardware mock-up
2.3 Performance Requirements

Performance requirements for the finished project are specified below.

2.3.1 Power Monitors

The power monitors will include

- Current sensors
  - Coarse sensor for higher power draw devices
  - Fine sensor for lower power draw devices
- Power switching capability
- Manual override to prevent inappropriate power cutoff
- Radio module for communication over radio network

2.3.2 Radio Network

The radio network will include

- Radio modules on each power-monitoring node
- Gateway node for translation between XBee and WiFi protocols
- WiFi-capable tablet computer

2.3.3 Graphical User Interface

The graphical user interface will include

- State of each device (on/off)
- Cost-of-use for each device (whether on or off)
- Autonomous switching control for each node
- Preset limits for automatic switching of nodes
- Real-time graphical generation of data
3 Testing and Verification

3.1 Testing Procedures

The process for testing components will be split into two parts: hardware and software. This allows us to focus on each part we use and find solutions more rapidly if problems surface.

3.1.1 Hardware

The voltage source implemented for this design will be a 5VDC power supply. The current sensor will be incorporated into our design because we need to monitor the current being sampled by each appliance. In order to measure the voltages, a voltage sensor composed of a voltage divider made up of two resistors in series will be used. At this point, the power chip can read off the RMS voltage, RMS current, line voltage, and line current. The power chip can also be used to measure the active, reactive, and apparent powers. The power chip is optimized to handle shunt resistors for current measurement and voltage dividers for voltage measurement. The power chip also provides a calibration system: it has an AC offset calibration which can be used to measure the RMS voltage and RMS current. Therefore, testing can be verified by using a digital multimeter and an oscilloscope (to measure frequency).

The power chip circuit will also incorporate an analog-to-digital converter. All this data will be fed into the ARM Cortex M0. The Cortex M0 32-bit processor is capable of reaching frequencies up to 50 MHz, has a built in Nested Vector Interrupt Controller (NVIC), a serial wire debug, and system tick timer. It also incorporates digital peripherals and up to forty-two general purpose input-output pins with adjustable pull-up/pull-down resistors. The LPC Xpresso, which is designed for 8- or 16-bit microcontrollers, enhances specific instruction sets and memory addressing. In order for this to be possible, we must run a solid state relay to separate the high AC voltage and lower voltage to DC switch circuit mode. Once the transmitter boards are properly assembled, we will install software that will configure and test the Xbee radios. The Xbee radios will be used as wireless controllers and compose a mesh network that operates over longer ranges.
3.1.2 Software

On the software side of the team, we will optimize the use of simulation and debugging to test our design. The web-server will be simulated to connect to a network, and our Android application will be emulated to view the user interface and validate the data being displayed. We will also deploy a server via Microsoft Visual Studio to simulate our actual web-server while it is still in testing. By doing this, we can verify that our Android application on the tablet and the router is sending and receiving data.

To test our web server’s functionality, we plan to create an isolated network consisting of the MOD5282 (Netburner embedded web server), an access point, and a PC. Using NB Eclipse, the MOD5282 will be flashed with code that will allow it to function as a simple TCP server. MTTTY will be used as a serial debug terminal, enabling us to view the status of our web server. Using Telnet on the PC, the user should be able to connect wirelessly to the web server. Connection is verified when MTTTY displays the IP address of the PC and the corresponding port. Likewise, our server should tell the PC (through Telnet) what IP address it successfully connected to. Once connection is established, the user should be able to send characters to the web server. MTTTY should document and show how many bytes were read as well as the corresponding keystroke.

After the simulations are successful, we must proceed to verify that the connections between the web-server to the router, and the router to the tablet, are being made. The best way to confirm this is to deploy it in real-time and check that the data being displayed in our Android application is what we want to see. A useful tool that we can utilize is Wireshark. Should we have problems with networking between the Zigbee radios all the way to our end device (the tablet), Wireshark will help us capture, browse, and analyze the traffic running on our small network. From there, we will continue the debugging process on all platforms until all devices successfully communicate with each other.
3.2 Benchmarks

The final product will be deemed satisfactorily functional when the performance requirements in Section 2.3 have been met. The tablet should be able to receive and display information acquired from the Cortex M0. Furthermore, appliances should accordingly scale their power consumption based on user input from the tablet or manual overrides.

4 Project Management

4.1 Project Plan

In order to meet the project deadline, the team will be split into two groups: hardware and software. The former will be responsible for managing the power-monitoring component of the project, up to and including the Zigbee network. The software group will be tasked with developing the application that will run on the tablet as well as implementing the web server that will bridge both the Zigbee and WiFi networks.

To assist us in our design, we will require the use of PCs to program the tablet, microcontrollers, and web server. Software, such as Mentor Graphics, will be necessary to design our printed circuit boards. Laboratory equipment, including digital multimeters, oscilloscopes, and power supplies, will be used to test our design. We will also need access to a machine shop to fabricate our display.

4.2 Milestones

See Appendix A for a Gantt chart outlining the major design tasks and their proposed completion dates.
5  Cost Analysis

The proposed project budget is shown below.

![Cost Analysis](image)

*Figure 6: Allocation of funds*
## Appendix A

### Gantt Charts and Milestones

<table>
<thead>
<tr>
<th>ID</th>
<th>Task Name</th>
<th>Duration</th>
<th>Start</th>
<th>Finish</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Microcontroller</td>
<td>40 days</td>
<td>Wed 1/12/11</td>
<td>Tue 2/6/11</td>
</tr>
<tr>
<td>2</td>
<td>Blink LED on LPC</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>3</td>
<td>UART communications on LPC</td>
<td>5 days</td>
<td>Wed 1/19/11</td>
<td>Tue 1/26/11</td>
</tr>
<tr>
<td>4</td>
<td>LCD Interface for Washer/Dryer</td>
<td>10 days</td>
<td>Wed 2/5/11</td>
<td>Tue 2/12/11</td>
</tr>
<tr>
<td>5</td>
<td>Rotary Encoder Interface for Washer/Dryer</td>
<td>10 days</td>
<td>Wed 2/5/11</td>
<td>Tue 2/12/11</td>
</tr>
<tr>
<td>6</td>
<td>Stand-alone Pruning Update with LPC</td>
<td>5 days</td>
<td>Wed 3/11/11</td>
<td>Tue 3/18/11</td>
</tr>
<tr>
<td>7</td>
<td>Establish Communication between Gateway and LPC</td>
<td>4 days</td>
<td>Wed 4/18/11</td>
<td>Mon 5/2/11</td>
</tr>
<tr>
<td>8</td>
<td>Develop Protocol for Communication with Gateway</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>9</td>
<td>Control device on/off using冷链</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>10</td>
<td>Xbee</td>
<td>10 days</td>
<td>Wed 1/26/11</td>
<td>Tue 2/2/11</td>
</tr>
<tr>
<td>11</td>
<td>Communication between two boards</td>
<td>5 days</td>
<td>Wed 1/26/11</td>
<td>Tue 2/2/11</td>
</tr>
<tr>
<td>12</td>
<td>Communication between multiple Xbees</td>
<td>5 days</td>
<td>Wed 1/26/11</td>
<td>Tue 2/2/11</td>
</tr>
<tr>
<td>13</td>
<td>Power Metering IC</td>
<td>20 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/26/11</td>
</tr>
<tr>
<td>14</td>
<td>Test Shunt Resistor and Hall Effect Sensor</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>15</td>
<td>SPI communication on LPC</td>
<td>5 days</td>
<td>Wed 1/23/11</td>
<td>Tue 1/30/11</td>
</tr>
<tr>
<td>16</td>
<td>SPI communication with CS5464</td>
<td>5 days</td>
<td>Wed 2/12/11</td>
<td>Tue 2/19/11</td>
</tr>
<tr>
<td>17</td>
<td>Circuit design for CS5464</td>
<td>5 days</td>
<td>Wed 2/12/11</td>
<td>Tue 2/19/11</td>
</tr>
<tr>
<td>18</td>
<td>Measuring power consumption using CS5464 and PC</td>
<td>5 days</td>
<td>Wed 1/15/11</td>
<td>Tue 1/22/11</td>
</tr>
<tr>
<td>19</td>
<td>Measuring power consumption using CS5464 and LPC</td>
<td>5 days</td>
<td>Wed 1/15/11</td>
<td>Tue 1/22/11</td>
</tr>
<tr>
<td>20</td>
<td>SSR</td>
<td>10 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/26/11</td>
</tr>
<tr>
<td>21</td>
<td>Test SSR</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>22</td>
<td>Function to turn device on/off</td>
<td>5 days</td>
<td>Wed 1/12/11</td>
<td>Tue 1/19/11</td>
</tr>
<tr>
<td>23</td>
<td>Physical Layout</td>
<td>16 days</td>
<td>Wed 1/19/11</td>
<td>Tue 2/5/11</td>
</tr>
<tr>
<td>24</td>
<td>Design Chasis</td>
<td>10 days</td>
<td>Wed 1/26/11</td>
<td>Tue 2/2/11</td>
</tr>
<tr>
<td>25</td>
<td>PCB Layout</td>
<td>10 days</td>
<td>Wed 1/26/11</td>
<td>Tue 2/2/11</td>
</tr>
<tr>
<td>26</td>
<td>Milestones</td>
<td>18 days</td>
<td>Wed 1/26/11</td>
<td>Mon 2/11/11</td>
</tr>
<tr>
<td>27</td>
<td>Move Data from Power Chip Register to Microcontroller</td>
<td>0 days</td>
<td>Wed 1/26/11</td>
<td>Wed 2/2/11</td>
</tr>
<tr>
<td>28</td>
<td>Seed Packet from Microcontroller to Tablet and Vice Versa</td>
<td>0 days</td>
<td>Mon 2/11/11</td>
<td>Mon 2/18/11</td>
</tr>
<tr>
<td>29</td>
<td>Measure Power Using PC Software</td>
<td>0 days</td>
<td>Wed 2/26/11</td>
<td>Wed 3/5/11</td>
</tr>
<tr>
<td>30</td>
<td>Measure Power Consumption of the Device</td>
<td>0 days</td>
<td>Tue 11/6/11</td>
<td>Tue 11/8/11</td>
</tr>
<tr>
<td>31</td>
<td>Deadlines</td>
<td>83 days</td>
<td>Mon 8/29/11</td>
<td>Wed 12/21/11</td>
</tr>
</tbody>
</table>

**Project Name:** SEMS Proposal  
**Date:** Wed 1/12/11
Table 1: Project Milestones and Anticipated Dates of Completion

<table>
<thead>
<tr>
<th>Milestone</th>
<th>Finish By</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tablet Selected</td>
<td>9/7/2011</td>
</tr>
<tr>
<td>Base Station Hardware Selected</td>
<td>9/21/2011</td>
</tr>
<tr>
<td>Simulated WiFi Communication</td>
<td>10/5/2011</td>
</tr>
<tr>
<td>Simulated Real-Time Graphing</td>
<td>10/19/2011</td>
</tr>
<tr>
<td>Measure Power Using PC Software</td>
<td>10/26/2011</td>
</tr>
<tr>
<td>Move Data from Power Chip Register to Microcontroller Memory</td>
<td>11/2/2011</td>
</tr>
<tr>
<td>Tablet User Experience Done; Interacting with Base Station</td>
<td>11/2/2011</td>
</tr>
<tr>
<td>Graphing Real-Time Power Data from Real Hardware</td>
<td>11/16/2011</td>
</tr>
<tr>
<td>Send Packet from Microcontroller to Tablet and Vice Versa</td>
<td>11/21/2011</td>
</tr>
<tr>
<td>System Complete; Only Small Tweaks Required</td>
<td>11/30/2011</td>
</tr>
<tr>
<td>Final Website Online</td>
<td>12/8/2011</td>
</tr>
<tr>
<td>Design Day</td>
<td>12/9/2011</td>
</tr>
<tr>
<td>Final Report Due</td>
<td>12/21/2011</td>
</tr>
</tbody>
</table>
Team Members
Software Team:
Chris Skalka
James Smith
Jhen Almeda
Jodi Quilalang
Wahid Baha

Hardware Team:
Bob Severson
Jose Cardenas
Kyle Dickerson
Marizol Cruz
Saul Lima
Wade Dorough

Team Leaders:
Ben Lauer
Sherwin Yari

Project Description:
The vision of SEMS is to create a smart home energy management system. SEMS will allow home owners to monitor their energy consumption through a network of sensors that monitor home appliances. They will be able to view all the information utilizing a user-friendly application for an Android tablet. The end result is consumers saving money and energy.

Project Requirements:
• Construct a network of sensors that will monitor home appliances
• Create a central communication gateway that will allow the sensors to communicate with the GUI application
• Develop a user-friendly GUI application