An Inexpensive, Open-Source USB Arduino Data Acquisition Device for Chemical Instrumentation

James P. Grinias,† Jason T. Whitfield,‡ Erik D. Guetschow,† and Robert T. Kennedy*,†§

†Department of Chemistry, University of Michigan, Ann Arbor, Michigan 48109, United States
‡Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, Michigan 48109, United States
§Department of Pharmacology, University of Michigan, Ann Arbor, Michigan 48109, United States

Supporting Information

ABSTRACT: Many research and teaching laboratories rely on USB data acquisition devices to collect voltage signals from instrumentation. However, these devices can be cost-prohibitive (especially when large numbers are needed for teaching laboratories) and require software to be developed for operation. In this article, we describe the development and use of an open-source USB data acquisition device (with 16-bit acquisition resolution) built using simple electronic components and an Arduino Uno that costs under $50. Additionally, open-source software written in Python is included so that data can be acquired using nearly any PC or Mac computer with a simple USB connection. Use of the device was demonstrated for a sophomore-level analytical experiment using gas chromatography and a capillary electrophoresis-UV separation on an instrument used for research purposes.

KEYWORDS: Second-Year Undergraduate, Graduate Education/Research, Analytical Chemistry, Laboratory Instruction, Computer-Based Learning, Instrumental Methods, Laboratory Computing/Interfacing, Laboratory Equipment/Apparatus

INTRODUCTION

Open-source microelectronics have become increasingly useful in the field of analytical chemistry due to their low cost and integrated development interfaces. In the research lab, microcontroller boards (such as the Arduino Uno) can be coupled to both in-house and portable instrumentation for user control, data acquisition, and analysis. In undergraduate teaching laboratories, these devices enable students to gain hands-on experience with building and programming the instrumentation used in class projects. Examples of Arduino-based instrumentation used for laboratory experiments include a photometer, automated burets, and a polymerase chain reaction (PCR) thermocycler. During classroom demonstrations, Arduinos have been used to couple sensors such as pH meters or thermometers to either large LCD displays or real-time data graphs that can be displayed on a digital projector. In this technology report, we describe an in-house developed Arduino-based circuit for electronic data acquisition. While data acquisition schemes using Arduinos have been reported for other instruments, this setup utilizes open-source software written in Python to enable easy, low-cost data acquisition compatible with any PC or Mac. With all of the necessary software and hardware instructions provided here, this device can be a useful alternative for USB-driven data acquisition devices requiring more expensive software packages.

HOW IT WORKS

The device utilizes an Arduino Uno and 16-bit analog-to-digital converter (ADC) (ADS1115, Adafruit Industries, New York) for control and data acquisition (Figure 1). The ADC, which...
enables higher resolution data acquisition than is possible with an Arduino Uno alone, is mounted on a solderless breadboard and connected to the microcontroller via jumper cables (full circuit diagrams for setup are provided in the Supporting Information). Though not shown here, an electronics enclosure can be used to protect the device from laboratory spills if needed. During use, the Arduino is connected to a standard USB-port on any laptop or desktop computer to supply power (up to 5 V at ~500 mA) and enable data acquisition. The entire cost of the setup shown in Figure 1 is under $50, which is one-half to one-third the cost of widely used, commercially available USB data acquisition devices. Details on parts and vendors for the device described here, as well as a table of direct comparisons with five commercial USB devices, are included in the Supporting Information.

The open-source software used to control this device was developed in Python. A Python-based toolset for instrument control using Arduino called Instrumentino was previously described, but we have used a different set of routines to simplify the user interface for USB data acquisition. On the instrument front panel (see the Supporting Information), users are able to select the serial port for data transfer to the connected computer, the analog input channel for data acquisition, the data acquisition rate, the file time length, and the file path/name. The data are output as a tab-delimited text file and can easily be replotted and analyzed in any spreadsheet or graphing program. Although data acquisition rates up to 860 Hz are possible with the ADC breakout board, the maximum rate was limited to 500 Hz in the software to ensure stable data acquisition over time. This rate is lower than that achievable by many commercial USB data acquisition setups but is sufficient for most classroom laboratory experiments (excluding the use of ultrafast techniques and oversampling filters)

A key advantage of the open-source software is that it can be installed on nearly any PC or Mac computer (including student laptops, if so desired) without a software license, which can be a cost-saving benefit in teaching and research laboratory settings. With such home-built detectors, there is a need to convert a chemical signal into a voltage that can be detected; the study of these circuits can be added to the Supporting Information. To demonstrate the use of the Arduino Uno-based data acquisition device with our open-source software and compare it to commercial acquisition modes, experiments commonly found in teaching and research laboratory settings were conducted. Within our analytical teaching laboratory (a sophomore-level course typically enrolling about 20 students in a 4 h laboratory period led by a teaching assistant), one two-period experiment involves the separation and identification of unknown samples using gas chromatography with a thermal conductivity detector (GC-TCD). Figure 2A compares chromatograms from the separation of three known analytes (diethyl ether, benzene, and toluene) on a packed GC column acquired using the existing LabJack interface and the Arduino-based device described here. The chromatograms are identical, demonstrating that this open-source device is a cost-effective option for data acquisition with 16-bit signal resolution.

