An Open Source Intelligent Auto-Wakeup Solar Energy Harvesting System for Supercapacitor-Based Energy Buffering

Abstract:

Energy harvesting systems that couple solar panels with supercapacitor buffers offer an attractive option for powering computational systems deployed in field settings, where power infrastructure is inaccessible. Supercapacitors offer a particularly compelling advantage over electrochemical batteries for such settings because of their ability to survive many more charge–discharge cycles. We share a versatile open source design for such a harvesting system that targets embedded system applications requiring power in the 1–10 W range. Our system is designed for high efficiency and controllability and, importantly, supports auto-wakeup from a state of complete energy depletion.

Existing system:

Low-power (10–100 mW) distributed sensing and communication devices, such as those used in wireless sensor networks (WSNs), already make use of energy harvesting relying on energy sources such as RF (radio-frequency), vibration, and solar radiation. Low-power sensing and communication platforms utilize low-
complexity energy harvesting circuits and methods, such as direct connection of the energy source and buffer and harvesters built upon passive circuit components.

**Disadvantage:**

- The harvesting efficiency of these systems range from 30–65%.
- Since the availability of the environmental energy is intermittent, energy buffering also necessary with harvesting.

**Block Diagram:**
Proposed system:
The control module consists of the µC and the firmware that is loaded in its flash ROM. The input/output (I/O) signals that allow the firmware to interface with the hardware components are shown in Fig and consist of four groups:

Measurement: Solar panel voltage, solar panel current, supercapacitor block voltage, supercapacitor input current, and supercapacitor output current. The firmware accesses these values from the measurement module via the µC’s ADC.

Voltage Domains: The voltage domain signal pins allow the microcontroller to enable/disable the voltage domains feeding the computational device, Bluetooth module, and RS-232 level converter, respectively.

Communication: The TX and RX signals transmit and receive data from either Bluetooth or RS232 communication devices using the RS-232 protocol, based on the user’s selection.

Harvester: The pulse width modulation (PWM) signal from the µC controls the MOSFET switch of the harvester. Since the current drive capability of the PWM pin is not sufficient (25 mA) to drive the gate of the MOSFET directly, a gate driver is used as a buffer, which allows a drive current of 2 A. These control signals provide the interface between the firmware in the control module and the hardware components in the other modules. They allow the control algorithms to be implemented in firmware and eliminate the need for hardware modifications when slight adjustments need to be made to the algorithms.
Advantages:

- The ability to maintain sustained operation over a two week period when the solar panel and buffer are sized appropriately.
- A robust auto wakeup functionality that resume system operation upon the availability of harvestable energy.

Conclusion:

In this paper, an open-source energy harvesting system is presented, which uses solar panels as its sole energy input and super capacitors as its sole energy buffer. The system is able to harvest a maximum solar power of 15 W and provide a regulated 5 V voltage to an external embedded device (termed computational device throughout the paper) that has a maximum power consumption of 10W. Designed to operate in harsh environmental conditions where the solar energy might be absent for extended periods of time, the system is able to wake up and resume functionality from a fully depleted state, when the super capacitors have zero remaining energy. During its normal operation, the system uses its built-in RS-232 or Bluetooth communication capability to transmit vital energy-state information to the external computational devices.
Such information includes the solar voltage, super capacitor block voltage, solar current, and supercapacitor charge/discharge currents. Using this information, the embedded device could make software-level decisions to maximize its energy efficiency by intelligently using different software components, corresponding to different energy consumption levels.

Reference:


