Residence Energy Control System Based on Wireless Smart Socket and IoT

KUN-LIN TSAI1, FANG-YIE LEU2, (Member, IEEE), AND ILSUN YOU3, (Senior Member, IEEE)
1Department of Electrical Engineering, Tunghai University, Taichung 407, Taiwan
2Department of Computer Science, Tunghai University, Taichung 407, Taiwan
3Department of Information Security Engineering, Soonchunhyang University, Asan 31538, South Korea
Corresponding author: I. You (ilsunu@gmail.com)

This work was supported in part by the Soonchunhyang University Research Fund and in part by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2014R1A1A1005915).

ABSTRACT To avoid resources on green earth being exhausted much earlier by human beings, energy saving has been one of the key issues in our everyday lives. In fact, energy control for some appliances is an effective method to save energy at home, since it prevents users from consuming too much energy. Even though there are numerous commercial energy-effective products that are helpful in energy saving for particular appliances, it is still hard to find a comprehensive solution to effectively reduce appliances’ energy consumption in a house. Therefore, in this paper, an intelligent energy control scheme, named the residence energy control system (RECoS) is proposed, which is developed based on wireless smart socket and Internet of Things technology to minimize energy consumption of home appliances without deploying sensors. The RECoS provides four control modes, including peak-time control, energy-limit control, automatic control, and user control. The former two are operated for all smart sockets in a house, while the latter two are used by individual smart sockets, aiming to enhance the functionality of energy control. The experimental results show that the proposed scheme can save up to 43.4% of energy for some appliances in one weekday.

INDEX TERMS Energy control system, Internet of Things (IoT), neural network, smart socket, smart living.

I. INTRODUCTION
In the past decade, due to greenhouse effect [1], energy saving has been one of the critical issues in designing electronic appliances. Smart-houses, which are houses equipped with highly advanced automatic lighting systems, temperature control system, security control mechanisms and many other functions, can be seen everywhere in the world. The purpose of constructing these systems and functions is amenity and energy efficiency. In fact, to save energy, a residence management system with intelligent and automatic energy control policies is required and essential [2]. On the other hand, a smart-house developed on the basis of the Internet of Things (IoT) can save more energy, where IoT is a network system consisting of electronic devices, software, sensors and networks that connect all concerned network entities together to make the system more valuable and able to provide many more services to users.

Up to present, many energy control methods have been proposed. By utilizing IoT, [3] developed a tablet-computer based Home Energy Management scheme to monitor the consumption of home energy. With this scheme, users can set up management policies to control home energy consumption based on the time of a day. Kopytoff and Kim [4] showed a power meter which provides real-time information about home energy consumption to users. The main goal is raising consumers’ energy consumption awareness, potentially inspiring them to be more energy efficient.

Most of the electronic appliance control applications utilize a lot of sensors to sense users’ locations and activities. Some of them even use Open Service Gateway initiative (OSGi) [5] or Service-Oriented Architecture (SOA) [6] to predict users’ behaviors in a house, with which to manage the home appliances in this house. Some previous studies [7], [8] improved functions of sockets set up in a house and connected with wireless networks to control home appliances. Sensing users’ location, motion, and habits with a large number of sensors may not be an energy efficient method since these sensors consume considerable resources. So it would be better for them to be low cost and high coverage. Also, according to the survey on developed countries by the International...
Energy Agency, the energy consumed by idle appliances, called standby energy, in a house is about 3% to 11% of total energy consumed by the house [9]. Basically, house energy can be further reduced if standby energy is effectively lowered without significantly affecting users’ everyday lives, implying that the conflict between house energy saving and user’s living convenience need to be balanced.

Therefore, in this paper, an intelligent energy saving scheme, named the Residence Energy Control System (RECoS for short), is proposed to reduce the energy consumption of home appliances without deploying sensors. The RECoS, based on wireless smart sockets and IoT technology, not only monitors/controls the standby power consumption of an individual appliance, but also manages energy consumed by all controllable appliances. The RECoS also invokes the neural network algorithm to study user’s lifestyle and automatically turns off the power of each smart socket connected to IoT when the electric appliances are not in use. The experiments demonstrate that the RECoS can save up to 43% of energy for some appliances in a weekday.

The rest of this paper is organized as follows. Section 2 briefly introduces the related work of this study. Section 3 describes the RECoS architecture. Energy control policies of the proposed scheme are presented in Section 4. Experiment results are shown and discussed in Section 5. Section 6 concludes this paper and outlines future research.

II. RELATED STUDIES

Nowadays, many related studies of home energy control have been proposed. Mohsenian-Rad et al. [10] introduced a game-based approach for optimizing energy consumed by a residential building. But they did not consider users’ satisfaction degree for their efficient task scheduling. Optimal scheduling of in-home appliances with storage devices has been discussed in [11], in which the total cost minimization is one of the objectives of its optimization attempt. Basically, these two techniques were developed mainly based on deterministic and/or meta-heuristic methods. But they failed to consider users’ convenience and comfort levels for their cost optimization process. Anvari-Moghadam et al. [12] developed an integer nonlinear programming model for optimal energy use in a smart home by considering a meaningful balance between energy saving and a comfortable lifestyle. Through incorporation of a mixed objective function under different system constraints and user preferences, the algorithm presented in [12] reduced the domestic energy usage and utility bills, and ensured an optimal task scheduling and a thermal comfort zone for its inhabitants. However, if IoT techniques can be applied to this model, the energy can be further reduced.

The developments of the IoT and wireless sensor networks come up with new solutions for residence management. In such a home management system, a fix IP address is required, and remote users need a high-speed connection to access the system. Yeoh et al. [13] established an e2Home association which enables remote users to manage smart home appliances, and uses emails as the communication medium. The advantage is that a user does not have to establish a high speed Internet connection before he/she can effectively manage home appliances. However, the complex email services result in the fact that the system is a little hard to be constructed. Das et al. [14] published an adaptive versatile home architecture which creates a rational agent as a home servant to seek for a method that can maximize inhabitant comfort and minimize operation cost for users. Choi et al. [15] developed a context-aware middleware that provides users with an automatic home service inside a smart home following the users’ preference. This middleware uses open service gateway as the framework of the home network, and employs sensed data to predict the users’ preference for home appliances. This sensed data includes pulse, body temperature, facial expression, room temperature, time and location.

On the other hand, Robles et al. [16] proposed a smart water management model which integrates IoT technologies with business process coordination and decision support systems. Wang et al. [17] demonstrated a smart home control system consisting of an embedded controller, signal converters and terminal devices. With this system, users can control multiple systems via a smart phone. Then the synchronizer of this system will send user commands to all systems. Suh and Ko [18] introduced an active sensor network to sense user activities and control the on/off state of home appliances.

Although these IoT systems and technologies are relatively novel, there are still many untapped applications areas that need to be developed, and numerous technical challenges and issues that can be further improved and broadly explored [19]. The area of home energy control by using IoT has also been widely proposed. Aram et al. [20] contributed to the energy conservation approaches by reducing the amount of required communication. The method predicts the amount of sensed data by using non-linear autoregressive neural networks. Its performance is evaluated by using data obtained from temperature and humidity sensors under different conditions, indicating that the method indeed substantially reduces power consumption for wireless sensor networks. Lee et al. [7] designed an intelligent power management device which adopts user’s locations, motion detection, and living patterns as its parameters to reduce the energy consumed by some appliances, like lights and humidifier. As a sensor-based system, it could achieve 7.5% of power saving.

Also, Han and Lim [21] introduced a home energy control system developed based on IEEE 802.15.4 and ZigBee to provide users with intelligent services to enrich their lives. This system integrates diversified physical sensing information and controls various home devices to assign various home network tasks to appropriate components. Besides, the authors also presented a disjoint multipath-based routing protocol to improve the system performance. Lien et al. [22] proposed a wireless power-controlled outlet module which manages home power with a scalable mechanism and integrates multiple AC power sockets and a microcontroller as a
part of a power outlet to turn on/off the power of the sockets. Park et al. [8] developed a control scheme to minimize the power consumed by a home gateway by employing a sleep and wake-up mechanism, which changes its mode depending on whether or not any user service traffic is now being delivered or any embedded services are serving users through the home gateway. In [23], the deployment of a common client/server architecture focusing on monitoring energy consumption is described, but no control action is mentioned.

Most previous home control systems gathered information from various sensors, such as temperature sensor, distance sensor, position finder, passive infrared sensor, and ambient light sensor to sense the users’ related data, even lifestyle. Although they can have higher accuracy in predicting users’ behaviors, the cost and power consumption of a huge number of employed sensors are big problems needed to be solved.

III. THE RECoS ARCHITECTURE

A. SYSTEM OVERVIEW

In the RECoS, no sensor is deployed, and the information of appliances’ energy consumption is collected by smart sockets through IoT. Those home appliances regularly or periodically staying in their standby states will be turned off by switching off the corresponding power supply embedded in their smart sockets with an electronic approach. After receiving user-defined energy limit for a smart socket, the RECoS gives a one-day energy quota to the smart socket, and accordingly controls the energy consumed by those appliances connected to the socket.

FIGURE 1. The IoT-based system architecture of the RECoS.

Fig. 1 shows the system architecture of the RECoS, in which the smart sockets measure connected devices’ current electricity data, including voltage, current, power, etc., which are acquired by home gateway and then sent to the energy controller. On receiving a “turn-on” or “turn-off” command issued by the energy controller, home gateway transmits it to the target smart socket which will accordingly turn on/off its power supply. Energy controller implemented in a cloud server, besides storing electricity data, also determines the state of a socket, communicates with users, manages the energy consumption of a house and so on. Furthermore, users can set energy limit, and control smart sockets manually.

The wireless communication protocol between smart sockets and home gateway is ZigBee which consumes very low power and is often employed by personal area networks. The communication protocols between the cloud server and home gateway (and users) can be 3G, 4G or Internet (and WiFi), since the data transmitted between them is often large and long-distant.

In general, IoT has three layers, i.e., sensor layer, network layer, and application layer. From the IoT viewpoint, the sensor layer in the architecture of the RECoS is the smart sockets, and the network layer is ZigBee, 3G, 4G, etc., whereas the application layer is the smart grid and energy management mechanism.

B. SYSTEM CONTROL MODES

The RECoS has four control modes, including peak-time control (PTC), energy-limit control (ELC), automatic control (AC) and user control (UC). The PTC and ELC modes are exclusively used to control all smart sockets in a house, meaning at any moment, only one of the PTC mode and ELC mode can be utilized. In the PTC mode, the RECoS controls each socket’s on/off state according to the user-defined peak-time and its peak-time energy limit. When the total energy consumption of the underlying house exceeds the energy limit, low priority sockets will be turned off to reduce the peak power consumption, and a warning message will be sent to the users. On the other hand, after the user sets the energy limit of a house for the following 4 weeks, the ELC mode will be triggered. Then the RECoS calculates one-day quota for each socket according to the energy consumption history of all sockets. Once the energy consumption of a certain smart socket in a specific weekday or weekend exceeds its quota, the RECoS follows the ELC-mode policy to turn off its power supply.

The AC and UC modes are applied to control individual smart sockets following automatic control policy and user-defined control policy, respectively. Like the exclusion between PTC and ELC modes, at any instance, either AC or UC can be used to control a smart socket. The RECoS in its AC mode determines the on/off state of a smart socket based on the socket’s energy usage history in the last 4 weeks. The RECoS in its UC mode provides an interface with which users can flexibly set up the on/off state of a certain socket. Note that the AC/UC mode of a smart socket can co-exist with the PTC/ELC mode of the energy controller. For example, the energy controller of a house is now in its PTC mode, and a smart socket in the house is in its UC mode or AC mode. Details of AC and UC modes will be described later.
C. THE ENERGY CONTROLLER

The energy controller (i.e., the cloud server) of the RECoS, as shown in Fig. 2, consists of Memory, action trigger and seven modules, including learning module, user interface module, information/command transceiver module, automatic control (AC) module, user control (UC) module, energy-limit control (ELC) module, and peak-time control (PTC) module.

The Memory, consisting of a Dynamic Random Access Memory (DRAM) and a hard disk, is used to store smart sockets’ data, output of the learning module, user-defined energy limit, user commands, etc. Old data is stored in the hard disk and retrieved to DRAM when necessary. Action trigger is an internal system clock utilized to send trigger signals to the learning module, information/command transceiver module and four control modules (including the AC module, UC module, ELC module, and PTC module) to trigger further activities. The frequency of sending a trigger signal to each of the 4 control modules is once per 10 minutes throughout a day, i.e., at 00:10, 00:20, 00:30, and so on. The purposes and functions of the seven modules will be described below.

1) LEARNING MODULE
The learning module, as the kernel of energy controller, is implemented by using the neural network algorithm for learning and calculating a smart sockets’ operation time. The action trigger sends trigger signals to the learning module at 12:30 and 00:30. On receiving the first trigger signal (i.e. at 12:30), the learning module requests energy consumption histories of all smart sockets, and the peak-time periods set by users for the sockets as its input to calculate the time periods in which a smart socket will be turned on or off for the time duration from 00:00 to 12:00 (i.e., AM) of the next day. The time periods as the output of this module are stored in the Memory for easily being accessed by other modules. When the second trigger signal arrives, of course, at 00:30, the inputs and outputs are similar to those when receiving the first trigger signal. But this time, the calculation is for the time duration from 12:00 PM to 00:00 of current day.

2) USER INTERFACE MODULE
This module provides interfaces for users to check the energy/power consumption, accumulated energy consumption and the on/off state of a socket, and accumulative energy consumed by a house. User can also check the energy consumption history of a socket via this display module. Besides, when receiving user’s commands, which may be energy limit, peak-time setting, a socket’s priority, AC/UC mode switching, etc., this module sends a data request or control signal to the corresponding control module or stores data into the Memory. For example, if user sets the energy limit via user command processing module, then the value will be sent to ELC module to control the total energy consumption of smart sockets.

3) INFORMATION/COMMAND TRANSCEIVER MODULE
This module establishes a connection between itself and the home gateway through the Internet or a 3G/4G network so that smart sockets’ data can be sent to cloud server, and system’s/user’s commands can be transmitted to the home gateway. As mentioned before, the action trigger activates this module every 10 minutes. Once activated, the module sends a data request to home gateway. On receiving the reply, the module stores the received data with the smart socket’s ID in the Memory.

4) AUTOMATIC CONTROL (AC) MODULE
If the RECoS is now in its AC mode, according to output of the learning module, the AC module turns on/off the power of smart sockets for particular time periods. Besides, when total energy consumption of a smart socket exceeds user-defined limit, the AC module will receive requests issued by PTC/ELC module and then accordingly turn off those low-priority smart sockets.

