

屋外設置型端末のノーマリオフ実装

荒 川 祐 真[†] 長 崎 健[†] 戸 田 真 志^{††}
平 田 圭 二[†] 松 原 仁[†]

本研究では、屋外設置型端末に対してノーマリオフにより省電力化を図る。ノーマリオフとは、「使わないときは電気を切る」という考えで、待機時の消費電力 0 を目指す。しかし、揮発性デバイスは電気を切ると内部情報が消えてしまう。そこで、これらの内部情報を退避する為に不揮発性メモリ (NVRAM) を用いる。本研究では、ノーマリオフを実現する機能を OS 上のタスクとして実現する。このタスクは省電力機能を管理する枠組みとして動作する。TOPPERS/ASP カーネルに省電力手法を実装し、RX63N 評価ボードを用いて消費電力の評価を行った。実験結果から、ノーマリオフの機能を利用した場合の消費電力は既存の省電力手法を利用した場合の半分に抑えられた。

Implementation of the Normally-off for a Outdoor Terminal

YUMA ARAKAWA,[†] TAKESHI NAGASAKI,[†] MASASHI TODA,^{††}
KEIJI HIRATA[†] and HITOSHI MATSUBARA[†]

In this study, we reduce power consumption for the outdoor terminal by the normally-off. The normally-off is the idea that turns the power-off when the system is idle. By the normally-off, the power consumption is zero when the system is idle. The volatile devices, however, can't keep own internal information when the system is the power-off. So we use non volatile RAM (NVRAM) to save the internal information. We made the normally-off function and implemented it as a task on an OS. It is a framework for the power-saving function. We implemented it into the TOPPERS/ASP kernel and evaluated the power consumption on the RX63N evaluation board. As a result of the experiment, the power consumption using the normally-off function was half of the power consumption using the existing power-saving function.

1. 背 景

屋外設置型の端末で、液晶パネルなどの表示装置、タッチパネルやボタン、焦電センサーなどの入力装置を備えたものが増えている。近年では、スマートシティを背景に、知的バス停などの研究も広まっており¹⁾、このような屋外設置型システムは今後様々な応用がなされていくものと思われる。

このようなシステムの場合、屋外という設置場所の制約上、電源の確保が困難である。特に電池駆動である場合にはシステムの消費電力は電池交換にかかるメンテナンスコストや、電池の大きさによるシステム自体の物理的な大きさにも繋がる。そのため、このようなシステムにおいては、システムの低消費電力化は魅力的である。

屋外設置型のシステムにも様々なものがある。人と関わって動作するものとしては、人へ情報を提示するデバイス、人が情報を入力するデバイス、またそれら両方の機能を持ったデバイスがある。また、自律的に環境の情報を収集するデバイスとしては、センサーネットワークデバイスがある。本論文では、人と関わって動作するシステムで、人へ情報を提示するデバイスを対象として扱う。

屋外設置型のこのようなデバイスは、24 時間常に人に対して情報を提示しているという状況は考えにくく、大半は人の感知を待つ待機の状態である。そのため、この待機時間において効率的に省電力化を図ることが有効である。

2. 目 的

本研究の目的は、システムのノーマリオフ化により環境設置型システムの低消費電力化を実現することである。「ノーマリオフ」とは、端的には「使わないときには電気を切る」という考え²⁾で、処理すべき作業の

[†] 公立はこだて未来大学
Future University Hakodate
^{††} 熊本大学
Kumamoto University

無い待機時は，CPU を含めたシステム上のデバイスの電源について，積極的な電源遮断を図る．

しかし，CPU や RAM などの揮発性のデバイスは，電源を切ると内部に保存された情報を失ってしまう．そのため，電源を切る際には揮発性デバイスの内部情報を不揮発性メモリ（NVRAM）へ退避する．また，動作を再開する際には揮発性デバイスに保存された内部情報を NVRAM から復帰する．本研究では，このようなノーマリオフを実現するためのハードウェアの枠組みと，OS の機能を提案，実装する．

OS への追加機能は，省電力化のための動作を自律的に行うものとして実装する．また，省電力機能の追加の際に既存のタスクを修正することなく適用可能で，省電力の方針自体もアプリケーションのタスク開発者から設定可能な枠組みとする．

3. 関連研究

3.1 既存の省電力技術

既存の省電力技術としては，プロセッサコアの間欠動作（パワーゲーティング）により低消費電力化を図る研究がある（木村ら³⁾）．木村らは，プロセッサコアの各演算ユニットに対してパワーゲーティングが行えるプロセッサコア「Geyser」を用いて，パワーゲーティングの実施ポリシーを定める手法を提案している．そこで，木村らは各演算ユニットのパワーゲーティングにおける損益分岐点がプロセッサコアの温度情報により変化する点に着目し，損益分岐の条件に温度のパラメータを含めた条件式を提案している．

また，プロセッサ動作周波数を各プロセス毎に動的に変更することで省電力化を図る研究がある（宮川ら⁴⁾）．宮川らの研究では，OS レベルの電力制御手法として電力制御スケジューラ PCCS（Power Consumption Controlling Scheduler）を提案し，プロトタイプを実装している．プロセッサの動作速度と消費電力量がトレードオフの関係にあることから，このスケジューラにより各プロセスの動作速度に基づいて動作周波数を変更している．

周期動作を行うセンサーネットワークデバイスを対象に，待機時の省電力化を図る研究がある（安部ら⁵⁾）．安部らの研究では，待機時は周辺回路の電源を遮断し，MCU と無線モジュールを省電力モードへ移行することで省電力化を図る．ソフトウェア制御で周辺回路の電源を遮断するために，周辺回路の電源を MCU から管理するための MOS-FET を設置している．

木村らの研究³⁾ や宮川らの研究⁴⁾ は CPU のみを省電力の対象としたものであり，安部らの研究⁵⁾ は待機

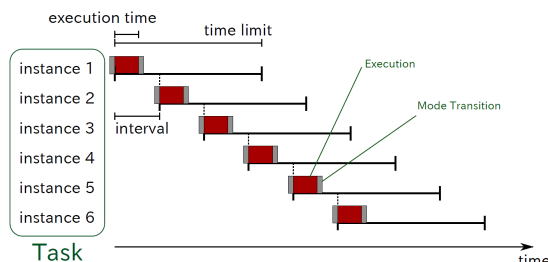


図 1 従来のタスクスケジューリング⁶⁾
Fig. 1 Traditional task scheduling⁶⁾

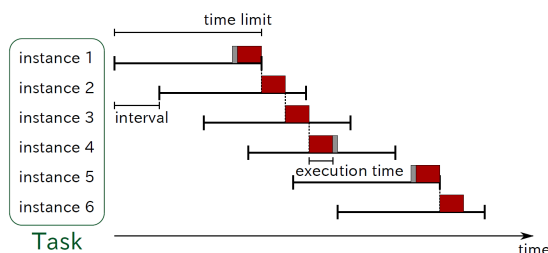


図 2 処理をまとめるタスクスケジューリング⁶⁾
Fig. 2 Task scheduling that packs up the processes⁶⁾

時であっても MCU は省電力モードではあるが電力を消費し続ける．これらの研究に対して，システム全体の省電力化を図る本手法と組み合わせることで，より省電力化が図れると考えられる．特に，本研究のノーマリオフにおいては「CPU の電源も切る」という点がこれらの省電力手法とは異なる点である．

3.2 ノーマリオフを目指した研究

ノーマリオフを対象とした研究として，複数の周期タスクが存在するシステム上で，タスクスケジューリングによる省電力化を図る研究がある（岡本ら⁶⁾）．これは，マイクロプロセッサを備えたセンサであるスマートセンサを対象とした研究である．そして，このようなシステムでは入力データのサンプリング周期とデータ送信（デッドライン）の周期は必ずしも一致せず，後者の周期の方が前者の周期よりはるかに大きい，すなわち時間制約がゆるいという点に着目している．

そこで，システム上の複数のタスクについて，それらの実行順序を調整しアイドル時間をまとめることで，省電力モードへの状態遷移の回数を減らし，状態遷移に伴う電力オーバーヘッドを削減する方法を提案している（図 1，図 2）．

岡本らの手法は本研究においても有効なものであり，本研究と組み合わせることでよりシステムの低消費電力化が期待できる．ただし，岡本らの研究は理論を提案するものであり，その有効性についてはシミュレーション評価を行っている．対して，本研究はノーマリ

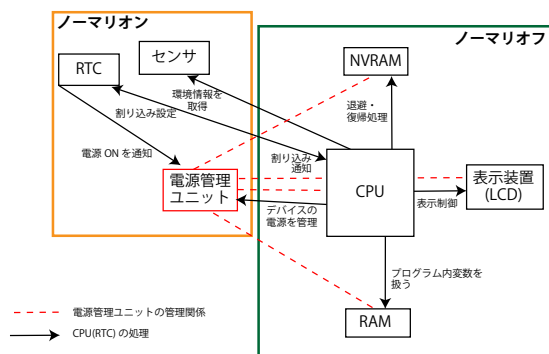


図 3 提案システムの構成
Fig. 3 Proposed system structure

オフを実現するためのハードウェアとソフトウェアの枠組みを提案・実装するものである。特に、ソフトウェアの枠組みについては、OS 上の省電力状態管理の枠組みを提案・実装している。

4. 提案システム

4.1 システム構成

本論文で提案する屋外設置型システムの構成を図 3 に示す。図 3 の各デバイスについて、CPU と RAM はシステムが動作する上で最低限必要なものである。そして、屋外設置型端末の中でも人へ情報を提示するデバイスを対象として扱うことから、表示装置 (LCD) と必要なタイミングで情報を提示するために環境情報を取得するセンサを備えている。

また、図 3 の内、本論文で提案するノーマリオフを実現する際に利用するデバイスは NVRAM と電源管理ユニット、リアルタイムクロック (RTC) である。NVRAM は CPU と RAM の内部情報を退避・復帰するために利用する。電源管理ユニットは CPU と RAM, LCD, NVRAM への電源供給を管理するデバイスで、CPU や RTC から制御する。ノーマリオフによる省電力状態の際には CPU の電源も遮断されるため、定期的にシステムが起床するために RTC を用いる。

なお、提案システムは、ノーマリオフの領域とノーマリオン領域で構成されている。ノーマリオフの領域は、必要なときにのみ電源を投入し、実行すべきタスクの無い待機時には電源を切るデバイスが含まれる。そして、ノーマリオン領域には、常に電源供給がなされる必要のあるデバイスが含まれる。

本研究では、ノーマリオン領域を最低限とし、CPU を含めたシステムの大部分をノーマリオフとするための省電力手法を提案する。

表 1 追加したシステムコール
Table 1 Added system calls

システムコール名	動作概要
saveTASK	タスク資源の退避
loadTASK	タスク資源の復帰
setPowerSavingMode	省電力状態 (ノーマリオフ、スリープモード) への移行

4.2 省電力手法

本論文では、省電力化のための枠組みを「省電力状態管理タスク」というタスクとして実装する。

4.2.1 省電力状態管理タスク

省電力状態管理タスクは、ノーマリオフを含む省電力状態への移行と復帰を管理するタスクで、省電力状態への移行と復帰を行うタイミングでこのタスクが起床する。省電力状態への移行と復帰に関する処理は、デバイスドライバとシステムコールで実装する (表 1)。そして、省電力状態管理タスクは、既存のシステムコールと新たに追加されたシステムコールを利用して省電力方針を記述する。このような構成とすることで、通常のタスクを記述する場合と同様に、省電力状態管理タスクの振る舞いを記述できる。そのため、本手法においては省電力のための設定を行う際にカーネル部の設計についての知識を必要としない。

なお、CPU と RAM の内部情報は OS の管理資源を単位として退避と復帰を行う。対象となる管理資源として、タスクに関しては、タスク管理ブロック (TCB) とタスクスタック領域、静的変数領域などで、その他にはメモリプール領域などがある。退避と復帰のシステムコールは OS の管理資源毎に作成する。

