

# みんなでおでんきPJ ～スマートメータシステムの実装～

二宮 政浩<sup>1</sup> 鈴木 才太<sup>1</sup> 佐々木 隆志<sup>1</sup> 都築 伸二<sup>1</sup> 鈴木 信<sup>2</sup> 兼築 史季<sup>2</sup> 早田 洋一<sup>3</sup>  
山田 芳郎<sup>1</sup>

**概要:** 太陽光パネルの発電量を見せ合うことで、コミュニティの形成や再生可能エネルギーの利活用の促進を目指す「みんなでおでんき」プロジェクトを筆者らは進めている。本稿では、製作した3種類のスマートメータと、それらの計測データを蓄積するクラウドサーバとの通信手順について述べている。提案するシステムが広く普及することを目的として、オープンソースハードウェアである Arduino マイコンや、Web サイト上で開発できる mbed マイコンを使ってスマートメータシステムを自作した。また既存の電力測定器も Home Gateway を経由させることにより、スマートメータ化した。クラウドサーバとの通信は、http の get/post メソッドを用いて、直観的に分かりやすい実装にしている。またクラウドサービス例として、HEMS (home energy management system) 用発電予報サービスを提案し、その予報アルゴリズムについて述べている。

## Cooperative Power Generation & Utilization, “Minna-de-Odenki” Project ～ Implementation of Smart Meter System ~

**Abstract:** The authors have been working on “Minna-de-Odenki” project aims to promote the utilization of renewable energy and form a community by showing the amounts of generating power by solar panels to the community members. In this paper, three types of experimental smart meters and a communication protocol with a cloud server, where the measured data are stored, have been described. To make the proposing system become familiar and widely used, the smart meters were implemented on Arduino microcontrollers which are open source hardware and “mbed” microcontrollers whose software can be developed on a website. It is also explained how to include an existing power meter into the smart meter system by using a proposing home gateway. The communication protocol used the get/post methods of http to make it easy and intuitive. As an example of cloud services, a forecast algorithm of electric power generation for HEMS (home energy management system) has been described.

### 1. はじめに

平成 24 年 7 月 1 日に開始された、再生可能エネルギーの固定価格買取制度 (FIT) によって、再生可能エネルギーによる発電設備の導入がすすんでいる。しかし、「環境 (CO<sub>2</sub>削減)」や「経済 (FIT による収益保証)」目的だけではなく、「地域でやらなければならない理由 (社会性)」を持たなければ、地域での導入が加速化/一般化しないことも指摘され始めている [1]。

防災は自助、共助そして公助の組み合わせであると言われるが、電力供給に関して言えば自家発電だけで閉じたシステムは自助、商用配電網に依存したシステムは公助に相当すると考えられ、前者は発電量として、後者は売電量として可視化できる。現状の電力供給においては共助という考え方が欠けており、地域やコミュニティでの電力の融通はこれを埋めるものである。電力における共助としては街区程度の規模での配電網や蓄電池を介して地域やコミュニティの構成員に電力の利用を許可する仕組みなどが考えられる。このような仕組みはいわばソーシャル給電とも言えるが、そのためには発電設備の所有者によるコミュニティの形成と、コミュニティ全体で利用可能な電力に関

<sup>1</sup> 愛媛大学  
<sup>2</sup> 株式会社パルソフトウェアサービス  
<sup>3</sup> 株式会社エス・ピー・シー

する情報の可視化が必要である。

そこで本研究<sup>\*1</sup>では、再生可能エネルギーによる発電設備の所有者や設置に関心を持つ者がコミュニティの中で発電量を見せ合ったり、地域での総発電量の集計に参加したり、発電に使用している機器やソーシャル給電に関する情報を交換したりするための基盤となるシステムを開発しており [2]、このプロジェクトを“みんなでおでんき”プロジェクトと呼んでいる。コミュニティの形成には既に広く使われている各種 SNS から取得できる人間関係であるソーシャルグラフをオプトインで利用する。コミュニティ全体での発電量やユーザが所属する地理的属性にもとづいた発電量を集計するためにはスマートメータを用いる。

本プロジェクトではハードウェア設計及びソフトウェア実装共に順次、プロジェクトサイト<sup>\*2</sup>に公開してゆく予定である。これにより再生可能エネルギーの個人による自助としての利活用が地域やコミュニティによる利活用に拡大され、地域で融通可能な電力量や災害時に利用可能な給電スポットの可視化といった電力供給における共助を可能にする基盤となると考えている。さらに再生可能エネルギーの普及におけるモチベーションの維持や、その利活用をきっかけとしたコミュニティの形成に寄与し、結果として街の付加価値を高める取り組みにつながることを期待している。

本稿の構成は以下のとおりである。第 2 章では、関連研究および技術を紹介する。第 3 章では、HGW(Home Gateway)の実装方法において Ubuntu-Linux を用いる場合の、メータと HGW 間の通信手順と、該当 HGW とクラウドサーバ間通信について述べる。また、これらのメータと HGW を一体化した mbed マイコン版 HGW についても述べる。第 4 章では、HGW 経由でクラウドサーバに蓄積された発電量データを利用した HEMS(Home Energy Management System) へのサービス例として発電予報を検討した結果を述べる。第 5 章はまとめである。

