Sunny with a Chance of Servos: Solar-Powered Arduinos

Tim Gfroerer, Michael Adenew, and Ella Williams, Davidson College, Davidson, NC

The Arduino microcontroller is finding its way into labs throughout undergraduate physics curricula, from introductory courses[1,2] to a variety of beyond-the-first-year[3,4] laboratory classes. At Davidson College, we use Arduinos in a gateway STEM course for students who are interested in energy and the environment. Students learn to build simple circuits and write the accompanying Arduino code to control the temperature in solar-powered model buildings. To make the models fully solar powered, the Arduino itself must be powered by the Sun—no batteries allowed! Hence, we replace the 9-V battery that is usually used to power an Arduino with a 13.5 cm × 12.5 cm 9-V solar panel (DFRobot part #FIT0330), which can generate a maximum short-circuit current of approximately 200 mA. We find that the solar panel works well for most tasks, including temperature measurements, liquid-crystal display (LCD) illumination, and SD card module operation, but cannot generate enough power to drive a servo motor, which needs several hundred milliamps. For these situations, a 9-V, 1-F capacitor[5] connected in parallel with the solar panel can store energy during the rest period between brief high-current operations and supplement the solar panel when higher power is required.

A representative *I–V* characteristic of the solar panel is shown in Fig. 1. This solar panel is composed of thirty-six 2.5 cm × 1.2 cm 0.5-V solar cells with parallel pairs connected in series so that the current generation is doubled (relative to a single cell) and the net output voltage is 18 × 0.5 V = 9 V.[6] The panel generates a maximum power of 1.3 W near the knee of the *I–V* curve. Since the individual cell area is 3×10^{-4} m^2, the total device area is 0.011 m^2. For an average solar flux of 950 W/m^2, the incident solar power equals 10 W and the power conversion efficiency is 1.3 W/10 W = 13%.

Pictorial and circuit diagrams of our experimental setup are shown in Fig. 2. When building the circuit, it is important to ascertain the polarity of the solar panel and ensure that it aligns with the polarity of the capacitor. These devices should be connected to the Arduino DC power input V_{in}, perhaps via screw terminals on a 9-V barrel jack adapter (like DFRobot part #FIT0110), and should never be connected to the 3.3-V or 5-V power pins on the Arduino board. The analog input pins on the Arduino itself can be used to measure the Arduino current and voltage,[7–9] but we seek to obtain independent measurements that do not use additional Arduino power. Hence, we measure the Arduino current and voltage at a sample rate of 20 Hz using Pasco probes and software. While a bare Arduino Uno draws approximately 45 mA of steady-state current, all experiments are conducted with an active LCD keypad shield (DFRobot part #DFR0009, backlight on) mounted on the Arduino, which draws approximately 20 mA of additional current. We use the DFRobot LCD Keypad shield for user-friendly input/output because it comes preassembled, fits directly onto the Arduino board with no need for additional parts, and provides electrical feedthroughs for all unused pins. For solar testing, we maximize the measured current to ensure that the solar panel is oriented directly facing the Sun, and we use an Apogee SP-110 pyranometer

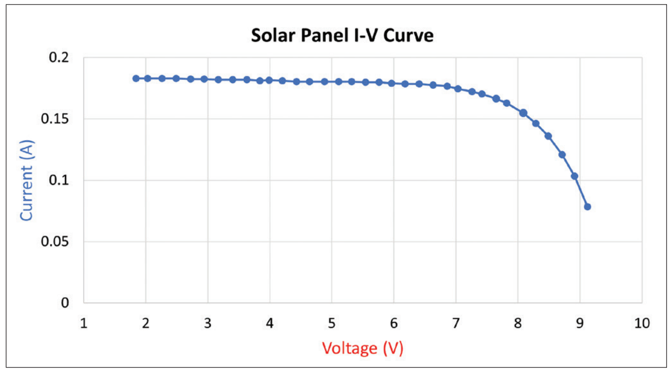

Fig. 1. Current generation vs. output voltage from the solar panel under nominal full sun, measured using a 100-Ω potentiometer. See Ref. 6 for more details on measuring solar panel *I–V* curves.

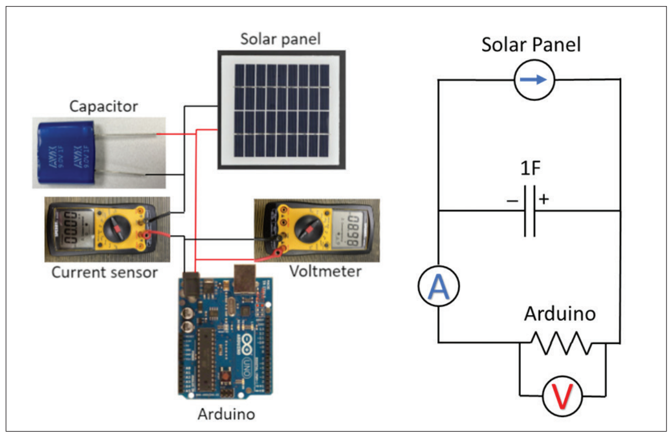

Fig. 2. Experimental setup along with the circuit diagram showing the capacitor and Arduino in parallel with the solar panel and the configuration of the ammeter and voltmeter for measurements of *I* and *V*.

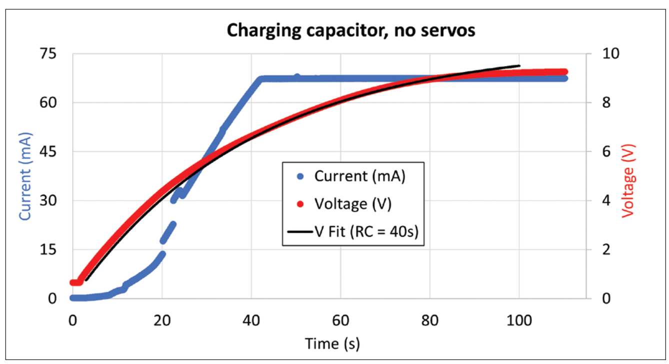

Fig. 3. Arduino current (blue) and voltage (red) when the capacitor is initially uncharged and the solar panel is illuminated by sunlight. The solid black line is a fit to the voltage data using Eq. (1) with *RC* = 40 s.

 DOI: 10.1119/5.0065597

to monitor the solar flux, which generally varied between 900 and 1000 W/m^2 depending on the presence of thin high clouds.

With sunlight and no servo operation, the solar panel charges the capacitor while powering the Arduino (see Fig. 3). The Arduino starts up when the voltage reaches approximately 5 V, resulting in discontinuities in the Arduino current. When the voltage reaches approximately 7 V, the Arduino current levels off, but the capacitor continues to charge. This 7-V threshold can be attributed to the combination of a 5-V voltage regulator and a reverse voltage protection diode on the Arduino external power input. The regulator has a dropout voltage of approximately 1 V, and we lose another ~0.7 V across the silicon diode, so the Arduino manufacturer recommends a minimum supply voltage of 7 V: "The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable."[10]

Since the solar panel, capacitor, and Arduino are connected in parallel, the instantaneous potential drop $V(t)$ across all three devices must be the same. In this situation, where the voltage is changing with time, the solar panel acts more like a current source than a voltage source. From Fig. 1, we see that the current generation holds steady up to about 7 V, and then the current starts to drop off. Analogous to the capacitor charging voltage in a series resistor–capacitor (RC) circuit with a constant voltage source, the charging voltage in a parallel RC circuit with a constant current source is expected to vary as

$$V(t) = V_0(1 - e^{-t/RC}), \quad (1)$$

where V_0 is the maximum voltage of the current source, R is the resistance, C is the capacitance, and t is time. Fitting our results to this expression (see Fig. 3), we obtain an RC time constant of approximately 40 s, indicating that the effective resistance R of the Arduino is about 40 Ω. This resistance is a good match for the solar panel, which generates maximum power with a 47-Ω load.[6] In midday full sun, the solar panel can reach an output of nearly 10 V. Hence, when fully charged by the solar panel, the energy stored in the capacitor is given by $(1/2)CV^2 = (1/2)(1\text{ F})(10\text{ V})^2 = 50$ J. While exceeding the voltage rating on a capacitor is generally ill advised, we did not observe any evidence of overheating, and experiments suggest that even drastically overcharged supercapacitors are unlikely to autoignite.[11]

If the temperature gets too high in our solar-powered model buildings, we use servo motors (DFRobot part #SER0006) to close window shades or open flaps for ventilation. These low-cost 9-g 180° micro servos are specified to provide up to 1.6 kg · cm of torque at a working current of less than 500 mA. Current and voltage measurements during solar-powered servo function with and without a capacitor are shown in Fig. 4. For this experiment, the Arduino code instructs the servo to rotate 180° once every 10 s. A single servo operation requires about 0.5 J of energy in approximately 0.3 s, which is more than the solar panel alone can provide. Hence, without a capacitor, the voltage falls sharply as the circuit tries to draw more current in accordance with the $I–V$ characteristic shown in Fig. 1. In this failure mode, the LCD malfunctions, and the servo continues to attempt to function repeatedly without success. However, with a capacitor to supplement power from the solar panel, the servo operates correctly as the Arduino draws the necessary current from the solar panel and the capacitor together.

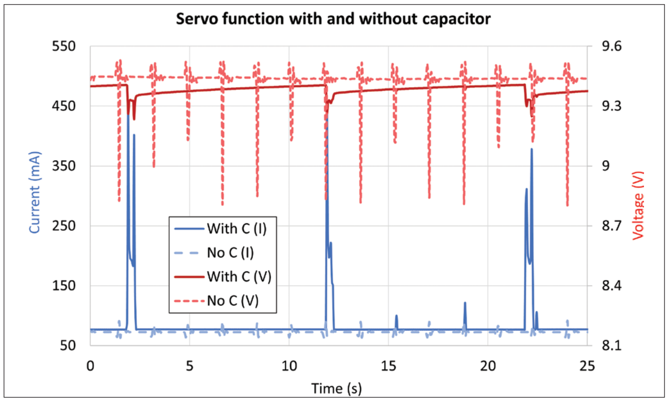

Fig. 4. Solar-powered Arduino current (blue) and voltage (red) with periodic calls for servo rotation. A capacitor is required to provide sufficient current for functional servo operation.

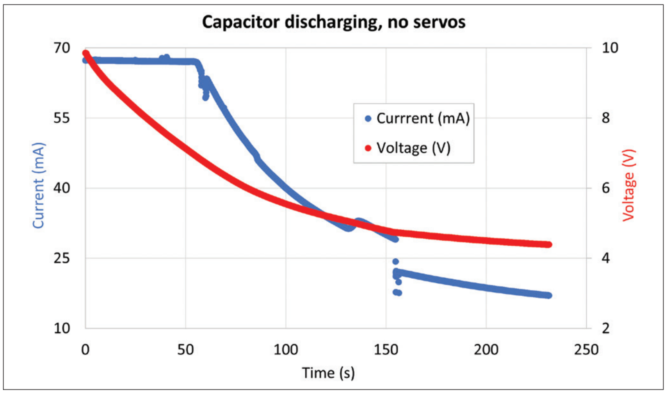

Fig. 5. Arduino current (blue) and voltage (red) when the capacitor is initially fully charged and illumination of the solar panel is blocked. The Arduino continues to operate for over 100 s without sunlight.