4.2.2 省電力状態管理タスクの振る舞い

省電力状態管理タスクの振る舞いについて図 4, 図 5, 図 6 に示す。それぞれの図は、スケジューラと省電力状態管理タスク、ユーザ設定タスクの関わりを示している。図 4 の「ユーザ設定タスク」は、OS を利用するユーザ (組込システム開発者) が設定するタスクである。このタスクは定期的に行う処理を終えると、システムコールを通じて自タスクをタスク待機状態へ遷移させる。

図 4 では省電力状態管理タスクの基本的なふるまいを示している。ノーマリオフのために必要な退避と復帰の処理は、それぞれ「全てのタスクがタスク待ち状態へ遷移した直後」と「システムの動作再開時」に実行されるべきである。このタイミングを保証するために「省電力状態管理タスク」を平常時は最低優先度としておく。このように実装することで、他の全てのタスクが待ち状態へ遷移した後に退避の処理と電源管理

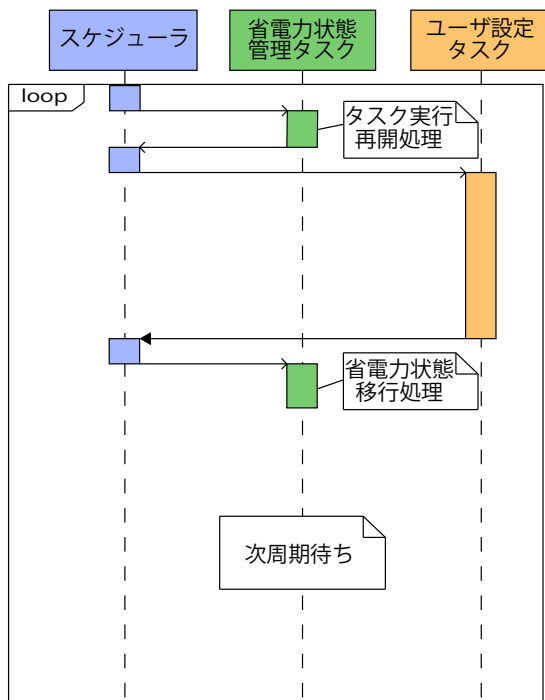


図 4 全体のシーケンス図
Fig. 4 Entire sequence

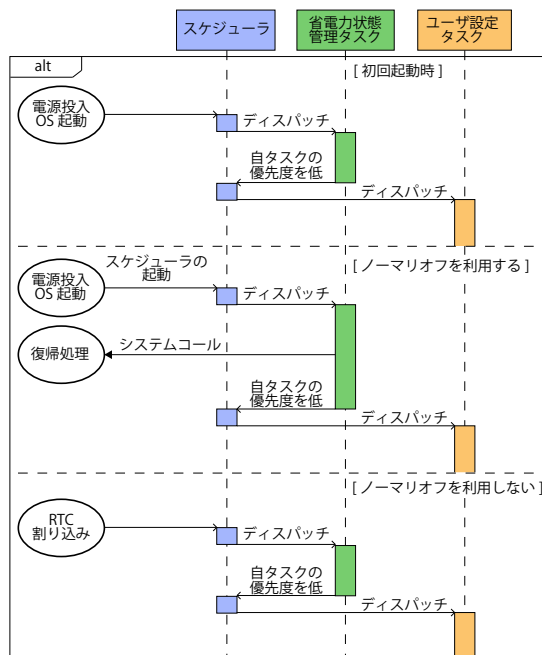


図 5 タスク実行再開処理のシーケンス図
Fig. 5 Sequence of restarting the task processing

ユニットへの電源遮断の通知をすることが可能となる。

また、省電力状態管理タスクが省電力状態への移行で呼び出された際に、省電力状態管理タスクが自身を最高優先度で登録する(図6)。この設定により、電源管理ユニットが電源を投入し、OSが立ち上がった際に最初に省電力状態管理タスクを実行させることができる(図5)。その後、必要であれば復帰の処理を行い、システムコールを通じて自身を最低優先度へ変更する。そして、ユーザ設定タスクの実行が再開する。

シーケンス図に関して図5は、図4で「タスク実行再開処理」と示してある部分について説明している。初回起動時か、あるいはノーマリオフを利用するか否かの3つに処理が分かれている。初回起動時、あるいはノーマリオフを利用しない場合は、システムコールにより自身の優先度を最も低くする処理のみを行う。また、ノーマリオフを利用する場合、すなわち退避済みのデータがある場合は、それらを復帰した後に自身の優先度を変更する。

そして、図6では図4で「省電力状態移行処理」と示してある部分について説明している。ノーマリオフを利用する場合と利用しない場合に分かれている。いずれの場合においても、ユーザ設定タスクがタスク待ち状態へ遷移し、他に実行可能タスクが無いことから、最低優先度で動作している省電力状態管理タスク

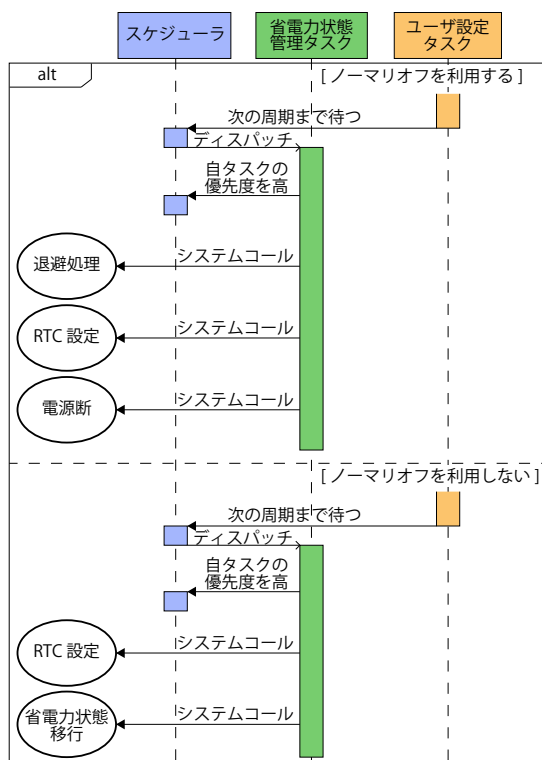


図 6 省電力状態移行処理のシーケンス図
Fig. 6 Sequence of changing to the power-saving mode

ヘディスパッチすると、まず自身のタスク優先度を最も高い状態へ設定する。その後は、ノーマリオフを利用する場合は退避処理と RTC 設定、電源管理ユニットによる電源の遮断をシステムコールを通じて行う。また、ノーマリオフを利用しない場合は、自身のタスク優先度変更の後、RTC 設定と省電力状態移行をシステムコールを通じて行う。ノーマリオフを利用する場合は電源管理ユニットによる電源再投入で復帰することになる。また、ノーマリオフを利用しない場合は CPU の省電力機能を活用するため、一般に割り込み処理で復帰することになる。

このように、他のタスクがタスク待ち状態へ遷移したことをきっかけに動作するため、省電力状態管理タスクの追加に伴うユーザが設定したタスクの修正は不要である。また、本省電力手法はその実行タイミングから、タスク実行周期が長いほど、退避処理後に電源を遮断してから復帰処理を始めるまでの待機時間が長くなる。そのため、タスク実行周期が長いほど省電力効果が高くなる手法である。

4.2.3 省電力状態管理タスク内の設定

省電力状態管理タスクでは省電力状態としてノーマリオフのみではなく、システムが利用できる他の省電力状態への移行と復帰も管理できる枠組みとしている。本論文の実装では CPU のスリープモードも省電力状態として選択可能である。

このような、ノーマリオフを含む省電力状態をタスクとして実装した本省電力状態管理タスクで記述可能な設定としては、損益分岐による自律的な省電力状態の選択がある。この設定は、すべてのタスクがタスク待ち状態へ遷移し、実行すべきタスクがシステム上に無いアイドル時間の長さに応じて移行する省電力状態を自律的に切り替える設定である。

自律動作するバス停や情報表示端末など屋外設置型の端末では、システム上で複数の周期タスクが動作することが想定される。そのため、アイドル時間の長さにも複数のバリエーションが存在することになる。省電力状態としてノーマリオフを利用する場合は OS 管理資源の退避と復帰のための時間と電力を消費するため、アイドル時間の長さによってはそのオーバーヘッドに見合った電力削減効果を得られない場合がある。そのため、ノーマリオフを利用するか否かの損益分岐点となるアイドル時間の長さを本タスク内には設定できる。なお、この設定は本論文で実装した実験でも活用されている。

また、退避と復帰に関わる OS 資源についても、省電力状態管理タスク内で設定可能である。退避と復帰

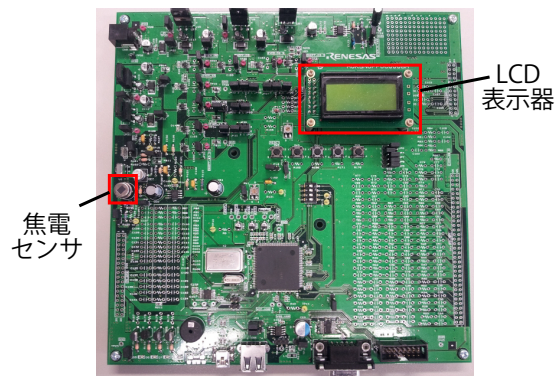


図 7 RX63N 評価ボード
Fig. 7 RX63N evaluation board

表 2 RX63N 評価ボードの構成
Table 2 Description of RX63N evaluation board

項目名	メーカー名 (型番)
	ルネサスエレクトロニクス 株式会社
CPU	(R5F563NEDDFC) FORDATA ELECTRONIC CO., LTD
LCD	(FDCC0802B) 日本セラムック株式会社
焦電センサ	(RE-200B)

に関わる OS 資源の数は、NVRAM への書き込みと NVRAM からの読み出しにかかる電力消費に繋がるオーバーヘッドである。本設定を用いて、退避と復帰すべき情報を持たない OS 資源は、退避と復帰の対象としない設定とすることで、電力オーバーヘッドを削減できる。この設定も本論文の実験で使用している。

5. 実 験

提案システムの章で述べたシステム構成と省電力手法について、その振る舞いと電力消費削減量について評価する。

5.1 実験環境

実験に際して、RX63N の CPU を搭載した評価ボードを利用した (図 7)。評価ボードの構成を表 2 に示す。この評価ボードは各種センサや表示器を備えており、電力評価の為に評価を行わないデバイスの電源をジャンパピンにより遮断できる。表 2 には本実験においてジャンパピンで有効化したものを挙げている。今回は CPU の他には焦電センサとキャラクタタイプ LCD (2 行 8 桁) のみ電源をとり、他は遮断した。なお、退避と復帰の領域には RX63N に内蔵されているデータフラッシュ領域を用いた。

また、本評価ボードは複数の電源系統があり、一部の電源系統は CPU から汎用ポートを通じて制御が可能

表 3 RX63N 評価ボードの電源系統

Table 3 Power controlling of RX63N evaluation board

電源系統名	電源供給先
常時電源 (デジタル 3.3V) ON/OFF 電源	RX63N デジタル部
(デジタル 3.3V) 常時電源 (アナログ 3.3V) ON/OFF 電源	LCD (データピン用)
(アナログ 3.3V) ON/OFF 電源	RX63N AD 電源
(アナログ 3.3V) ON/OFF 電源	焦電センサ
(デジタル 5V)	LCD 電源

表 4 タスク設定

Table 4 Task settings

タスク名	動作周期 [秒]	起床条件
焦電センサタスク	1/2	省電力状態管理タスク
LCD タスク	1/64	焦電センサタスク

である。電源系統とその電源供給先との対応を表 3 に示す。表 3 に関して「常時電源」の系統はシステムの電源 ON の際には常に供給される電源で「ON/OFF 電源」の系統は CPU から汎用ポートを通じて ON/OFF の制御が可能な電源である。