## 2. 関連研究および技術

近年の組み込みコンピューティングにおいて開発環境のコモディティ化が進んでいる。すなわち従来はある程度の初期投資が必要であったが、マイコンを用いたものづくりに対する敷居が低くなってきている。さらに 3D プリンターや卓上 CNC(Computerized Numerically Controlled) 工作機械の入手が容易になったことから、ものづくりの新しい潮流としてパーソナル・ファブリケーションが注目されている [3]。この流れを受けて本研究では特に HGW の設計およびクラウド上のサービスとの間の API を公開することで幅広い参加者に門戸を開いている。ソースコードを

公開することでコミュニティの知恵と経験を活かすオープンソース・ソフトウェアが広く受け入れられて久しいが、本研究ではいわばオープン・ハードウェアを目指している。

オープン化の流れはソフトウェアやハードウェアにとどまらず、データそのものにも及んでいる。すなわち、従来は特定の分野だけで閉じた利用にとどまっていたデータを、個人のプライバシーや社会への影響に配慮しながら公開することで、効果的な利活用を推進するオープンデータの流れである [4]。本研究でも参加者から収集したデータのうち許諾を得たものについては匿名化もしくは統計処理を経て API を通じて公開する予定である。現在はビッグデータと呼べるほどの規模のデータは蓄積されていないが、API を公開することで他のオープンデータとのマッシュアップやリアルタイムな分析に、本研究で収集するデータを活用することができる [5]。

「みんなでおでんき」プロジェクトでは参加者が送信した発電量に関する情報を可視化するにあたって、SNS のソーシャルグラフを使っており、これによりコミュニティとしての発電量の算出やコミュニティ間での発電量の比較を可能にしている。これは送信者だけが見ることのできるパーソナルな可視化とオープンデータのギャップを埋めるものである。コミュニティとしての発電量が可視化されることで、電力買い取り制度を利用した売電という経済的なインセンティブだけでなく、電力共助を担う一員としての達成感が共有され、PEB (Pro-Environmental Behavior, 環境配慮行動) が促進されることが期待される。命令や強制によってではなくコミュニティの雰囲気や達成感といった緩やかで暗示的なコミュニケーションが PEB の促進に果たす効果についての研究も進んでいる [6]。発電ではないが達成感を共有するという点で類似の取り組みとして、個人による野菜の育成を促進させるためのコミュニケーションシステムについての研究がある [7]。

地域やコミュニティでのエネルギー管理は CEMS(Cluster/Community Energy Management System) と呼ばれ、主に電力需要に応じた電力供給を行うデマンドレスポンス (需要応答) のために用いられる。CEMS では各電力消費者の需要をリアルタイムで把握する必要があり、そのために IEEE 1888 といった大規模センシング・データ・レジストリ向けの標準規格を使った取り組みが行われている [8]。さらに近年では複数の電力需要家がピークをずらすなど協調に電力を消費するための研究も行われている [9], [10]。一方、家庭内のエネルギー管理は HEMS と呼ばれ、我が国では主に家電機器の電力監視及び制御を目的としたネットワーク規格である ECHONET Lite に基づく標準化がすすめられている [11]。また HEMS の制御をクラウドサービスとして実現しようとする研究も行われている [12]。HEMS には温湿度や人感センサを含めることができるため、この情報を空調や照明の制御に用

<sup>\*1</sup> 本研究は総務省の戦略的情報通信研究開発推進制度 (SCOPE) の受託研究の一部である。

<sup>\*2</sup> <http://www.odenk.org/>

いることができる。さらに個人ごとの省エネに対する意識の違いを推定しより満足度の高い省エネ行動を推奨したり [13], ニュースやソーシャルデータといった多様なデータストリームも合わせて利用することでユーザの状況を詳しく把握し, よりきめ細やかな住環境の制御を実現する取り組みもなされている [14].

本プロジェクトに先立ち, 平成 22 年度に愛媛県松山市の住宅分譲地「ていれぎ」にて環境負荷軽減のための ICT システムに関する実証実験が行われた\*3。この実証実験では実験に協力可能な世帯を 10 世帯募集し, 分電盤, 電力コンセント, 蓄電池, 太陽光発電, 風力発電機, ガス, 上水道にスマートメータを, また屋内外に温度, 湿度, 照度, 人感センサを取り付けモニタリングを行うとともに電気自動車によるカーシェアリングを行った。一部の世帯については現在もセンサ類からデータを収集中である。本稿で述べるスマートメータシステムではデータをクラウドサーバで保存しているが, 蓄積すべきデータの種類やデータ量の見積もりにはこの実証実験によって得られた知見を活かした。

本研究では開発したスマートメータシステムの応用例として, 天気予報, 洗濯指数, 降水確率から発電量を予想する HEMS 用発電予報サービスを提案する。なお, 短期的な発電量の予測予報については気象観測情報を用いない手法も提案されている [15].