When our model buildings are deployed, solar power can be interrupted by shading from unsuspecting visitors or cloud cover. We encourage fellow students and faculty to stop by and see our work, but their shadows are not welcome on our solar-powered buildings! Fortunately, in the absence of sunlight, the capacitor can also sustain power to the Arduino. We have tested this scenario with a fully charged capacitor by blocking the sunlight on the solar panel (see Fig. 5). Under variable cloud cover, we note that the charge state of the capacitor can vary over time, but the voltage can be used to monitor the system and determine when the capacitor is fully charged. Even though the voltage starts to fall immediately, the current delivered to the Arduino holds steady for approximately 1 min. Then the current from the capacitor begins to drop, and when the voltage falls below 5 V, the Arduino powers down. The total energy supplied by the discharging capacitor can be calculated by integrating the instantaneous power $P = IV$. We find that $\int \text{Power}(t)dt = 52$ J, in relatively good agreement with our estimate of the total energy stored. Since the tolerance on the capacitance rating is 30%, it is possible that the actual capacitance of the device is as high as 1.3 F, yielding a maximum energy storage of 65 J.

In summary, we find that the DFRobot FIT0330 solar panel is a good match for the electrical requirements of the Arduino Uno. When a 9-V, 1-F capacitor is connected in parallel with the solar panel, our servo motors remain operational, and the Arduino can function for over a minute without solar panel illumination. Within specified tolerances, we also find that the energy supplied during a solar shading event is consistent with the energy stored in the capacitor. Hence, even if it is partly cloudy with intermittent servo operation, it can still be a great day to work outside! More generally, we hope this investigation will facilitate other solar-powered Arduino applications as we look ahead to a brighter energy future.

Acknowledgment

This course was inspired by a similar offering at Smith College taught by Achilles Professor of Engineering Susan Voss. Our work at Davidson College was supported by the Research in Science Experience Program, a four-week intensive summer research internship program in the sciences intended to introduce students from groups historically underrepresented in science to undergraduate research.

References

1. Amtul Mujeeb Chaudry, "Using Arduino Uno microcontroller to create interest in physics," *Phys. Teach*. **58**, 418–421 (2020).
2. Matthew J. Farrar, "Arduino microprocessors as a platform for modeling framework labs," *Am. J. Phys*. **89**, 596–602 (2021).
3. Danny Doucette, Brian D'Urso, and Chandralekha Singh, "Lessons from transforming second-year honors physics lab," *Am. J. Phys*. **88**, 838–844 (2020).
4. F. Bouquet, J. Bobroff, M. Fuchs-Gallezot, and L. Maurines, "Project-based physics labs using low-cost open-source hardware," *Am. J. Phys*. **85**, 216–222 (2017).
5. AVX part # SCMR22L105SRBB0.
6. Tim Gfroerer, "Circuits in the sun: Solar panel physics," *Phys. Teach*. **51**, 403–405 (2013).
7. Alexander Pusch, Malte S. Ubben, Daniel Laumann, Susanne Heinicke, and Stefan Heusler, "Real-time data acquisition using Arduino and phyphox: Measuring the electrical power of solar panels in contexts of exposure to light in physics classroom," *Phys. Educ*. **56**, 045001 (2021).
8. Fatih Önder, Esra Bilal Önder, and Mehmet Oğur, "Determination of diode characteristics by using Arduino," *Phys. Teach*. **57**, 244–245 (2019).
9. Calin Galeriu, Cheryl Letson, and Geoffrey Esper, "An Arduino investigation of the RC circuit," *Phys. Teach*. 53, 285–288 (2015).
10. https://www.arduino.cc/en/Main/ArduinoBoardUnoSMD.
11. Ginger Walden, Jeremiah Stepan, and Celina Mikolajczak, "Safety considerations when designing portable electronics with electric double-layer capacitors (supercapacitors)," *2011 IEEE Symposium on Product Compliance Engineering Proceedings* (San Diego, CA, 2011), pp. 1–5.

Tim Gfroerer *is a professor and chair of physics at Davidson College. His Energy, the Environment, and Engineering Design course culminates in Tiny Town, a design testing and display event on a sunny December afternoon in Davidson.*
tigfroerer@davidson.edu

Michael Adenew *'24 is a computer science major with a passion for emerging technologies and their ability to transform day-to-day life. After this project, he pursued internships and studies that focused on artificial intelligence, blockchain, and web development, with plans to continue this work after graduation.*

Ella Williams *'24 is a biology major at Davidson College. After graduation, she is interested in using biological processes to improve battery efficiency and capacity.*