今回利用する評価ボードにおいては CPU の電源自体は常時電源であるため、ノーマリオフ利用時の低消費電力状態として RX63N のディープソフトウェアスタンバイモードを利用している。これは、CPU、内蔵周辺機能、RAM の一部分、および発振器の全ての機能を停止し、さらにこれらの内部電源の供給を停止するモードである⁷⁾。ただし、内蔵周辺機能に関しては、RTC などディープソフトウェアスタンバイモードの解除の際に用いる一部の機能に関しては電源は動作を続ける。また、RAM に関しては、レジスタ設定により全ての領域の電源を遮断することができ、本実験においても全ての RAM の電源は遮断される。

なお、評価ボードの都合上、電流の計測を行う地点に 2 つの外部オシレータの電流も含まれてしまうため、10~15mA ほど余分に計測されている。

本実験においては TOPPERS/ASP カーネル⁸⁾を対象に省電力手法を実装した。スケジューラが優先度方式であること、またカーネル部でも 13000 行ほどと扱いやすいことから、今回はこれを用いた。

5.2 タスク設定

表 4 に示す 2 つのタスクを作成して、実験を行った。これらのタスクは「人を感じると LCD 表示器の電源を投入して日時を表示する」という振る舞いを実現する。

「焦電センサタスク」は 0.5 秒周期で動作する。こ

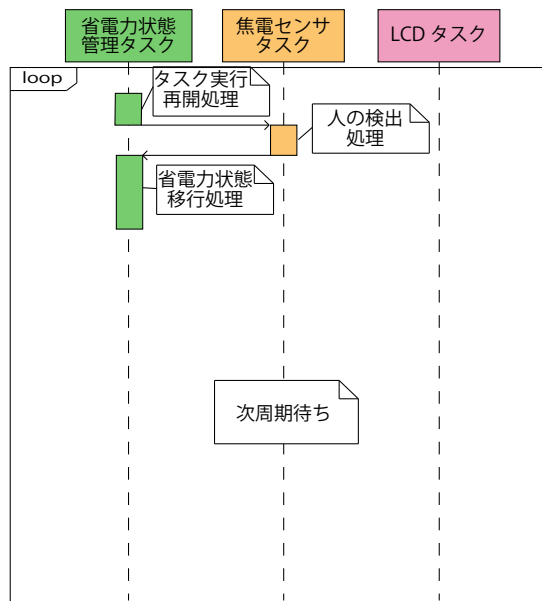


図 8 2 つのタスクに関するシーケンス図 (人検出待ち)

Fig. 8 Sequence about 2 tasks (Waiting for the detection of human)

のタスクは、焦電センサにより人が居ることをを検知すると LCD タスクを待ち状態から起床させ、設定された時間だけ LCD を表示させる。今回は 5 秒を設定しており、焦電センサタスクは人を検知すると 0.5 秒周期で 5 秒間 LCD タスクを繰り返し起床させる。

また、「LCD タスク」は、LCD 表示器への情報表示を担うタスクである。LCD の表示を更新する際に LCD タスクは LCD の仕様により、1/64 秒 (約 16ms) の周期で制御信号を LCD へ送信する。今回使用する LCD 表示器は 2 行 8 桁 (合計 16 文字) の表示領域を持ち、行を選択する信号も別途送る必要があるため、合計 18 回の LCD 制御を行う。なお、表示内容は時計で、年月日と曜日、そして時分秒を表示するため、更新の周期は 1 秒である。そのため、焦電センサタスクが LCD タスクを呼び出しても、情報更新を行う必要が無い場合は何も行わずに LCD タスクは待ち状態へ遷移する。

ここで、表 4 のタスクの振る舞いについて、シーケンス図を図 8 と図 9 に示す。図 8 は焦電センサタスクが人の検出を待っている状態を表しており、図 9 は焦電センサタスクで人を検出した場合に、LCD タスクで LCD に表示を行う様子を表している。なお、図 8 と図 9 は共に簡単に為にスケジューラは省いて示している。

5.3 省電力状態のしきい値設定

今回の実験においては、焦電センサタスクと LCD タ

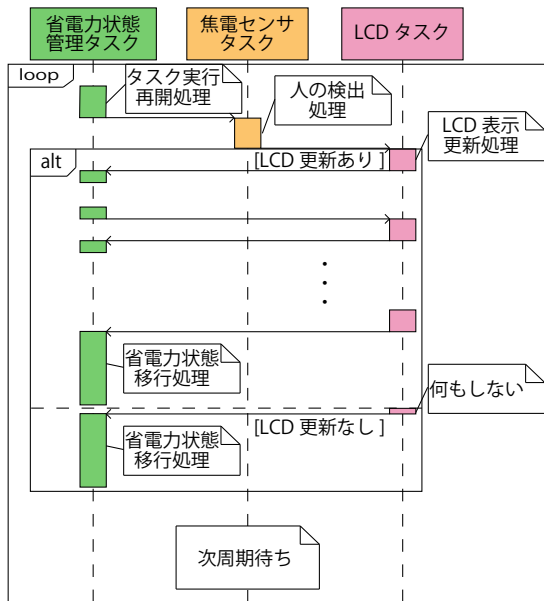


図 9 2つのタスクに関するシーケンス図 (LCD 表示中)
Fig. 9 Sequence about 2 tasks (Displaying on the LCD)

表 5 各状態の電力消費の平均と時間

Table 5 Average current and the time of each state of the Normally-off

項目名	変数	値	単位
消費電流 (ノーマリオフ時)	I_N	10	mA
消費電流 (スリープ時)	I_S	32	mA
消費電流 (ノーマリオフ退避時)	I_{save}	46	mA
時間 (ノーマリオフ退避時)	T_{save}	56	ms
消費電流 (ノーマリオフ復帰時)	I_{load}	39	mA
時間 (ノーマリオフ復帰時)	T_{load}	0.3	ms

スクが共にタスク待ち状態となることで、システムが完全に待機状態となる。タスク設定の関係から、この待機状態の長さにはいくつかのパターンが存在する。そこで、ノーマリオフを利用するか否かを、2つのタスクが共にタスク待ち状態である時間の長さで分ける。そのしきい値となる時間を設定するため、表5に示すパラメータを事前に計測した。なお、以降の実験も含めて計測には Tektronix 社の MSO4104 オシロスコープを用いた。また、CPU の省電力状態にはスリープモードを利用する。

表5からスリープではなくノーマリオフを選択すべきしきい値を概算する。表5のパラメータから、待機時間 t におけるノーマリオフ時の消費電力量 E_N (単位: mAms[ミリアンペア・ミリ秒]) を式1のように表せる。

$$E_N = I_N(t - T_{save} - T_{load}) + I_{save}T_{save} + I_{load}T_{load} \quad (t \geq T_{save} + T_{load}) \quad (1)$$

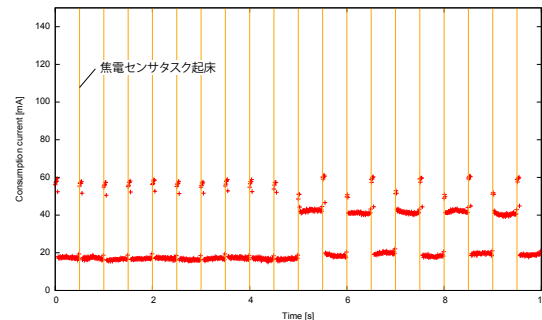


図 10 ノーマリオフ機能有効時の 10 秒間の計測結果
Fig. 10 Measuring result for 10 seconds when the normally-off function was enabled

また、同じく待機時間 t におけるスリープ時の消費電力量 E_S (単位: mAms) を式2のように表せる。

$$E_S = I_S t \quad (2)$$

$$t = \frac{(I_{save} - I_N)T_{save} + (I_{load} - I_N)T_{load}}{I_S - I_N} \quad (3)$$

式1と式2を連立させると式3が得られる。これに表5のパラメータを代入すると、 t は約 92ms であることが分かる。この結果から、今回の実験においてはノーマリオフを利用するしきい値を 100ms と定めた。

なお、LCD 表示中はディープソフトウェアスタンバイモードへ移行しない。LCD 表示期間が終了し、ディープソフトウェアスタンバイモードへ移行する際には LCD タスクに保持しておくべき情報は無いため、今回の実験では省電力状態管理タスク内へ LCD タスクはノーマリオフのための内部情報の退避と復帰の対象とはしない設定を行った。従って、しきい値によりノーマリオフ選択時に退避と復帰の対象となる OS の管理資源は焦電センサタスクのタスク資源のみである。

5.4 結果

「最初の 5 秒間は人の検出を待ち、次の 5 秒間は LCD を表示する」という状況で計測を行った。結果を図10に示す。また、比較対象として「スリープのみ利用する場合」と「アイドルループのみ利用する場合」についても計測を行った。スリープモードのみ利用時の結果を図11に、アイドルループのみ利用時の結果を図12に示す。

なお、「アイドルループのみ利用する場合」は、システム上の全てのタスクがタスク待ち状態へ遷移しシステムがアイドル状態となった際に、次のタスク実行までアイドルループ(フリーラン)で待機する状態である。この場合は、いかなる省電力状態も利用しない。オシロスコープでは 0.1ms 毎のデータを取得したが、図10、図11、図12では見やすさの為に 10ms 毎

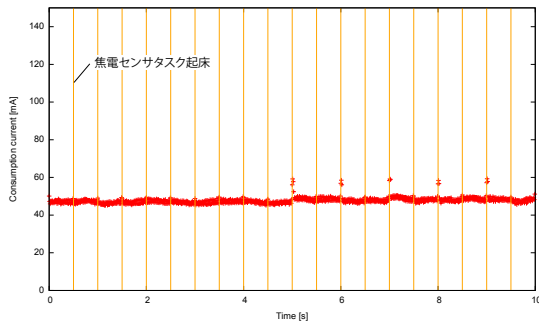


図 11 スリープのみ利用時の 10 秒間の計測結果
Fig. 11 Measuring result for 10 seconds when the only sleep mode was enabled

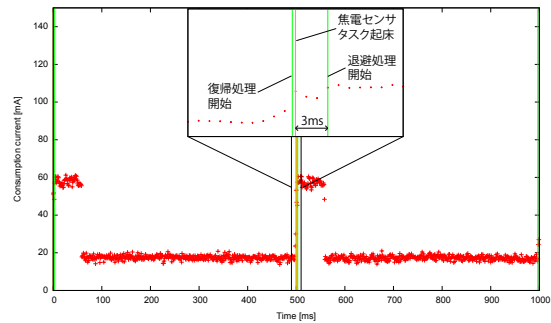


図 13 ノーマリオフ時の人検出待ち状態
Fig. 13 Waiting for the detection of human when the normally-off was enabled

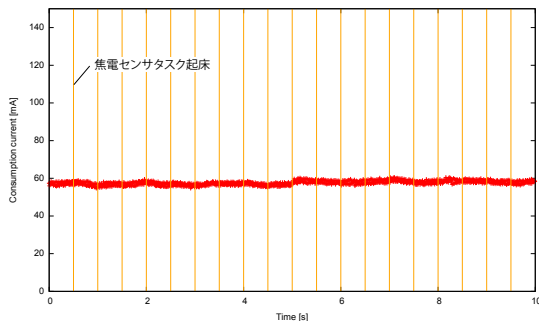


図 12 アイドルループのみ利用時の 10 秒間の計測結果
Fig. 12 Measuring result for 10 seconds when the only idle loop was enabled

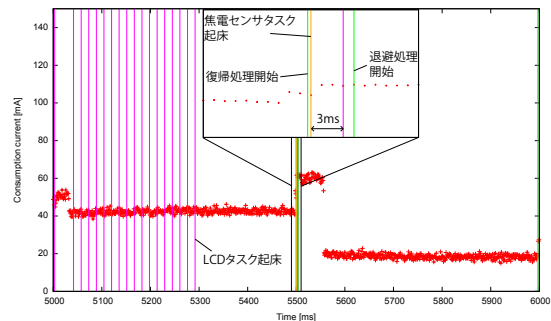


図 14 ノーマリオフ時の LCD 表示状態
Fig. 14 Displaying on the LCD when the normally-off was enabled