### 3. スマートメータシステム

本稿では, 発電量の他消費電力量, 気温等の環境データを計測し, クラウドサーバに伝送する機能, およびそのデータを処理した結果に基づき配電機器や家電製品の制御を行う HEMS 機能を装備したシステムを, スマートメータシステムと呼ぶ。

#### 3.1 Ubuntu-Linux 版 HGW

測定した電力データをクラウドサーバへ伝送する HGW を, Linux ディストリビューションの一つである Ubuntu\*4 を用いて PC に実装した。電力メータは Arduino マイコン\*5 を用いて自作したものと, 既存の電力計を USB で HGW に接続することによりスマートメータ化したものの 2 種類である。図 1 に, 本節で述べる Ubuntu-Linux 版 HGW を利用したスマートメータシステムの構成例を示す。

なお本稿では述べないが, 上記ソーシャル給電用に“ソーシャルコンセント”を作成している [16]。このコンセントを搭載する場合は, 図 1 の家電機器の一部となる。

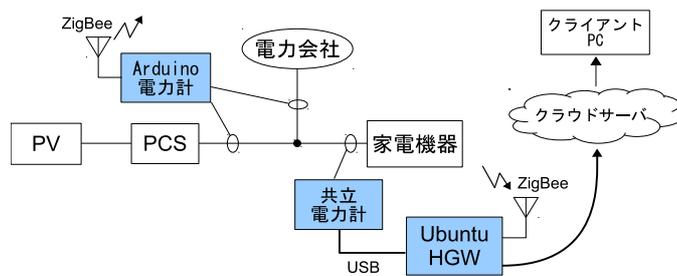


図 1 スマートメータシステムの構成例  
 (Ubuntu-Linux 版 HGW を利用する場合)

Fig. 1 Example of structure of the proposing smart meter system using HGW based on Ubuntu-Linux.

#### 3.1.1 Arduino 電力計-HGW 間通信

自作電力メータ用に使用した Arduino マイコンは, オープンソースハードウェアである。図 1 に示すように, 太陽光パネル (PV) の直流出力を交流に変換する PCS (Power Conditioning Subsystem) の出力電力と, 電力会社から購入している電力を同時に測定できるように設計し製作した。余剰電力を売電している時は, 購入電力量が負値になるため, PV 側と電力会社側との間の電力の売買の関係を示すことができる。また, 2 箇所の電力を同時に測定することによって, 家電機器で消費している電力が PV の発電電力だけで補えているのかといったことが分かる。

Arduino 電力計は図 1 に示すように ZigBee 規格の無線モジュールを搭載し, 測定した電圧, 電流, 及びこれらから求めた電力, 及び力率のデータを一定間隔毎 (本稿では 40 秒毎) に送信している。ビットレートは 9600bps である。HGW 側にも ZigBee 無線モジュールが接続されており, 測定データを一旦 HGW で収集する。

Arduino 電力計と HGW 間のデータフォーマットを図 2 に示す。1 パケット 48Byte (B) とし, プリアンプルは乱数 4B, ArduinoID (2B) は上位バイトがクラウドから Arduino を特定するための ID, 下位バイトが Arduino に接続されているデバイスを特定するための ID となっている。この下位バイトのデバイス ID で, 測定データが発電量であるか, 購入電力量であるかを区別している。AD[0]~AD[5] (4B) は Arduino に 6 個の AD コンバータがあること, また cal[0]~cal[2] (4B) はそれらの信号処理結果として元々定義した名称の名残りであり, 本稿では図 2 の通りの値が挿入されている。DD (2B) は Arduino の 14 本のデジタル I/O の状態を示す。誤り検出符号には CRC16 (Cyclic Redundancy Check - 16bit) を用い, HGW で誤りチェックを行う。CRC16 のアルゴリズムは Arduino ライブラリ\*6 を用いている。受信した測定データは, 一旦 HGW に保管され, 一定間隔毎 (本稿では 1 分毎) にクラウドサーバへ伝送する。

\*3 総務省による平成 22 年度環境負荷軽減型地域 ICT システム基盤確率事業「愛媛県松山市における ICT の技術仕様の検証のための地域実験, 受託者: 鹿島建設, アバンソシエイツ, エス・ピー・シー, 横須賀テレコムリサーチパーク, ユーシーテクノロジー

\*4 Meet Ubuntu, <http://www.ubuntu.com/ubuntu>

\*5 What is Arduino?, <http://arduino.cc/en/Guide/Introduction>

\*6 <https://github.com/arduino/avrduide/blob/master/crc16.c>

0	1	2	3	4	5	6	7
preamble(乱数)			Length	R/W 他 Control	Arduino ID		
AD[0] : 電圧1 [V]			AD[1] : 電流1 [A]				
AD[2] : 皮相電力1 [VA]			AD[3] : 力率1 [%]				
AD[4] : 有効電力1 [W]			AD[5] : 電流2 [A]				
cal[0] : 全皮相電力 [VA]			cal[1] : 全力率 [%]				
cal[2] : 全有効電力 [W]			DD: digital I/O		CRC16		

(注)  
CRC16の計算範囲

図 2 Arduino 電力計-HGW 間データフォーマット

Fig. 2 Data format of communication between Wattmeter by Arduino and HGW.