5) USER CONTROL (UC) MODULE
The UC module provides friendly control functions, with which users can input control commands via user devices (e.g., smart phone, PC, etc.), no matter whether the RECoS is now in its AC or UC mode. Through user interface module, the commands will be sent to UC module to set particular socket into always-on/always-off state. Besides, the UC module also controls a particular smart socket’s energy consumption following user-defined energy limit.

6) ENERGY-LIMIT CONTROL (ELC) MODULE
When a user sets a 4-week energy limit, a socket’s one-day quota actually is calculated by the ELC module. If the energy controller is now in its ELC mode, the ELC module will control a socket’s one-day energy limit based on the fact that today is a weekday or weekend. When the energy consumption of a day exceeds the quota, if the socket is now in its AC mode, the ELC module sends a control signal to AC module to turn off the corresponding socket. But if the socket is now in its UC mode, the ELC module will send a warning signal to UC module to alert the user.
7) PEAK-TIME CONTROL (PTC) MODULE
On receiving the user-defined peak-time periods and a socket’s priority (also defined by user), the PTC module sends signals to AC and UC modules to control the on/off state of a socket. As mentioned above, when the total energy consumption of a house exceeds its peak-time energy limit, lower priority sockets will be turned off.

IV. ENERGY CONTROL POLICY
A. INFORMATION GATHERING AND SMART LEARNING
In the RECoS, a smart socket automatically measures the electricity data of an appliance, including voltage, current, power, accumulated energy, phase, frequency, etc. In the following experiment, the measurement frequency is 10 times per second. After each measurement, if the data is sent to home gateway, for each time period, i.e., 10 minutes (controlled by action trigger), a total of 6000 (=10 times/second * 60 second/minute * 10 minutes) times of data delivery will be performed by a smart socket. However, such a large number of data may congest the wireless network, i.e., ZigBee. Thus, for each time period, after measurement, the smart socket only transmits the average voltage, average current, average power consumption, and accumulated energy consumption (=average power * time) to home gateway. Namely, a smart socket sends data to home gateway 144 (= 6 times/hour * 24 hours) times per day. After gathering the electricity data sent by smart sockets, home gateway transmits the data to the energy controller via the Internet or a mobile communication network. The energy controller collects the data and then stores it in the statistical table previously created in the cloud server.

| TABLE 1. The statistical table recording the total energy consumption of a house in the last 28 days.

<table>
<thead>
<tr>
<th>Time Period</th>
<th>Energy Consumption (0.00-0.10)</th>
<th>Energy Consumption (0.10-0.20)</th>
<th>Energy Consumption (0.20-0.30)</th>
<th>Energy Consumption (0.30-0.40)</th>
<th>Daily Accumulated Energy (0.00-0.06)</th>
<th>Daily Accumulated Energy (0.06-0.12)</th>
<th>Daily Accumulated Energy (0.12-0.18)</th>
<th>Daily Accumulated Energy (0.18-0.24)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The statistical table keeps the energy consumed by all appliances connected to an individual socket and the total energy consumed by all appliances connected to all sockets of a house. An example of the latter is shown in Table I. $TP_i$ field, $1 \leq i \leq 144$, records the time period in which energy consumed by appliances is measured. $E_{\text{predict},i}$ shows the predicted energy consumption for $TP_i$ by using proposed learning module, and $E_{\text{actual},i}$ indicates the actual energy consumed in $TP_i$. Daily accumulated energy field holds the energy accumulatively consumed in a day. The dates of the last 4 weeks from today $D_i$ are denoted from $D_{i-28}$ to $D_{i-1}$. That means the statistical table only records data for the last 28 days, and the history data is stored in the hard disk of the cloud server. Taking Table I as an example, assuming that today is 8/29, i.e., $D_1$, and its $E_{\text{predict},1}$, denoted by $E_{\text{predict},1}^D_1$, is 0.51 kWh, while its actual energy consumption, denoted by $E_{\text{actual},1}^D$, is 0.52 kWh. NA means the data is unavailable since the time period has not reached. Up to the last time period, the daily accumulated energy on 8/29 is 17.59 kWh, $E_{\text{predict},j}^D$, is calculated according to all $E_{\text{actual},i}$ of the same time period in the same weekday or the weekend in the last 4 weeks, i.e., $E_{\text{predict},7}^D$, $E_{\text{predict},14}^D$, $E_{\text{predict},21}^D$, and $E_{\text{predict},28}^D$.

### B. SMART LEARNING
To predict energy consumption for a family’s everyday life, the RECoS invokes the backpropagation neural network (BPNN) algorithm to establish an energy usage model for the family given the energy consumption history of each smart socket used by this family. The BPNN algorithm has three layers, including input layer, hidden layer, and output layer [24]. In the RECoS, as mentioned previously, the inputs of the BPNN algorithm are the home appliance’s actual energy consumption in the same time periods, e.g., $TP_i$, of a specific weekday or weekend of the past 4 weeks, including $D_{i-7}$, $D_{i-14}$, $D_{i-21}$, $D_{i-28}$ if today is $D_i$. The energies consumed by a weekday and a weekend are independent so that the energy prediction accuracy is higher. The output of the BPNN algorithm is the predicted energy consumption $E_{\text{predict},i}^D$.