に平均をとった結果をグラフ化している。また、グラフ中のオレンジの縦線は焦電センサタスクの起床を表しており、ノーマリオフ利用時（図 10）は、焦電センサタスクの実行周期を乱すことなく、積極的に省電力化を図っている事が分かる。

局所的な振る舞いとして、焦電センサタスクのみ動作している状態（10 秒間のグラフの 0～1 秒目）と LCD タスクが動作している状態（10 秒間のグラフの 5～6 秒目）を、ノーマリオフ利用時について図 13 と図 14 に、スリープのみ利用時について図 15 と図 16 に、アイドルループのみ利用時について図 17 と図 18 に示す。

オレンジの縦線は 10 秒間のグラフ同様、焦電センサタスクの起床を表す。そして、ピンクの縦線は LCD タスクの起床を表し、緑の縦線はノーマリオフ時の退避と復帰の開始を表す。図 13 と図 14 に関して、省電力状態管理タスクが起床してからノーマリオフの省電力状態が実施されるまでの電力消費はそのほとんどがデータフラッシュへの書き込みの電力消費である。

図 13 や図 14 を見ると、復帰にかかる時間は退避に

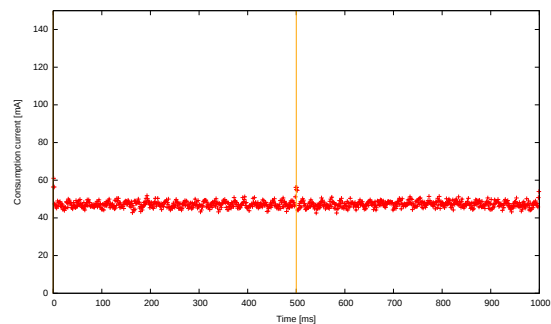


図 15 スリープのみ利用時の人検出待ち状態
Fig. 15 Waiting for the detection of human when the only sleep mode was enabled

比べ短いことから、ノーマリオフ利用に伴う電力オーバーヘッドのほとんどは退避の際に発生することがわかる。また、図 14 と図 16、図 18 に関して、LCD 更新時（5000ms～5500ms）の LCD タスク周期実行において、初回の LCD タスクでは LCD 初期化のために 1/64 秒の周期を超えて長く実行されている。以降の LCD コマンド実行は 0.1ms 以下で完了していること

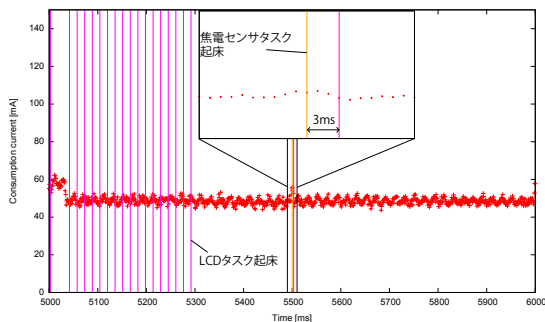


図 16 スリープのみ利用時の LCD 表示状態

Fig. 16 Displaying on the LCD when the only sleep mode was enabled

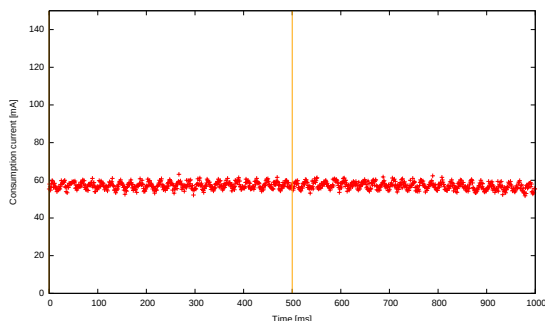


図 17 アイドルループのみ利用時の人検出待ち状態

Fig. 17 Waiting for the detection of human when the only idle loop was enabled

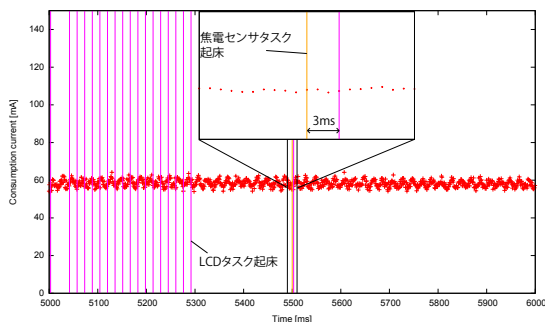


図 18 アイドルループのみ利用時の LCD 表示状態

Fig. 18 Displaying on the LCD when the only idle loop was enabled

から図 14 では常にスリープであるように見えている。

以上により、提案システムで示したノーマリオフの枠組みが正しく機能していることが確認できた。特に、LCD 更新中の LCD タスク間はしきい値によりノーマリオフではなくスリープモードを自律的に選択していることが確認できた。

ノーマリオフ機能利用時、スリープモードのみ利用

表 6 平均消費電流
Table 6 Average current

省電力状態	区間	値	単位
ノーマリオフ機能利用時	10 秒間全体	27	mA
	人検出待ち状態	22	mA
	LCD 表示状態	33	mA
スリープのみ利用時	10 秒間全体	48	mA
	人検出待ち状態	47	mA
	LCD 表示状態	49	mA
アイドルループのみ利用時	10 秒間全体	58	mA
	人検出待ち状態	57	mA
	LCD 表示状態	58	mA

表 7 消費電流の削減率

Table 7 Rate of consumption current reduction

省電力状態	区間	削減率 [%]
ノーマリオフ機能利用時	10 秒間全体	53
	人検出待ち状態	61
	LCD 表示状態	43
スリープのみ利用時	10 秒間全体	17
	人検出待ち状態	18
	LCD 表示状態	16

時、アイドルループのみ利用時の 3 つに対して、平均消費電流を算出した。結果を表 6 に示す。

表 6 の結果から、ノーマリオフ機能利用時とアイドルループのみ利用時を比較すると、ノーマリオフ機能利用時はアイドルループのみ利用時に比べ消費電流を約半分に抑えられていることが分かった。また、ノーマリオフ機能利用時とスリープのみ利用時を比較した場合も同様に、ノーマリオフ機能利用時は消費電流を約半分に抑えられていた。以上の結果から、本論文で提案するノーマリオフの枠組みが消費電力低減に寄与することが確認できた。