### 3.1.2 共立電力計-HGW 間通信

既存の測定器を利用する例として、共立電力計 (KY-ORITSU 製デジタルパワーメータ KEW6305) を本稿では用いた。USB で HGW に直接接続することによりスマートメータ化した、HGW から一定間隔毎 (本稿では 1 分毎) に測定データ取得コマンドを共立電力計に送ることにより、データを取得した。取得できる測定データは図 2 と同様であり、測定時刻、電圧、電流、有効電力、力率、皮相電力、電力量などのデータが含まれる。

### 3.2 HGW-クラウドサーバ間通信

HGW では、各電力計から受信したデータを以下に述べるフォーマットに変換し、クラウドサーバへ伝送する。Arduino 電力計からデータを受信した場合、ArduinoID によって測定対象が判別できるので、それに従ってクラウドサーバへ伝送するデータフォーマットを使い分ける。

発電量データを送信する場合の HGW とクラウドサーバ間のデータフォーマット例を図 3(a) に、購入電力量データを送信する場合を図 3(b) に示す。ArduinoID により緑色背景のフィールドが変化する。測定値に加えて、HGW の製品名 (productName), HGW のシリアル番号 (serialNumber), HGW の配下に接続されているマイコンモジュール ID (moduleId), マイコンモジュールに接続されているセンサ名 (sensorName), センサを識別する識別子 (sensorId), 測定地の緯度 (longitude), 経度 (latitude), 所在地 (address), 測定時刻 (time) などのデータを HGW で作成する。またセンサからは計測対象やセンサの種類によって様々な測定値が送られてくるので、対象となる測定値の発生源 (from), 対象となる測定値が消費される先 (to), 単位 (unit), 測定値 (value), 測定値の分解能 (resolution), 測定場所 (in, on, at, of など) などのプロパティも用意した。白色背景のフィールドは、value フィールドに付随するデータである。

同様に、共立電力計から測定データを受信した場合の、HGW とクラウドサーバ間のデータフォーマットを図 4 に示している。この場合も複数台の測定器が USB 接続される場合は、緑色のフィールドが変化する。

HGW からクラウドサーバへの伝送間隔は 1 分とし、ク

データ名	データ例	HGW が作成	Arduino が作成
productName	Ubuntu-nino	○	
serialNumber	001	○	
moduleId	Arduino-UnoR3-01	○	
sensorName	Wattmeter-power.gen	○	
sensorId	CT12-1	○	
from	pcs	○	
unit	Watt	○	
value (全有効電力)	987.123456		○
resolution	10	○	
in	house	○	
latitude	33.850437	○	
longitude	132.774011	○	
address	愛媛大学	○	
time	20130101120000	○	
arduinoId	0x2101		○
ad0 (電圧1)	106.06661		○
ad1 (電流1)	1.384978		○
ad2 (皮相電力1)	146.89996		○
ad3 (力率1)	61.776749		○
ad4 (有効電力1)	90.750023		○
ad5 (電流2)	1.361387		○
cal0 (全皮相電力)	150.89996		○
cal1 (全力率)	71.776749		○
cal2 (全有効電力)	987.123456		○
dd (Digital I/O)	0xFFFF		○

(a) 発電量データの場合

データ名	データ例	HGW が作成	Arduino が作成
productName	Ubuntu-nino	○	
serialNumber	001	○	
moduleId	Arduino-UnoR3-01	○	
sensorName	Wattmeter-power.buy	○	
sensorId	CT34-1	○	
from	epc	○	
unit	Watt	○	
value (全有効電力)	987.123456 (負の数 のときは売電を意味する)		○
resolution	10	○	
of	Whole-House	○	
latitude	33.850437	○	
longitude	132.774011	○	
address	愛媛大学	○	
time	20130101120000	○	
arduinoId	0x2111		○
ad0 (電圧1)	106.06661		○
ad1 (電流1)	1.384978		○
ad2 (皮相電力1)	146.89996		○
ad3 (力率1)	61.776749		○
ad4 (有効電力1)	90.750023		○
ad5 (電流2)	1.361387		○
cal0 (全皮相電力)	150.89996		○
cal1 (全力率)	71.776749		○
cal2 (全有効電力)	987.123456		○
dd (Digital I/O)	0xFFFF		○

(b) 購入電力量データの場合

図 3 HGW-クラウドサーバ間データフォーマット  
(Arduino 電力計の場合)

Fig. 3 Data format of communication between HGW and Server. (When Wattmeter by Arduino was used.)

クラウドサーバとの通信は、http の get メソッドを用いて、直観的に分かりやすい実装にしている。クラウドサーバには、httpd 以外に特別なソフトウェアは不要である。本稿では実装していないが、get コマンドのレスポンスに HEMS 用のコマンドを付加すれば、家電機器の遠隔制御も可能で

データ名	データ例	HGW が作成	KEW6305 が作成
productName	Ubuntu-nino	○	
serialNumber	001	○	
moduleId	KEW6305-01	○	
sensorName	Wattmeter-power.consume	○	
sensorId	1P3W	○	
unit	Watt	○	
value (有効電力総和)	534.123456		○
resolution	1	○	
in	house	○	
of	air-conditioner	○	
latitude	33.828529	○	
longitude	132.77695	○	
address	愛媛大学	○	
time	20130101120000		○
ELAPSED-TIME (経過時間)	00024:21:57		○
V1 (電圧1)	105.300003		○
V2 (電圧2)	105.199997		○
A1 (電流1)	5.441		○
A2 (電流2)	5.391		○
P (有効電力総和)	534.123456		○
P1 (有効電力1)	240.300003		○
PF (全体の力率)	0.469		○
PF1 (力率1)	0.42		○
S (皮相電力総和)	1140		○
S1 (皮相電力1)	572.700012		○
+WP (有効電力量総和・消費)	5781.259766		○
+WP1 (有効電力量1・消費)	2676.689941		○
-WP (有効電力量総和・回生)	0		○
-WP1 (有効電力量1・回生)	0		○
WPT (有効電力量総和・総合)	5781.259766		○
WPT1 (有効電力量1・総合)	2676.689941		○

図 4 HGW-クラウドサーバ間データフォーマット  
 (共立電力計の場合)

Fig. 4 Data format of communication between HGW and Server. (When Wattmeter by Kyoritsu was used.)

ある。ただしセキュリティへの配慮が必要となるため、今後の課題である。

### 3.3 mbed マイコン版 HGW

Ubuntu-Linux 版 HGW の他に、Web サイト上で開発できる mbed マイコン\*7 による HGW も実装した。使用したマイコンの主なスペックを表 1 に示す。標準でインターネットのインターフェースを搭載しており、Internet に接続することができる。そこで 3.2 節で述べた HGW 機能を実装することで、Ubuntu-Linux 版 HGW を経由することなくクラウドサーバへ測定データを直接伝送できる。動作時の消費電力は約 1W であり、マイコン本体は 5,000～6,000 円で入手できるため、低消費電力化、低コスト化に有用である。

「みんなでおでんき」プロジェクトでは発電量データ以外に、温度、湿度、照度も収集しており、本稿では図 5 に示すように、これらを mbed マイコンで測定し直接クラウドサーバへ伝送した。NTPClient ライブラリ\*8 を用いて現在時刻を NTP サーバから取得し、クラウドサーバと

\*7 mbed NXP LPC1768, <http://mbed.org/handbook/mbed-NXP-LPC1768>

\*8 [http://mbed.org/users/donatien/code/NTPClient\\_HelloWorld/](http://mbed.org/users/donatien/code/NTPClient_HelloWorld/)

表 1 mbed マイコンのスペック

Table 1 Specifications of mbed

メーカー	NXP Semiconductors
プロセッサ	ARM Cortex-M3 (32bit)
クロック	96MHz
FLASH	512KB
RAM	64KB
動作電圧	3.3V
電源	USB または VIN へ 4.5～9V
消費電力	約 1W
価格	5000～6000 円

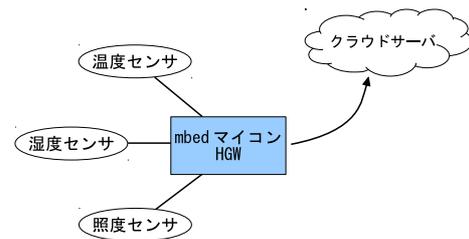


図 5 スマートメータシステム  
 (mbed マイコン版 HGW を利用する場合)

Fig. 5 Smart meter system using mbed based HGW.

データ名	データ例	mbed HGW が作成
productName	mbed	○
serialNumber	001	○
moduleId	mbed-001	○
sensorName	Temperature-sensor	○
sensorId	LM60	○
unit	degC	○
value (気温)	20.123456	○
resolution	1	○
in	laboratory	○
latitude	33.828529	○
longitude	132.77695	○
address	愛媛大学	○
time	20130101120000	○
temp (気温)	20.123456	○
RH (湿度)	30.123456	○
Lux (照度)	40.123456	○

図 6 mbed マイコン版 HGW-クラウドサーバ間データフォーマット

Fig. 6 Data format of communication between HGW based on mbed and Server.

の通信には、httpClient ライブラリ\*9 の post メソッドを使用している。Ubuntu-Linux 版 HGW とクラウドサーバ間での通信と同様に、図 6 に示すデータフォーマットを作成し、1 分間隔で測定データをクラウドサーバに伝送している。なお、mbed マイコンの配下に Arduino マイコンが接続される場合は、productName, serialNumber 以外のフィールドは図 3 のデータフォーマットになる。

\*9 [https://mbed.org/users/donatien/code/HTTPClient\\_HelloWorld/](https://mbed.org/users/donatien/code/HTTPClient_HelloWorld/)

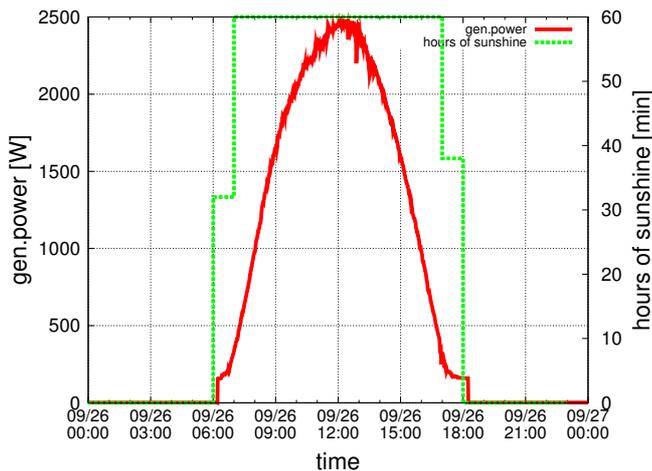


図 7 発電量と日照時間の関係。(2012/9/26 測定)

Fig. 7 Relationship between power generation and hours of sunshine.(2012/9/26 measurement)

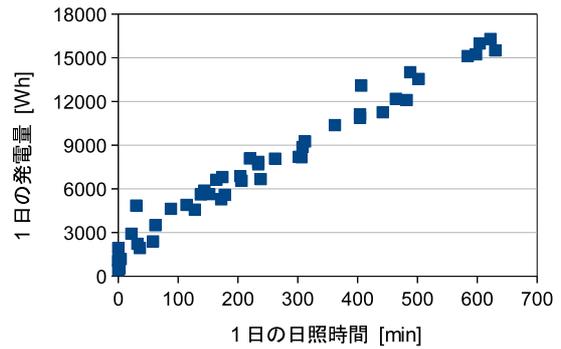


図 8 発電量と日照時間との相関。(2012/10/15~12/10)

Fig. 8 Correlation between power generation and hours of sunshine. (2012/10/15~12/10)

## 4. クラウドサービス例：発電予報

### 4.1 発電量と日照時間との相関

測定した発電量データは、HGW を経由してクラウドサーバに蓄積されている。そのデータをクライアント側 PC が、一定時間毎に取得するようにした。また、発電量と相関が強いと考えられる日照時間データも天気予報サイトのアメダス\*10から取得し、1日の発電量と日照時間を同じグラフ上に重ねて表示できるようにした。日照時間とは、日照計で測定される日射量が  $120W/m^2$  以上である時間と定義され、使用したアメダスでは、1時間ごとに観測された積算日照時間が記録されている。1日の発電量と日照時間を比較した例を図 7 に示す。この例では1日中晴れていたため日照時間は最高 (60min/h) であり、発電量は綺麗な放物線を描いている。PV は南向きに設置しているので、12時に最高出力を得ている。なお快晴ではなかったため発電量を示す線はギザギザしている。

約 2 ヶ月分の、1日の合計日照時間と発電量との関係を図 8 に示す。これらの結果から、発電量と日照時間との関係は極めて強いことがわかる。

### 4.2 発電予報アルゴリズム

サーバに蓄積された発電量データを利用したクラウドサービス例として、HEMS 用「発電予報」サービスを提案する。予想発電量に応じて家電機器の消費電力を調整し、省エネ・節電を行ったり、深夜電力を蓄電しているのであればその放電計画を立てたりすることが可能となる。4.1 節で日照時間には発電量と強い相関があることを示したが、日照時間予報のデータを無料で提供しているサイトは無い

ため、本稿では無料で提供されている天気予報、洗濯指数、降水確率を利用した発電予想アルゴリズムを検討した。

以下に列挙するように、毎日 6, 9, 12, 15, 18 時の 3 時間おきの実際の天気と、これらの時間に対応する 3 時間毎の発電量、及びその日の洗濯指数、降水確率 (午前、午後) を記録する。

- 日付
- 6 時の天気  $\alpha$  (晴れの時 = 1, 曇りの時 = 2, 雨の時 = 3)
- 9 時の天気  $\beta$  (同上)
- 12 時の天気  $\gamma$  (同上)
- 15 時の天気  $\delta$  (同上)
- 18 時の天気  $\epsilon$  (同上)
- 4 時半~7 時半の発電量  $P_6$  [Wh]
- 7 時半~10 時半の発電量  $P_9$  [Wh]
- 10 時半~13 時半の発電量  $P_{12}$  [Wh]
- 13 時半~16 時半の発電量  $P_{15}$  [Wh]
- 16 時半~19 時半の発電量  $P_{18}$  [Wh]
- 1 日の合計発電量  $P_{day}$  [Wh]
- 洗濯指数  $\hat{W}_i$  (0~100 の整数値)
- 降水確率 (午前)  $\hat{R}_1$  [%]
- 降水確率 (午後)  $\hat{R}_2$  [%]

これらのデータログの中から、発電予報には直前の過去 30 日分のデータを使用する。まず各時間帯の天気に対する発電実績の平均値を求める。つまり過去 30 日のうち、6 時の天気が '晴れ' だった日の 4 時半~7 時半の発電量の平均値、同様に '曇り' だった日、'雨' だった日の 3 通りの平均値  $\bar{P}_6^{\alpha=1}$ ,  $\bar{P}_6^{\alpha=2}$ ,  $\bar{P}_6^{\alpha=3}$  [Wh] をそれぞれ求める。同じく、9 時, 12 時, 15 時, 18 時の 4 通りの場合においても平均値を算出する。

これらの平均値を用いて、翌日の予報発電量  $\hat{P}_{day}$  [Wh] を次式のように定義する。なお各パラメータに重みをつけるため、A~H の定数を付加している。

\*10 [http://tenki.jp/slashcomponent/slashstatic\\\_api/slashrss/slashamedas/point-1037.xml](http://tenki.jp/slashcomponent/slashstatic\_api/slashrss/slashamedas/point-1037.xml)

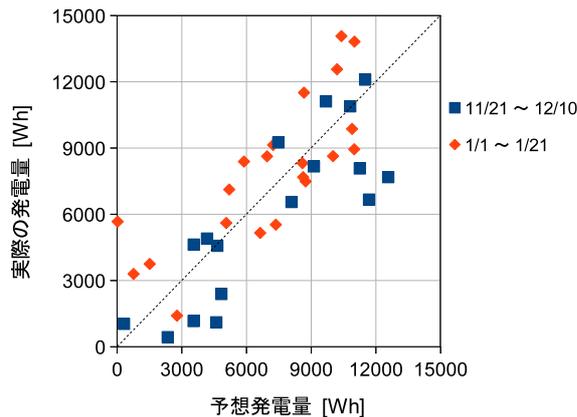


図 9 発電予報例

Fig. 9 An example of power generation forecast

$$\hat{P}_{day} = A \times \bar{P}_6^{\hat{\alpha}} + B \times \bar{P}_9^{\hat{\beta}} + C \times \bar{P}_{12}^{\hat{\gamma}} + D \times \bar{P}_{15}^{\hat{\delta}} + E \times \bar{P}_{18}^{\hat{\epsilon}} + F \times \hat{W}_i + G \times \hat{R}_1 + H \times \hat{R}_2 \quad [\text{Wh}] \quad (1)$$

式(1)を過去30日のデータに適用し、重回帰分析によりA~Hの値を算出した。求めたA~Hの値を式(1)に代入したものが発電量予報の式となる。つまり、予測したい日の天気予報値( $\hat{\alpha}$ ,  $\hat{\beta}$ ,  $\hat{\gamma}$ ,  $\hat{\delta}$ ,  $\hat{\epsilon}$ )とそれに対応した3時間ごとの過去の実績平均発電量、洗濯指数予報、降水確率予報(午前、午後)を代入し、発電量を予報する。

以上の発電予報アルゴリズムを用いたときの、予想発電量と実際の発電量との関係の一例を図9に示す。ある程度の相関は得られたが、時期により傾向の違いが生じたので、2012年11月21日~12月10日と2013年1月1日~1月21日の場合に分けて表示している。破線は予想と実測値が一致したときの理想線を示している。11月21日~12月10日の場合、実測値より高めに予想する日が多かったが、1月1日~1月21日の場合は低めの予想であった。両期間の間に冬至を挟んでおり、季節の違いによる影響が原因の一つであると考察している。

## 5. おわりに

本稿では、自作したArduino電力計とUbuntu-Linux上に実装したHGWから構成されるスマートメータシステムの実装方法について述べた。またArduino電力計以外にも、既存の電力測定器もHGWに接続できる場合は提案するスマートメータシステムに組み込むことを示した。HGWには、測定器からデータを一旦収集しフォーマット変換後、クラウドサーバへ伝送する機能を実装した。各電力計とHGW間の通信手順について述べ、HGWとクラウドサーバ間の通信に使用するフォーマット例を提案した。さらに、低消費電力化、低コスト化を目的として、Arduino電力計とHGWを一体型にしたmbedマイコン版HGWも

提案し、気温等の電力量以外のセンサデータをクラウドサーバへ伝送する機能を実装した。

クラウドサーバに蓄積されたデータを利用したサービス例として、HEMS用発電予報サービスを提案した。天気、洗濯指数、降水確率予報をもとにするアルゴリズムであり、発電予報と実測値とがある程度一致することを示した。今後予報精度を高めるためには、季節の変化を考慮したアルゴリズムに改良する必要がある。また考案した発電予報サービスのシステムをHEMSに適用することが今後の課題である。

発電予報サービス以外にも、クラウドサーバに蓄積されたデータを有効活用したサービスについても検討し、「みんなでおでんき」プロジェクトが、再生可能エネルギーの利活用促進の一助となることを期待している。

## 参考文献

- [1] 相川高信, 阿部剛志, 大澤拓人, 浅田陽子, 小川拓哉, 高橋 溪, 村上聡江: エネルギー自治の必要性と現状, そして将来への課題, 季刊 政策・経営研究, Vol. 3 (オンライン), 入手先 ([http://www.murc.jp/thinktank/rc/quarterly/quarterly\\_detail/201203.01](http://www.murc.jp/thinktank/rc/quarterly/quarterly_detail/201203.01)) (2012).
- [2] 佐々木隆志, 鈴木 信, 兼築史季, 早田洋一, 都築伸二: スマートメータとSNS連携による再生可能エネルギー利活用促進に関する取り組み, 情報処理学会研究報告. IS, 情報システムと社会環境研究会, Vol. 2012-IS-119, No. 7, pp. 1-6 (2012).
- [3] 田中浩也: デジタルファブリケーション:8. パーソナルファブリケーション時代におけるものづくりのオープンソース化の動向とFab Commonsの提案, 情報処理, Vol. 54, No. 2, pp. 127-134 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/110009500483/>) (2013).
- [4] 庄司昌彦: [招待講演] 日本におけるオープンデータの活用に向けて, 情報処理学会研究報告. EIP, [電子化知的財産・社会基盤], Vol. 2012, No. 4, pp. 1-5 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/110009478993/>) (2012).
- [5] 喜田弘司, 藤山健一郎, 中村暢達: 実世界から発生するビッグデータをリアルタイムに処理するストリーム分析基盤(ユビキタス・センサネットワークを支えるシステム開発論文特集), 電子情報通信学会論文誌. B, 通信, Vol. 95, No. 11, pp. 1396-1403 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/40019479807/>) (2012).
- [6] 青柳西蔵, 岡村智明, 石井裕剛, 下田 宏: ゆるいコミュニケーションによる環境配慮行動の継続促進手法の提案と評価, ヒューマンインタフェース学会論文誌, Vol. 13, No. 3, pp. 207-220 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/10029465536/>) (2011).
- [7] 山内正人, 藤枝 慶, 芝原隼人, 安澤太郎, 西條鉄太郎, 廣井 慧, 砂原秀樹: aroots: 個人間での野菜育成を促進させるコミュニケーションシステムの提案, 情報処理学会論文誌 コンシューマ・デバイス&システム (CDS), Vol. 2, No. 1, pp. 20-27 (2012).
- [8] NTT コムウェア品質生産性技術本部研究開発部: スマートコミュニティを支える基盤技術: 大量センシングポイント・コンポーネントの管理, アドレス解決技術 IEEE1888 レジストリの概要, NTT 技術ジャーナル, Vol. 24, No. 11, pp. 38-41 (オンライン), 入手先 (<http://www.ntt.co.jp/journal/1211/files/jn201211038.pdf>) (2012).
- [9] 朝妻智裕, 大場義和: クラウド活用サービスと需要家協調型

- スマートグリッド (特集需要設備向けスマートグリッド対応技術), 電気設備学会誌, Vol. 32, No. 8, pp. 577-580 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/40019411789/>) (2012).
- [10] 松澤茂雄, 山田孝裕, 宮崎一彦: Web サービス技術を用いて地域需要家の連携を実現する CEMS (特集 将来のエネルギーインフラを支えるスマートグリッド技術), 東芝レビュー, Vol. 66, No. 12, pp. 12-15 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/40019142019/>) (2011).
- [11] 昌二望月: ECHONET Lite の概要と普及に向けた取り組み (特集 GreenIT による省エネルギー支援), 産業と環境, Vol. 41, No. 5, pp. 29-32 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/40019339721/>) (2012).
- [12] 増尾 剛, 中村二郎, 松岡茂登: リアルタイム Web 技術による HEMS サービスクラウド化の検討 (ネットワークシステム), 電子情報通信学会技術研究報告: 信学技報, Vol. 112, No. 350, pp. 1-6 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/40019545513/>) (2012).
- [13] 岡村雄敬, まつ本真佑, 中村匡秀: ペルソナシナリオ法を用いた個人適応型省エネ行動の分析法 (ホームネットワーク, ユビキタスネットワーク, クラウドコンピューティング, コンテキストアウェア, 位置情報サービス, e コマース及び一般), 電子情報通信学会技術研究報告. IN, 情報ネットワーク, Vol. 110, No. 289, pp. 77-82 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/110008145106/>) (2010).
- [14] 鈴木誠二, 峰野博史, 太田 賢, 水野忠則: 多種多様なデータストリームを用いた家電制御システムの構築, 情報処理学会研究報告. GN, [グループウェアとネットワークサービス], Vol. 2011, No. 17, pp. 1-8 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/110007891153/>) (2011).
- [15] 隼 松本, 大介石井, 聡 岡本, 英司大木, 直明山中: スマートグリッド環境において気象観測情報を用いない高精度な短期的太陽光発電出力予測手法の提案 (特集スマートな社会を支えるインターネットアーキテクチャ論文), 電子情報通信学会論文誌. B, 通信, Vol. 94, No. 10, pp. 1373-1382 (オンライン), 入手先 (<http://ci.nii.ac.jp/naid/110008749652/>) (2011).
- [16] 松重, 鈴木, 佐々木, 都築, 鈴木, 兼築, 早田, 山田: みんなでおでんき PJ ~ソーシャルコンセンソの実装~, 情報処理学会研究報告. IS, 情報システムと社会環境研究会, Vol. 2013-IS-123 (2013).