In the learning phase, the operation process of the BPNN algorithm includes the learning phase and recalling phase. The supervised learning [24], which is the machine learning task for inferring a model from labeled training data, is invoked. The training data consists of a set of training examples, and each example is a pair of data comprising an input and a desired output when the input is given. After an inferred model is generated, the model will map a new input
to its corresponding output. The input and output then form a new example. The gradient steepest descent method [25] is employed to adjust the weight and bias of learning parameters. The steps of the learning phase are as follows.

1. Determine the numbers of network layers and neurons for each layer.
2. Randomly set the initial values to weights and biases of neurons.
3. Input training examples, including their inputs and the desired outputs.
4. Calculate the inferred outputs via inferred function for inputs.
5. Calculate the error rates between output layer’s neurons and hidden layer’s neurons.
6. Calculate the correction amount of weights and biases for each layer.
7. Update the weights and biases for each layer.
8. Repeat step (3) to step (7) until the BPNN converges or the number of training cycles reaches user-defined upper limit.

The recalling phase has four steps.
1. Read the trained weights and biases.
2. Read a test example.
3. Infer the output value.
4. Repeat steps (2) to (4) until no more test examples are inputted.

C. AUTOMATIC CONTROL AND USER CONTROL

An example showing how the RECoS on AC mode works is that the members of a family very often get up at 7:00 and go to work/school at 8:00. During this period of time, they use microwave oven, water dispenser, oven, and electric boiler. After 17:00, they come home, watch TV, turn on air conditioner, use microwave oven and so on. After smart learning, the RECoS turns off the power supply of unused appliances, such as microwave oven, water dispenser, and electric boiler at 8:10, and turns on the power supply of those appliances, that may be used, at 16:50. The UC mode controls the operation time of a socket following user’s settings.

The control flow of an individual smart socket on AC and UC modes is shown in Fig. 3. Each time when action trigger is activated be a timer (recall, every 10 minutes, e.g., \( T_P \)), the energy controller sends an energy-usage-request to home gateway. The home gateway will reply the electricity data (e.g., voltage, current, power, accumulated energy, etc.) of smart sockets in the corresponding time period. When receiving the data, the energy controller stores the data in the Memory. Also, current time is used to verify whether it is the learning time (00:30 and 12:30) or not. If yes, the energy controller invokes the smart learning algorithm and saves the results in the Memory for smart socket control. But no matter whether the answer is yes or no, the energy controller firstly checks the control mode for each socket, e.g., SS.

If SS is currently in its UC mode, and the energy controller keeps monitoring the socket and waits for the next trigger. However, if the quota has been reached, the energy controller sends a warning message to user and asks the user to change the socket’s control mode to AC mode. If the user agrees, after the change, the socket is then automatically controlled by the energy controller, and waits for the next trigger. However, if the user disagrees, SS stays in its UC mode; the energy controller keeps monitoring the smart socket’s energy consumption; no warning messages will be sent to user in \( T_P \); the energy controller will not ask the user to change the control mode again.

Generally, the user’s lifestyle in a weekday and in a weekend is different. For example, many home appliances are in their standby in the daytime of a weekday; however, they may be in use if the user stay at home in a weekend. So when the smart socket lies in the AC mode, or at any moment, when the user sets the control mode of SS to AC mode, as shown in Fig. 3, the energy controller will check SS’s energy consumption status. If a socket staying in the AC mode in a weekday exceeds its one-day energy quota, and currently it is not in use, the RECoS following the weekday energy control rule will immediately turn it off, regardless of what the predicting result of smart learning (i.e., \( E_{Predict,TP} \)) is. If the day is a weekend, and the socket’s accumulated energy is lower than its energy limit, the RECoS verifies whether the socket is in use or not. If yes, the energy controller keeps monitoring the energy consumption of that smart socket and waits for next trigger. Otherwise, it reads the results of smart learning and accordingly turns off the smart socket.

In the RECoS, the UC mode has a higher priority than that of AC mode. That is why the user can turn on/off a smart socket anytime. Moreover, he/she can set the operation time of a smart socket. The purpose is to provide users...
TABLE 2. An example of energy-limit control table (unit: kWh).