6. 考 察

6.1 省電力性能の評価

実験結果の表 6 から電力消費の削減率を評価した。削減率はアイドルループのみを利用した場合を基準として式 4 のように定義する。評価結果を表 7 に示す。

$$\text{削減率} = \left(1 - \frac{\text{省電力状態の消費電流}}{\text{アイドルループ時の消費電流}} \right) \times 100 \quad (4)$$

表 7 の 10 秒間全体の結果に関して、スリープのみの利用時は削減率が 17% であるのに対して、ノーマリオフ利用時には 53% とスリープのみ利用時の 2 倍以上の削減率であることが確認できた。

また、表 7 のスリープのみ利用時の LCD 表示状態と人検出待ち状態の差が 2% であるのに対し、ノーマリオフ利用時はその差は 18% と、スリープに比べ 10

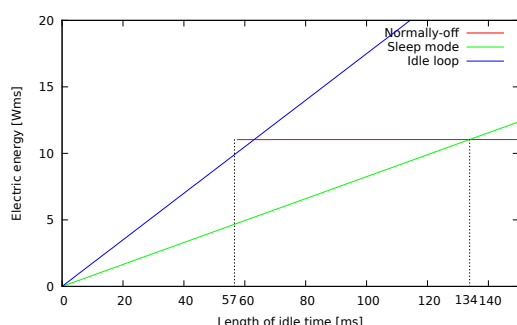


図 19 省電力状態の損益分岐
Fig. 19 BEP of each power-saving mode

倍近く離れている。これは、何もするべき作業の無い待機時間が長くなればなるほどノーマリオフの方が省電力効果が高まることを示しており、ノーマリオフの本質的な振る舞いを実現できていることが確認できた。

6.2 しきい値の設定について

本論文の実験では、省電力状態管理タスク内に設定するしきい値を計算式 3 を用いて概算した。このようなしきい値の評価方法は計算の手間の他に、事前にパラメータとなる消費電流や時間を計測する手間がかかる。

そこで、事前評価の方法として、データシートからのしきい値の算出を行った。図 19 に損益分岐のグラフを示す。図 19 では横軸にアイドル時間の長さ（単位：ms）、縦軸に消費電力量（単位：Wms[ワットミリ秒]）で表している。グラフ中の 3 つの線はアイドル時間の長さに対して、その時間をそれぞれの省電力状態で過ごした場合の消費電力量を表している。赤色がノーマリオフを利用した場合、緑色がスリープモードを利用した場合、青色がアイドルループの場合である。この場合はしきい値となる損益分岐点が 134ms であることが分かる。なお、ノーマリオフの場合の赤色の線は 57ms の時点から開始している。これは、ノーマリオフを利用するためのオーバーヘッドとなる退避と復帰の処理にかかる時間から、これより短い長さのアイドル時間ではノーマリオフを利用できないためである。

7. ま と め

本論文では、屋外設置型のシステムに対してノーマリオフ化による省電力化を図った。ノーマリオフを実現するためのハードウェア構成として電源管理ユニット、外部の RTC、揮発性デバイスの内部情報を退避/復帰するための NVRAM を搭載した枠組みを提案した。また、OS の枠組みとして、ノーマリオフのため

の機能をタスクとして実装する手法を提案した。

TOPPERS/ASP カーネルに提案手法を実装し、評価ボードを用いて消費電力の評価実験を行った。結果から、ノーマリオフ機能利用時には CPU のスリープモードのみを利用した場合に比べて消費電流が半分に抑えることができた。電力消費の削減率に関して、スリープモードのみでは 17% である所をノーマリオフ利用時には 53% まで削減率を高めることができた。

8. 謝 辞

本研究は、ルネサスエレクトロニクス株式会社による受託研究の一環として行った。

参 考 文 献

- 1) 林越正紀, 清水徹, 松原仁: ノーマリオフコンピューティング: 2. センサ端末におけるノーマリオフコンピューティング, 情報処理, Vol. 54, No. 7, pp. 661-667 (2013).
- 2) 安藤功児: 3. 不揮発性デバイス: ノーマリオフコンピュータは実現できるか (特集) エレクトロニクスの多様化を支える新デバイス技術-2020 年を見据えて-, 電子情報通信学会誌, Vol. 93, No. 11, pp. 913-917 (2010).
- 3) 木村一樹, 近藤正章, 天野英晴, 宇佐美公良, 中村宏, 佐藤未来子, 並木美太郎: コア温度情報を用いた OS による細粒度パワーゲーティング制御方式の設計, 技術報告 7, 東京農工大学, 電気通信大学, 慶應義塾大学, 芝浦工業大学, 東京大学, 東京農工大学, 東京農工大学 (2011).
- 4) 宮川大輔, 石川裕: 電力制御スケジューラのプロトタイプ実装, 技術報告 86(2006-OS-103), 東京大学, 東京大学 (2006).
- 5) 安部恵一, 水野忠則, 峰野博史: 無線センサネットワーク向けの省電力型無線ノードの開発, 技術報告 5, 静岡大学創造科学技術大学院 / 浜松職業能力開発短期大学校, 愛知工業大学, 静岡大学創造科学技術大学院 (2011).
- 6) 岡本和也, 薦田登志矢, 中田尚, 三輪忍, 佐藤洋平, 植木浩, 林越正紀, 清水徹, 中村宏: 周期実行システムにおける省電力スケジューリングの初期検討, 技術報告 4, 東京大学, 東京大学, 東京大学, 東京大学, ルネサスエレクトロニクス株式会社, ルネサスエレクトロニクス株式会社, ルネサスエレクトロニクス株式会社, 東京大学 (2012).
- 7) ルネサスエレクトロニクス株式会社: RX63N グループ, RX631 グループ ユーザーズマニュアル ハードウェア編 (2013).
- 8) TOPPERS プロジェクト: TOPPERS プロジェクト/ASP カーネル. (最終アクセス 2013 年 6 月 30 日).