In our own research group, the Arduino Uno was used to acquire data directly from an Agilent 7100 CE instrument with integrated UV detection. Here, the data acquired by the manufacturer software are the same as those obtained with the Uno and Python software for the separation of thiourea, dopamine, and serine. For many time-based experiments on home-built instrumentation where acquisition rates up to 500 Hz are sufficient, such as amperometry or chromatography, this device is an ideal option for data acquisition with 16-bit signal resolution. Commercially available devices capable of 16-bit data acquisition can exceed $300, making them potentially unsuitable for large-scale deployment in teaching settings. Many instruments are able to send a digital signal (via relay or contact closure) to trigger data acquisition. An option to detect this external trigger using a digital input on the Arduino is included in the software (see the Supporting Information) to enable further integration with existing instrumentation. The device described in this report enables logging of voltage signals from both commercial and home-built instrumentation into our open-source software that can be used on nearly any computer at a lower cost than that of many commercial alternatives. To further reduce costs, other parts of chemical instrumentation such as flow control and detection can be constructed and controlled with microcontroller devices whether for research or teaching needs. With such home-built detectors, there is a need to convert a chemical signal into a voltage that can be detected; the study of these circuits can be added to the

![Figure 2](image-url)

**Figure 2.** Comparison of data collected (and normalized to 1 V) using pre-existing software (red traces) to the Arduino-ADS1115 device described in this paper (black traces, offset by 1 V for clarity). In panel A, the separation of 1-propanol, 1-butanol, and 1-pentanol (with air as a dead-time marker) on a packed GC column for a sophomore-level analytical lab is shown. In panel B, the separation of thiourea, dopamine, and serine with a commercial capillary electrophoresis-UV research instrument is demonstrated. Full information on the separation conditions and instrumentation can be found in the Supporting Information.
Technology Report

Journal of Chemical Education

curriculum in both undergraduate and graduate analytical courses to further student understanding of instrument design. If the needs for a specific experiment exceed those of the device described here, a number of other microcontrollers and programming languages can be utilized, and necessary circuit components can be integrated onto a single printed circuit board.

**CONCLUSION**

In this report, we have detailed the design and use of a 16-bit data acquisition device using the Arduino Uno with open-source Python software as a tool for both teaching and research laboratories. At under $50, this device costs significantly less than comparable commercial products; it does not require access to commercial software licenses, either. For laboratory courses that require a number of acquisition setups, Arduinos are a cost-effective option, especially with the included software that can be installed on individual student computers if enough are not available in the lab. With an easy-to-construct layout using widely available parts, this USB data acquisition setup should be ready to use in nearly any setting where a voltage-vs-time signal needs to be collected.

**ASSOCIATED CONTENT**

Supporting Information

The Supporting Information is available on the ACS Publications website at DOI: 10.1021/acs.jchemed.6b00262.

Instructions on use and how to download other Python dependencies, wiring diagram and parts list, and a comparison to commercially available USB data acquisition devices

Copy of the described Python software

**AUTHOR INFORMATION**

Corresponding Author

*E-mail: rtkenn@umich.edu.

Notes

The authors declare no competing financial interest.

**ACKNOWLEDGMENTS**

The authors would like to acknowledge James Vollmer, Mou-Chi Cheng, and Daniel Steyer for help with acquiring the GC data, Claire Ouimet for help with acquiring the CE data, and Brian Coppola for helpful discussions regarding this manuscript. This project was supported by NIH Grant R37 EB003120 to R.T.K., NIH F32 Grant EB019800 to J.P.G., the American Chemical Society Division of Analytical Chemistry Graduate Fellowship sponsored by Eli Lilly and Company to E.D.G., and the University of Michigan Undergraduate Research Opportunity Program Supplementary Research Fund.

**REFERENCES**


DOI: 10.1021/acs.jchemed.6b00262

*J. Chem. Educ.* 2016, 93, 1316—1319