<table>
<thead>
<tr>
<th>Smart socket ID</th>
<th>Smart Socket 1</th>
<th>Smart Socket 2</th>
<th>Smart Socket 3</th>
<th>...</th>
<th>Smart Socket N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Weekday average</td>
<td>2.24</td>
<td>0.88</td>
<td>1.56</td>
<td>...</td>
<td>0.41</td>
</tr>
<tr>
<td>Weekend average</td>
<td>3.79</td>
<td>1.25</td>
<td>1.99</td>
<td>...</td>
<td>0.92</td>
</tr>
<tr>
<td>Energy limit</td>
<td>70.61</td>
<td>25.94</td>
<td>44.29</td>
<td>...</td>
<td>14.63</td>
</tr>
<tr>
<td>Weekend quota</td>
<td>2.11</td>
<td>0.83</td>
<td>1.47</td>
<td>...</td>
<td>0.39</td>
</tr>
<tr>
<td>Weekend quota</td>
<td>3.56</td>
<td>1.18</td>
<td>1.87</td>
<td>...</td>
<td>0.86</td>
</tr>
</tbody>
</table>

Accumulated energy consumption of the last 4 weeks: 404.36
User-defined total energy limit: 380

with high usage flexibility and user-friendliness operation environment.

D. PEAK-TIME CONTROL AND ENERGY-LIMIT CONTROL
The main difference between the PTC and ELC modes is the duration of monitoring time. The PTC manages the total energy/power consumption at the peak-load, while the ELC guarantees that the energy consumption of a house is not over user-defined limit in the following 4 weeks. In the PTC mode, when the total energy/power consumption of a house reaches the one-day quota, the energy controller sends a warning message to the user and turns off the power of those low-priority smart sockets currently lied in AC mode. Those now in their UC mode will not be turned off. But the energy controller keeps monitoring their energy/power consumption. This means that even though the one-day quota of a smart socket has been reached, users can set the socket to its UC mode and continuously use the corresponding appliances.

The RECoS in its ELC mode controls a smart socket, e.g., $Q$, with a given quota, e.g., $q(Q)$. When users set $Q$’s energy limit of the following 4 weeks (after referring to $Q$’s energy consumption in the last 4 weeks), the ELC module calculates the energy quota for $Q$ with the parameters, including $Q$’s one-day, e.g. day X’s, energy consumption history, what day is X (a weekday or weekend), the $Q$’s priority, etc. Table II shows an example of the energy-limit control table, in which the field of the “accumulated energy consumed of the last 4 weeks” is the actual amount of accumulated energy consumption in the last 4 weeks, and the field of the “user-defined total energy limit” is the expected energy consumption of the following 28 days which is 94% (≈380/404.36) of accumulated energy in the last 4 weeks. The rows of weekday average and weekend average are, respectively, the average energy consumption in a weekday and in Saturday or Sunday during the past 4 weeks. Energy limit which is defined by user based on the energy consumption history represents the energy limit of the following 4 weeks of a smart socket, and weekday/weekend quota is the socket’s one-day energy quota. Taking smart_socket_1 as an example, its weekday/weekend energy usage is 2.24 kWh/3.79 kWh in average, totally consuming 75.12 (≈2.24+3.79=8) kWh in the last 28 days. Now, when the user would like to save 6% of energy consumption, smart_socket_1 during the following 4 weeks has an energy limit of 70.61 (≈75.12*94%) kWh, which is shared by the following 4 weekends (8 days) and 20 weekdays. The quota of a weekday is 2.10 (≈2.24*94%) kWh, whereas the quota of a weekend is 3.56 (≈3.79*94%) kWh.

E. ENERGY MODEL OF A SOCKET SET
The energy consumption of a socket set consisting of a smart socket and an appliance in a day can be calculated with Eq. (2).

$$E_{socket-set} = E_{appliance} + (144 + r)E_{comm} + E_{circuit} \quad (2)$$

where $E_{appliance}$, $E_{comm}$, and $E_{circuit}$ are, respectively, the energy consumption of an appliance, $k$ bits ZigBee communication, and the circuit of a smart socket. A total of $144+r$ ZigBee messages are sent from the smart socket to home gateway in a day, where 144 is the number of time periods in a day and $r$ is the times of the aperiodic control command sent by home gateway to control the state of the smart socket also in a day.

$E_{appliance}$ as shown in Eq. (3) includes three parts, i.e., turn-on, standby, and power-off.

$$E_{appliance} = \sum_{\text{turnon periods}} V_s \cdot \int_{0}^{t_1} I_{\text{turnon}} \cdot \sum_{\text{standby periods}} V_s \cdot \int_{0}^{t_2} I_{\text{standby}} \cdot \sum_{\text{poweroff periods}} V_s \cdot \int_{0}^{t_3} I_{\text{poweroff}}$$

where $V_s$ indicates the supply voltage, $I_{\text{turnon}}$, $I_{\text{standby}}$, and $I_{\text{poweroff}}$ represent the currents consumed by the appliance in its turn-on state, standby state, and power-off state, respectively, and $t_1$, $t_2$, and $t_3$ are the time periods in which the appliance stays in its turn-on state, standby state, and power-off state, respectively. However, $I_{\text{poweroff}}$ is 0. The third part on the RHS of Eq. (3) is then 0.

$E_{circuit}$ as shown in Eq. (4) is defined as

$$E_{circuit} = \int_{0}^{t_4} (\alpha \cdot V_{dd}^2 \cdot C \cdot f + V_{dd} \cdot I_{\text{static}})$$

in which $\alpha$ is the on/off switching activity of transistor and the value is about 0.5 in general, $V_{dd}$ indicates the supply voltage of the circuit, $C$ is the total capacitance, $f$ represents...
TABLE 3. Appliances' energy consumption by using/without using the RECoS.

<table>
<thead>
<tr>
<th>Home appliance</th>
<th>Location</th>
<th>Priority</th>
<th>date</th>
<th>Energy consumption without using RECoS (kWh)</th>
<th>Energy consumption using RECoS (kWh)</th>
<th>Energy saving</th>
</tr>
</thead>
<tbody>
<tr>
<td>Water dispenser</td>
<td>Office</td>
<td>-</td>
<td>Weekday</td>
<td>3.062</td>
<td>1.939</td>
<td>36.7%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>1.858</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>Photocopier</td>
<td>Office</td>
<td>-</td>
<td>Weekday</td>
<td>5.920</td>
<td>3.970</td>
<td>32.9%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>3.120</td>
<td>0</td>
<td>100%</td>
</tr>
<tr>
<td>Water dispenser</td>
<td>House</td>
<td>2</td>
<td>Weekday</td>
<td>1.954</td>
<td>1.105</td>
<td>43.4%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>2.452</td>
<td>1.883</td>
<td>23.2%</td>
</tr>
<tr>
<td>Microwave oven</td>
<td>House</td>
<td>1</td>
<td>Weekday</td>
<td>0.716</td>
<td>0.626</td>
<td>12.6%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>1.961</td>
<td>1.875</td>
<td>4.4%</td>
</tr>
<tr>
<td>Electric boiler</td>
<td>House</td>
<td>1</td>
<td>Weekday</td>
<td>6.888</td>
<td>4.102</td>
<td>40.4%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>6.901</td>
<td>4.137</td>
<td>40.1%</td>
</tr>
<tr>
<td>Stereo</td>
<td>House</td>
<td>3</td>
<td>Weekday</td>
<td>0.233</td>
<td>0.147</td>
<td>36.9%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>0.325</td>
<td>0.231</td>
<td>28.9%</td>
</tr>
<tr>
<td>Washing machine</td>
<td>House</td>
<td>1</td>
<td>Weekday</td>
<td>0.524</td>
<td>0.428</td>
<td>18.3%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Weekend</td>
<td>0.839</td>
<td>0.746</td>
<td>11.1%</td>
</tr>
</tbody>
</table>

the clock frequency, $I_{\text{static}}$ stands for static current when the circuit is idle, and $t_4$ is the smart socket operation time. In fact, $\alpha \cdot V_{\text{dd}}^2 \cdot C \cdot f$ is the circuit’s dynamic power consumption, i.e., the power consumed by charging/discharging capacitors [26] when the transistors change their on/off state, while $V_{\text{dd}} \cdot I_{\text{static}}$ represents static power consumption, i.e., the power consumed due to current leakage [26].

ZigBee communication energy can be found in [27], and the energy required for transmitting ($E_T$) and receiving ($E_R$) $k$ bits of data are formulated in Eq. (5).

$$E_{\text{comm}} = E_T + E_R = \left( E_{\text{elec}} \times k + \epsilon_{\text{amp}} \times k \times d^2 \right) + \left( E_{\text{elec}} \times k \right) = 2E_{\text{elec}} \times k + \epsilon_{\text{amp}} \times k \times d^2$$

in which the radio electronics parameter $E_{\text{elec}}$ is about 50 nJ/bit, i.e., the energy consumed by antenna, the transmission amplifier parameter $\epsilon_{\text{amp}}$ is about 10 nJ/bit/m$^2$ [28] and $d$ is the transmission distance.

V. EXPERIMENTAL RESULTS AND DISCUSSIONS

A. IMPLEMENTATION ENVIRONMENT AND TOOL
To verify the feasibility and effectiveness of the RECoS, three major components are implemented and integrated, including smart socket, home gateway, and energy controller. The smart socket, as shown in Fig. 4a, was developed by Industrial Technology Research Institute of Taiwan, and equipped with a digital power meter. The voltage provided is between 50V and 350V, and current supplied is between 10 mA and 15A. Its power consumption is less than 3W. Besides, the smart socket has a ZigBee communication circuit. Home gateway, as shown in Fig. 4b, was implemented with 400 MHz operation frequency. The platform is also equipped with 64MB SDRAM, ZigBee module, 100Mbps Ethernet interface, and USB I/O interface. The energy controller was developed on a server with Intel i5-2300 2.8GHz processor, 16GB RAM, 1TB hard disk, and Linux 3.8.13 operating system. The ZigBee with bandwidth of 205Kbps is used for communication between smart sockets and home gateway.

The communication channel between home gateway and energy controller is 100 Mbps Ethernet. The user interface, as shown in Fig. 4c, is implemented on a smartphone.

B. EXPERIMENTAL RESULTS
Table III shows the home appliances’ energy consumption by using and without using the RECoS. Two appliances in an office and five appliances in a house are tested individually in a weekday and a weekend. All these appliances are controlled by the AC mode, and the priorities of appliances in the house are given 1 (the highest) to 3 (the lowest). In Table III, the water dispenser and photocopier in the office are turned off during weekends since no one uses them, and the standby energy saving can achieve 100%. For home appliances in the house, the water dispenser saves up to 43.4% of energy consumption in a weekday.

Fig. 5 further illustrates the energy consumption of the water dispenser (in office) during a weekday. The solid red line indicates the accumulated energy consumption without using the RECoS, whereas the dotted blue line is that when the RECoS is in use. Before 7:30 and after 18:30, the dotted blue line (labeled with letter A) is 0, because the proposed learning module knows that the water dispenser is only used between 8:00 and 18:00, and it turns off the water dispenser after 18:30 and turns it on at 7:30. After the energy controller turns on the water dispenser, it needs some energy (labeled with letter B) to boil cool water. When the RECoS is not in use, the water dispenser needs to keep water warm/hot all day long. Once the temperature goes down to a defined
threshold, it will boil the water again (labeled with letter C). Also, the peak difference in Fig. 5 is due to different amounts of water contained in it. The total energy consumption of the solid red line is 3.062 kWh, and that of the dotted blue line is 1.939 kWh. The energy saving is about 36.7% (= (3.062 - 1.939)/3.062).

VI. CONCLUSION AND FUTURE STUDIES
One of the main purposes of constructing a smart house is to automatically control those appliances in the house to achieve the goals of energy saving and smart living. In this paper, the RECoS controls the energy consumption in a residence through IoT and smart sockets. The RECoS provides four control modes to control the on/off state of home appliances connected to smart sockets. A simple IoT structure which integrates smart sockets, home gateway, energy controller, ZigBee, and Internet is proposed. Most importantly, the RECoS is sensorless and can be applied to outdated appliances, i.e., those without providing network connections. By using the neural network algorithm for smart learning, the RECoS can save unnecessary energy consumed by a house, and the experimental results show that up to 43.4% of energy can be reduced for a water dispenser in a weekday. Other appliances can also save some amounts of energy.

In the near future, the reliability and behaviour models would be derived for the RECoS so that the users can, respectively, predict its reliability and behaviours before using it. Besides, a simple user interface and personalized learning model will also be developed so that the RECoS can reduce the energy consumption more intelligently. Furthermore, security is an important issue in safely protecting the RECoS, e.g., encrypting the control commands sent to smart sockets [31], [32] to avoid hackers turning on/off the sockets that need to be turned off/on. These constitute the future studies.

REFERENCES


KUN-LIN TSAI received the B.S. degree in computer and information science from Tunghai University, Taichung, Taiwan, in 1999, the M.S. degree in computer science and information engineering from National Taiwan University, Taipei, in 2001, and the Ph.D. degree in electrical engineering from National Taiwan University in 2006. He held a post-doctoral position with the National Taiwan University of Science and Technology in 2007. He is currently an Associate Professor and the Chairman of the Department of Electrical Engineering with Tunghai University. His research interests are low-power system design, information security system, and VLSI design.

FANG-YIE LEU received the B.S. and M.S. degrees from the National Taiwan University of Science and Technology, Taiwan, in 1983 and 1986, respectively, the M.S. degree from the Knowledge Systems Institute, USA, in 1990, and the Ph.D. degree from the National Taiwan University of Science and Technology in 1991. He is currently a Workshop Organizer of CWECS and MCNCS workshops, a Professor with Tunghai University, Taiwan, the Director of the Database and Network Security Laboratory, and the Chairperson of the Department of Information Management of the University. He is also a member of the IEEE Computer Society and one of the editorial board members of at least six international journals. His research interests include wireless communication, network security, grid applications, and Chinese natural language processing.

ILSUN YOU (SM’13) received the M.S. and Ph.D. degrees in computer science from Dankook University, Seoul, South Korea, in 1997 and 2002, respectively, and the Ph.D. degree from Kyushu University, Japan, in 2012. From 1997 to 2004, he was with THINmultimedia Inc., Internet Security Company, Ltd., and Hanjo Engineering Company, Ltd., as a Research Engineer. Now, he is an Associate Professor with the Department of Information Security Engineering, Soonchunhyang University. He has served or is currently serving as a Main Organizer of international conferences and workshops, such as MobiWorld, MIST, ScCHID, AsiaARES, and so forth. His main research interests include internet security, authentication, access control, and formal security analysis. He is the Editor-in-Chief of the Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications. He is on the Editorial Board for Information Sciences, the Journal of Network and Computer Applications, the International Journal of Ad Hoc and Ubiquitous Computing, Computing and Informatics, the Journal of High Speed Networks, Intelligent Automation and Soft Computing, and Security and Communication Networks. He is a fellow of the IET.